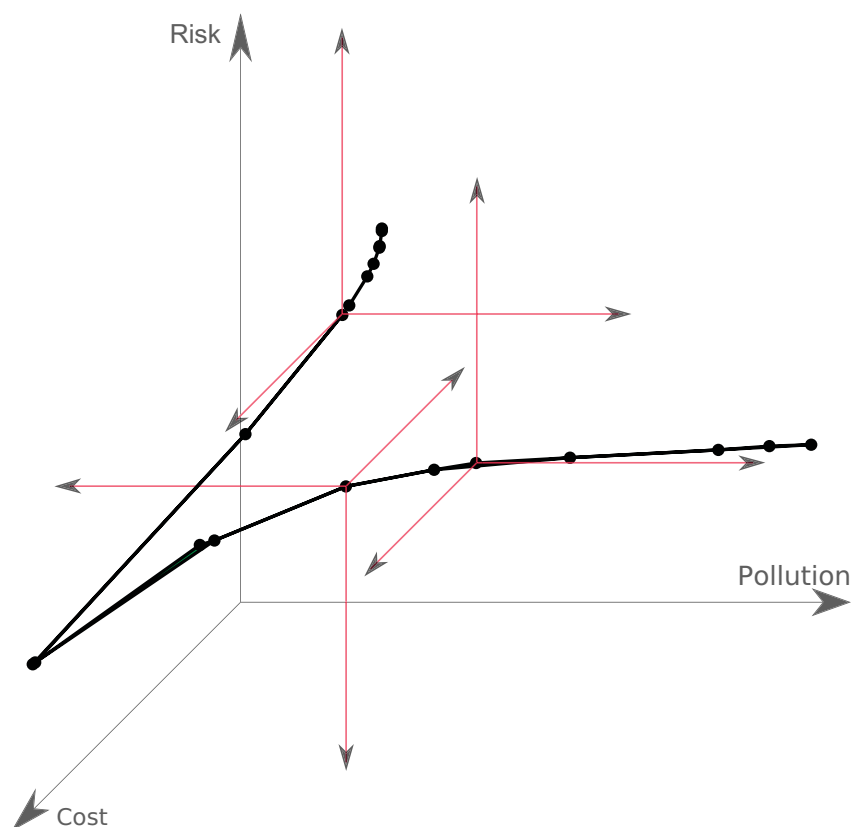


Markus Hartikainen

# Approximation through Interpolation in Nonconvex Multiobjective Optimization



JYVÄSKYLÄ STUDIES IN COMPUTING 141

Markus Hartikainen

Approximation through  
Interpolation in Nonconvex  
Multiobjective Optimization

Esitetään Jyväskylän yliopiston informaatioteknologian tiedekunnan suostumuksella  
julkisesti tarkastettavaksi yliopiston Agora-rakennuksen Alfa-salissa  
joulukuun 10. päivänä 2011 kello 12.

Academic dissertation to be publicly discussed, by permission of  
the Faculty of Information Technology of the University of Jyväskylä,  
in the building Agora, Alfa-hall, on December 10, 2011 at 12 o'clock noon.



UNIVERSITY OF JYVÄSKYLÄ

JYVÄSKYLÄ 2011

Approximation through  
Interpolation in Nonconvex  
Multiobjective Optimization

JYVÄSKYLÄ STUDIES IN COMPUTING 141

Markus Hartikainen

Approximation through  
Interpolation in Nonconvex  
Multiobjective Optimization



UNIVERSITY OF JYVÄSKYLÄ

JYVÄSKYLÄ 2011

Editors

Timo Männikkö

Department of Mathematical Information Technology, University of Jyväskylä

Pekka Olsbo, Ville Korhonen

Publishing Unit, University Library of Jyväskylä

URN:ISBN:978-951-39-4536-7

ISBN 978-951-39-4536-7 (PDF)

ISBN 978-951-39-4535-0 (nid.)

ISSN 1456-5390

Copyright © 2011, by University of Jyväskylä

Jyväskylä University Printing House, Jyväskylä 2011

## ABSTRACT

Hartikainen, Markus

Approximation through Interpolation in Nonconvex Multiobjective Optimization

Jyväskylä: University of Jyväskylä, 2011, 74 p.(+included articles)

(Jyväskylä Studies in Computing

ISSN 1456-5390; 141)

ISBN 978-951-39-4535-0 (nid.)

ISBN 978-951-39-4536-7 (PDF)

Finnish summary

Diss.

In this thesis, we develop a PAINTE method that constructs a Pareto front approximation by interpolating between a given set of Pareto optimal solutions to a (possibly nonconvex) multiobjective optimization problem. The Pareto front approximation produced by the PAINTE method implies a computationally inexpensive mixed integer linear surrogate problem for the original multiobjective optimization problem. The PAINTE method is especially useful in alleviating the computational burden of solving computationally expensive multiobjective optimization problems with interactive multiobjective optimization methods.

The theory behind the PAINTE method is based on the concept of inherent nondominance, which is also introduced in this thesis. We show that inherent nondominance is a desired property for a Pareto front approximation. In addition, the PAINTE method utilizes concepts from computational geometry by using the Delaunay triangulation as the basis for the approximation.

We illustrate the usefulness of the PAINTE method by solving two multiobjective optimization problems with the PAINTE method and the interactive NIMBUS method. The first problem is a three-objective heat exchanger network synthesis problem and the second one is a five-objective problem of designing and operating a wastewater treatment plant. In both problems, the PAINTE method proved to be a very useful tool.

In addition, we introduce a new IND-NIMBUS<sup>®</sup> PAINTE module. With this new module, one can use the PAINTE method together with the interactive NIMBUS method within a single graphical user interface. The PAINTE module is used in the wastewater treatment plant problem mentioned above.

Keywords: Multiobjective optimization, interactive decision making, computational cost, Pareto front approximation, surrogate problem, Pareto optimality, PAINTE

**Author** Markus Hartikainen  
Department of Mathematical Information Technology  
P.O. Box 35 (Agora)  
Fi-40014 University of Jyväskylä  
Finland

**Supervisor** Professor Kaisa Miettinen  
Department of Mathematical Information Technology  
P.O. Box 35 (Agora)  
Fi-40014 University of Jyväskylä  
Finland

**Reviewers** Professor Alexander Lotov  
Dept. of Mathematical Methods for Economic Decision Analysis  
Dorodnicyn Computing Centre of Russian Academy of Sciences  
Vavilova str. 40  
Moscow, 119333 Russia

Dr. Stefan Ruzika  
AG Optimization  
Department of Mathematics  
University of Kaiserslautern  
Paul-Ehrlich-Str. 14-445  
67663 Kaiserslautern, Germany

**Opponent** Professor Serpil Sayın  
College of Administrative Sciences and Economics  
Koç University  
34450 Sariyer, Istanbul, TURKEY

## ACKNOWLEDGEMENTS

First, I would like to thank my supervisor Professor Kaisa Miettinen. She has shown me what it takes to write good scientific texts. Professor Miettinen has also been an inexhaustible source of information during my PhD. thesis project. In addition, she has been an indispensable help in managing all the necessary tasks that go along with having an academic career, like applying for funding. Finally yet importantly, she has introduced me to the academic world by introducing me to her several contacts.

I would like to thank Professor Margaret M. Wiecek who was a visiting professor at our department during her sabbatical from fall 2009 to summer 2010. Although, she was never officially my supervisor she has made the second biggest influence on my scientific thinking after Professor Miettinen. She is the embodiment of scientific rigor.

For evaluating my thesis, I would like to thank Professor Alexander Lotov from the Dept. of Mathematical Methods for Economic Decision Analysis at the Dorodnicyn Computing Center of Russian Academy of Sciences and the Dept. of Systems Analysis at the Lomonosov Moscow State University and Dr. Stefan Ruzika Department of Mathematics at the University of Kaiserslautern.

In addition, I would like to thank all the current and former members of the research group on industrial optimization at the University of Jyväskylä – especially, Dr. Jussi Hakanen, Mr. Tomi Haanpää, Mr. Vesa Ojalehto, Mrs. Suvi Tarkkanen, Dr. Timo Aittokoski and Dr. Karthik Sindhya. It has been a luxury to study in a group of people sharing similar interests and this has definitely brought aspects into my PhD. thesis that I could not have thought of without them.

This research was financially supported by the Academy of Finland (grant number 128495), the Ellen and Artturi Nyysönen foundation and the Jenny and Antti Wihuri Foundation.

Finally, I would like to thank my wife Laura Hartikainen for her support during my PhD. studies. She has assisted me with the use of English and proof-read some of my texts. All the mistakes in my texts are my own, not hers. She has also helped me to channel my thoughts to other things whenever completing my PhD. thesis has felt unsurpassable.



## LIST OF FIGURES

FIGURE 1	The basic concepts of multiobjective optimization with $n = 2$ and $k = 2$ .....	16
FIGURE 2	The real Pareto front, a Pareto front representation and a Pareto front approximation .....	17
FIGURE 3	A possible product of an a posteriori method .....	24
FIGURE 4	The work flow of an interactive method .....	25
FIGURE 5	A screen shot of the graphical user interface of the NIMBUS method in the IND-NIMBUS <sup>®</sup> software framework .....	28
FIGURE 6	The basic functionality of simulation-based optimization .....	30
FIGURE 7	A Pareto front approximation produced by the feasible goals method in the nonconvex case.....	33
FIGURE 8	The three given Pareto optimal outcomes for our example problem.....	36
FIGURE 9	A convex hull approximation of the Pareto front for our example problem.....	37
FIGURE 10	A Pareto front approximation constructed with the feasible goals method for our example problem.....	38
FIGURE 11	The Pareto front approximation constructed with the PAINT method for our example problem .....	39
FIGURE 12	A 3-polytope, its face and its vertex .....	42
FIGURE 13	A Delaunay triangulation of 5 points in $\mathbb{R}^2$ .....	42
FIGURE 14	An inherently nondominated Pareto front approximation for the 4-objective DTLZ2 test problem .....	50
FIGURE 15	A flowchart of the decision making process with PAINT based on a given set of Pareto optimal outcomes .....	51
FIGURE 16	The IND-NIMBUS <sup>®</sup> PAINT module.....	53
FIGURE 17	A schematic layout of the wastewater treatment plant.....	58
FIGURE 18	The inherently nondominated Pareto front approximation that follows rules (R1) and (R2) for the heat exchanger network synthesis problem of Paper [PIII].....	65

## LIST OF TABLES

TABLE 1	The given set of Pareto optimal outcomes $P$ for the four-objective DTLZ2 test problem .....	49
TABLE 2	Polytopes of the inherently nondominated Pareto front approximation that follows rules (R1) and (R2) for the heat exchanger network synthesis problem of Paper [PIII] .....	65

## CONTENTS

ABSTRACT

ACKNOWLEDGEMENTS

LIST OF FIGURES AND TABLES

CONTENTS

LIST OF INCLUDED ARTICLES

1	INTRODUCTION .....	11
2	MULTIOBJECTIVE OPTIMIZATION .....	15
2.1	Multiobjective optimization problem.....	15
2.2	Scalarizing Multiobjective Optimization Problems.....	18
2.2.1	The Linear Scalarization .....	18
2.2.2	The Epsilon Constraint Problem.....	19
2.2.3	Achievement Scalarizing Functions .....	20
2.2.4	The STOM Scalarization .....	21
2.2.5	The Specific Scalarizations in the Synchronous NIMBUS ...	22
2.3	A Posteriori Methods .....	23
2.4	Interactive methods .....	24
2.4.1	The Synchronous NIMBUS Method.....	26
3	ON COMPUTATIONALLY EXPENSIVE MULTIOBJECTIVE OPTIMIZA- TION PROBLEMS .....	29
3.1	Simulation-Based Optimization .....	29
3.2	Approximation in Computationally Expensive Multiobjective Op- timization .....	31
4	TOWARDS THE PAINT METHOD .....	35
4.1	Motivating the PAINT Method .....	35
4.2	Research Objectives .....	39
5	THE PAINT METHOD .....	41
5.1	Prerequisites for the PAINT Method.....	41
5.2	Inherent Nondominance .....	43
5.2.1	An Inherently Nondominated Subcomplex of the Delau- nay Triangulation .....	44
5.3	The PAINT Method .....	45
5.3.1	The Algorithm.....	45
5.3.2	PAINT with the Four-Objective DTLZ2 Test Problem .....	48
5.4	Decision Making and the Surrogate Problem Implied by the Ap- proximation .....	51
6	APPLICATIONS .....	55
6.1	Heat Exchanger Network Synthesis.....	55

6.2	Designing and Operating a Wastewater Treatment Plant .....	57
7	AUTHOR'S CONTRIBUTION .....	60
8	CONCLUSIONS AND FURTHER RESEARCH .....	62
9	ERRATA .....	64
	YHTEENVETO (FINNISH SUMMARY) .....	66
	REFERENCES.....	67
	INCLUDED ARTICLES	

## LIST OF INCLUDED ARTICLES

- PI Markus Hartikainen · Kaisa Miettinen · Margaret M. Wiecek. Decision Making on Pareto Front Approximations with Inherent Non-Dominance. *in Y. Shi and S. Wang and G. Kou and J. Wallenius (editors) New State of MCDM in the 21st Century, Selected Papers of the 20th International Conference on Multiple Criteria Decision Making 2009, pages 35–46, 2011.*
- PII Markus Hartikainen · Kaisa Miettinen · Margaret M. Wiecek. Constructing a Pareto Front Approximation for Decision Making. *Mathematical Methods of Operations Research 73, pages 209–234, 2011.*
- PIII Markus Hartikainen · Kaisa Miettinen. A Computationally Inexpensive Approach in Multiobjective Heat Exchanger Network Synthesis. *in M. Colan (editor) Proceedings of the 2nd International Conference on Applied Operational Research - ICAOR'10, pages 99-109, 2010.*
- PIV Markus Hartikainen · Kaisa Miettinen · Margaret M. Wiecek. PAINT: Pareto Front Interpolation for Nonlinear Multiobjective Optimization. *Computational Optimization and Applications, To Appear.*  
DOI: 10.1007/s10589-011-9441-z.
- PV Markus Hartikainen · Vesa Ojalehto. Demonstrating the Applicability of PAINT to Computationally Expensive Real-life Multiobjective Optimization. *arXiv:1109.3411v1 [cs.NA], 2011.*

# 1 INTRODUCTION

Optimization is the science of finding the best alternative from a set of possible alternatives. Optimization problems can be found in a variety of areas of life. Real life optimization problems include (among many others) finding the optimal use of time e.g., in writing a PhD. thesis and the optimal design of a wastewater treatment plant.

Often for optimization, one needs to model the quantities to be optimized and the matters constraining the problem. Mathematical modeling is usually used for this purpose. It is an art itself to model the phenomena mathematically accurately enough to be able to use the models in optimization, but not to model anything redundant in order not to distract oneself from the optimization task. This thesis recognizes the difficulties in mathematical modeling, but in general, we assume that we are given models that are accurate enough for optimization.

Many real-life phenomena do not have mathematical models that define them in a closed form, i.e., as a mathematical function. This issue has been partly solved by using computer simulations. Many phenomena can be simulated with computers but cannot be modeled in a closed form. This has led to a vast increase in the so-called simulation-based optimization, where optimization methods are combined with simulators (see e.g., [3, 28, 29]). More on simulation-based optimization can be found in Section 3.1.

In practice, many optimization problems tend to have multiple conflicting objectives that need to be considered simultaneously. For example, when buying a car, one needs to consider e.g., performance, fuel consumption, price and appearance. One would naturally want a car with all the best features, but in real life this kind of cars do not exist. This leads to making compromises between the features. Multiobjective optimization is a systematic approach to finding good compromise solutions to problems with multiple conflicting objectives.

An optimization problem with multiple objectives appears different depending on one's role in the decision making process. A decision maker is an expert in the application area, whose aspiration is to find the best possible solution according to his/her preferences as easily as possible. The decision maker may have some preferences on ways of finding the solution, but usually he/she is merely

expected to be able to articulate preferences on different solutions to the problem. The analyst (or the facilitator) is responsible for helping the decision maker find the best solution according to his/her preferences. How the decision maker and the analyst interact in solving the problem depends on the choice of the multiobjective optimization method.

There are different philosophies about the involvement of the decision maker in solving a multiobjective optimization problem. This leads to a categorization of multiobjective optimization methods into no-preference methods, a priori methods, a posteriori methods and interactive methods, as categorized e.g., in [38, 55]. In no-preference methods, the decision maker is not asked any questions but the analyst merely finds a compromise solution according to some preset criteria. No-preference methods are only applicable to a very limited class of problems because the decision maker cannot affect the solution found. In a priori methods, the decision maker is asked for preference information and then the best solution according to this is found. The drawback with a priori methods is that the decision maker may find it hard to define preferences without ever seeing attainable solutions or seeing how the preferences affect the solutions found. In a posteriori methods, a representative set of solutions is found and the decision maker is expected to choose a preferred one. The difficulty with a posteriori methods is that generating a representative solution set may be time-consuming and choosing a preferred solution from a large set of solutions may be hard. In interactive methods, it is understood that any preference information given by the decision maker is only partial and may even be flawed. Thus, the decision maker is allowed to explore different solutions and guide the interactive method to find better solutions. For examples of different multiobjective optimization methods and further explanation, see e.g., [55]. Further aspects of a posteriori and interactive methods are discussed also in Sections 2.3 and 2.4 of this thesis, respectively.

The potential of the philosophy behind interactive methods in finding preferred solutions has been noted e.g., in [61]. The iterative nature of interactive methods allows the decision maker to learn about the problem and his/her preferences [61]. In addition, the decision maker can find a preferred solution without examining too many solutions, which is seen as a drawback for a posteriori methods e.g., in [46]. Furthermore, the cognitive load does not grow too high, because only easily understandable questions (e.g., classification of objectives, which has been found cognitively valid in [46]) are asked at a time (see also [61]). For more information about interactive methods, see e.g., [55, 61] or Section 2.4 in this thesis.

However, a class of problems challenges interactive methods: computationally expensive problems i.e., problems where function evaluations take a long time. In interactive methods, the idea is that the decision maker is present while solutions corresponding to his/her current preferences are computed and this is done on each iteration. For computationally expensive problems, this time may be even days (e.g., for problems like the one studied in [37]). This may lead to the decision maker getting impatient and may even cause him to lose his/her trust in the method, as argued in [45]. Chapter 3 includes further discussion about

computational cost.

In this thesis, we develop an approach to solving (possibly nonconvex) computationally expensive multiobjective optimization problems with interactive methods so that the time that the decision maker has to wait between iterations gets shorter. Our approach to solving these problems is roughly the following: (1) generate a set of solutions to the problem, (2) approximate the possible objective values of the intermediate solutions between the generated ones, (3) find a preferred one from the approximate objective values according to the decision maker's preferences with an interactive method (4) find the solution to the original problem that gives objective values closest to the approximate ones. In practice, Step (1) can be done with any a posteriori multiobjective optimization method. Step (2) is done with a novel PAreto front INTerpolation (PAINT) method, which is developed in this thesis. Step (3) is done by solving with an interactive method the computationally inexpensive multiobjective mixed integer linear problem that represents the approximate objective function values constructed with the PAINT method. This computationally inexpensive problem is also explained in this thesis and can be seen as a surrogate for the original problem in decision making. The objectives of the surrogate problem have the same meaning as the objectives of the original problem and, thus, the decision maker should be able to dictate his/her preferences on them. In Step (4), a vector containing the approximate objective values that are preferred in the Step (3) is taken as the preference of the decision maker for a reference point based a priori method. More specifically, we suggest using an achievement scalarizing problem from [86]. With our approach, the iterations of the interactive method are fast, because the mixed integer surrogate problem is computationally inexpensive.

This thesis is a collection of following papers with the author of this thesis as the main author: Paper [PI] introduces the notion of inherent nondominance for approximating the intermediate solutions of a multiobjective optimization problem and introduces a general form for the surrogate problem that is implied by the Pareto front approximation. Paper [PII] investigates the mathematical basis of using ideas from computational geometry in interpolating between a given set of solutions of a multiobjective optimization problem. Paper [PIII] applies the ideas from the previous paper to solve a multiobjective heat exchanger network synthesis problem. Paper [PIV] introduces the PAINT method that produces the approximation introduced in [PII] and gives a mixed integer linear formulation for the surrogate problem. As shown in Paper [PIV], this new formulation of the surrogate problem can be solved without computational cost with appropriate methods. Finally, Paper [PV] shows how the PAINT method can be used with the interactive NIMBUS method to solve a computationally expensive multiobjective optimization problem of designing and operating a wastewater treatment plant. Paper [PV] also includes a description of a new IND-NIMBUS<sup>®</sup> PAINT module that enables the use of the PAINT method with the interactive NIMBUS method.

This rest of this introduction has been divided into two parts. The first part of the introduction consists of Chapters 2, 3 and 4, and the supporting theme in

it is giving background for our research and motivating it. More specifically, in Chapter 2, basic definitions and methods of multiobjective optimization relevant for this thesis are given. In Chapter 3, the notion of computationally expensive multiobjective optimization problems is defined, some reasons for computational cost of optimization problems are discussed and approaches to solving computationally expensive multiobjective optimization problems are summarized. In Chapter 4, the PAIN method is motivated with an example problem and our research objectives are given. The second part of this introduction consists of Chapters 5, 6, 7 and 8, and it describes our research and our contribution. In Chapter 5, the PAIN method is introduced and its use in solving computationally expensive multiobjective optimization problems is discussed. Chapter 6 includes two applications. Finally, in Chapter 7, author's contribution in this thesis is described, and Chapter 8 concludes with some ideas for further research.



## 2 MULTIOBJECTIVE OPTIMIZATION

### 2.1 Multiobjective optimization problem

Figure 1 illustrates graphically the basic concepts of multiobjective optimization. Mathematically, a multiobjective optimization problem can be defined as

$$\begin{aligned} \min & \quad (f_1(x), \dots, f_k(x)) \\ \text{subject to} & \quad x \in S, \end{aligned} \tag{MOP}$$

where  $f_i : S \rightarrow \mathbb{R}$  are the objective functions,  $k$  is the number of objectives and  $S \subset \mathbb{R}^n$  is the feasible set. A vector  $x \in S$  is called a (feasible) solution. Often a vector valued objective function  $f$  is defined as

$$f : S \rightarrow \mathbb{R}^k : x \mapsto (f_1(x), \dots, f_k(x)).$$

With the function  $f$  one can define the outcomes  $z = f(x)$ , where  $x \in S$ , and the outcome (or feasible objective) set  $f(S) \subset \mathbb{R}^k$ . The set  $\mathbb{R}^k$  is called the outcome (or objective) space. Often, the feasible set is defined by constraint functions  $g_i$ ,  $i = 1, \dots, a$ , i.e., the feasible set can be written as  $S = \{x \in \mathbb{R}^n : g_i(x) \leq 0 \text{ for all } i = 1, \dots, a\}$ . In this thesis, we consider that we want to choose a single solution to problem (MOP) for implementation. In this thesis, this process of choosing a single solution for implementation is called solving the problem.

Without preference information, feasible solutions of a multiobjective optimization problem cannot be completely ordered like the solutions of a single objective optimization problem. There is, however, a natural partial ordering of the feasible set given by the so-called (Pareto) dominance. We say that a vector  $z^1$  in the outcome space (Pareto) dominates another vector  $z^2$  in the outcome space and denote  $z^1 \leq z^2$ , if  $z_i^1 \leq z_i^2$  for all  $i \in \{1, \dots, k\}$  and  $z_j^1 < z_j^2$  for some  $j \in \{1, \dots, k\}$ . In addition, we say that a solution  $x^1 \in S$  (Pareto) dominates another solution  $x^2 \in S$  if the outcome  $f(x^1)$  dominates outcome  $f(x^2)$ . Furthermore, a feasible solution  $x' \in S$  is called Pareto optimal, if there does not exist a feasible solution that dominates it, and an outcome  $z'$  is said to be Pareto optimal,

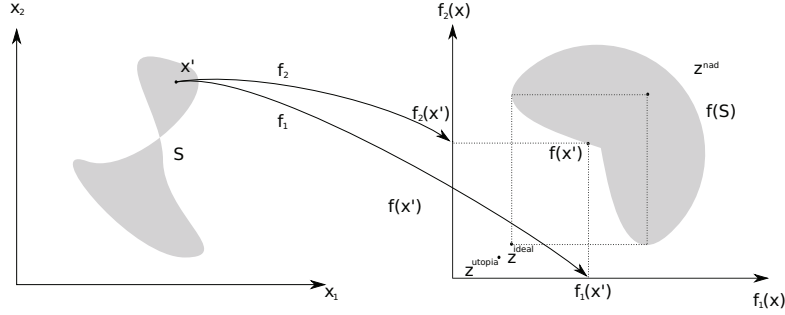


FIGURE 1 The basic concepts of multiobjective optimization with  $n = 2$  and  $k = 2$ . The solution  $x'$  and the outcome  $f(x')$  are Pareto optimal. The vectors  $z^{\text{nad}}$ ,  $z^{\text{ideal}}$  and  $z^{\text{utopia}}$  are respectively the nadir vector, the ideal vector and a utopia vector.

if it is given by a Pareto optimal solution. The set of Pareto optimal outcomes is called the Pareto front and denoted by  $PF$ . Finally, a solution  $x' \in S$  is called weakly Pareto optimal, if there does not exist another solution  $x'' \in S$  so that  $f_i(x'') < f_i(x')$  for all  $i = 1, \dots, k$ . In this thesis, we also need a definition of  $\epsilon$ -proper Pareto optimality. A solution  $x^* \in S$  is  $\epsilon$ -properly Pareto optimal (in the sense of Wierzbicki, see [86]), if there does not exist another solution  $x^{**} \in S$  so that  $\text{dist}(f(x^*) - f(x^{**}), \mathbb{R}_-^k) \leq \epsilon \|x^* - x^{**}\|$ , where  $\text{dist}(\cdot)$  is the Euclidean distance and  $\|\cdot\|$  is the Euclidean norm. If a solution is  $\epsilon$ -properly Pareto optimal for some  $\epsilon > 0$ , but  $\epsilon$  is either unknown or not important, then it is said that the solution is properly Pareto optimal. A locally Pareto optimal solution is a Pareto optimal solution to multiobjective optimization problem, where only a small ball in the feasible set is regarded.

In this thesis, we need to consider nondominated vectors in sets in the objective space and for that we need a couple of definitions. For a set  $A \subset \mathbb{R}^k$ , the set  $PF(A)$  is the set of vectors in  $A$  that are not dominated by any vector in  $A$ . The Edgeworth Pareto hull of problem (MOP) is the set  $EPH = \{z + y : z \in f(S), y \in \mathbb{R}_+^k\}$ , which is the largest set in the objective space with  $PF(EPH) = PF$ .

Some important vectors in the objective space are the nadir vector, the ideal vector and a utopia vector. A vector  $z^{\text{nad}} \in \mathbb{R}^k$  is called a nadir vector of a multiobjective optimization problem (MOP), if

$$z_i^{\text{nad}} = \max\{z_i : z \in PF\}$$

for all  $i = 1, \dots, k$ . A vector  $z^{\text{ideal}} \in \mathbb{R}^k$  is called the ideal vector of problem (MOP), if

$$z_i^{\text{ideal}} = \min\{z_i : z \in f(S)\}$$

for all  $i = 1, \dots, k$ . Finally, a vector  $z^{\text{utopia}} \in \mathbb{R}^k$  with  $z_i^{\text{utopia}} = z_i^{\text{ideal}} - \delta$  for all  $i = 1, \dots, k$  and for some small  $\delta > 0$  is called a utopia vector of problem (MOP).

Since this thesis is about Pareto front approximations and their use in solving computationally expensive multiobjective optimization problems, we need

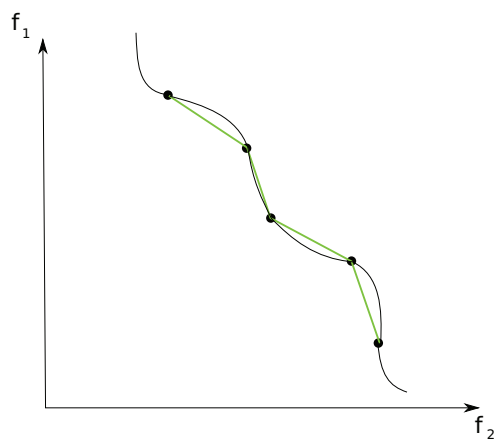


FIGURE 2 The real Pareto front (black curve), a Pareto front representation (black dots) and a Pareto front approximation constructed with the PAINT method (green line segments)

to set our terminology here (that is not completely consistent throughout the literature). First, we adopt the terminology of [75], and define that a Pareto front approximation is a set of points in the outcome space considered a surrogate of (a part of) the Pareto front usually of a simpler structure than the approximated set. Note that we distinguish between terms approximated (the object of approximation) and approximate (something that approximates). With this definition, a vector in a Pareto front approximation is called an approximate Pareto optimal outcome signifying that it approximates Pareto optimal outcomes. In addition, we call a discrete set of Pareto optimal outcomes on the Pareto front a Pareto front representation. In this way, we reserve the word Pareto front approximation for something that is more than a discrete set of computed outcomes. Figure 2 clears these definitions by showing a possible Pareto front, a Pareto front representation and a Pareto front approximation constructed with the PAINT method (developed in this thesis) for a two-objective optimization problem. We also classify different Pareto front approximations (following [75]) to inner approximations, outer approximations, sandwich approximations and others. An inner (outer) approximation contains points  $a$  in the outcome space so that, if  $a_i = z_i$ , where the vector  $z \in \mathbb{R}^k$  is a Pareto optimal outcome and the index  $i \in \{1, \dots, k\}$ , then  $a_j \geq z_j$  ( $a_j \leq z_j$ ) for all  $j \in \{1, \dots, k\} \setminus \{i\}$ . A sandwich approximation contains both an inner approximation and an outer approximation.

Next, in Section 2.2, we will describe different scalarizing functions used in multiobjective optimization. Scalarizing functions are essential to multiobjective optimization as they transform multiobjective optimization problems into single objective ones. Often, they include a way of taking into account the decision maker's preferences while constructing the problem.

## 2.2 Scalarizing Multiobjective Optimization Problems

The purpose of scalarization is to turn multiobjective optimization problems (MOP) into single objective optimization problems. Usually, these scalarizations include a way of taking into account the decision maker's preferences. Those preferences can be elicited as e.g., trade-off information, desirable objective function values or marginal rates of substitution and they are incorporated into the scalarization as parameters. Through scalarization, it is possible to take advantage of the well-developed methods of single objective optimization.

Technically, the scalarization is accomplished by introducing a new objective function, which somehow depends on the objective function values of the multiobjective optimization problem and possibly by introducing additional constraints. All the scalarizations discussed in this thesis are of the form

$$\begin{aligned} \min \quad & s_p(f(x)) \\ \text{subject to} \quad & x \in S_p, \end{aligned} \tag{SCAL}$$

where the  $p$  is a parameter describing the preferences of the decision maker in a set of possible (scalarization dependent) parameters  $P$ ,  $s_p : f(S) \rightarrow \mathbb{R}$  is a scalarizing function depending on the parameter  $p$  and  $S_p \subset \mathbb{R}^n$  is a subset of the feasible set  $S$  also depending on the parameter  $p$ .

Desirable properties of scalarizations are discussed e.g., in [79, 83, 85, 86]. With the notation of scalarization (SCAL) the most important properties of scalarization functions can be stated as

1. for any  $p \in P$ , each optimal solution to problem (SCAL) is (perhaps weakly or properly) Pareto optimal, and
2. each (perhaps weakly or properly) Pareto optimal solution is an optimal solution to problem (SCAL) for some  $p \in P$ .

In addition, it is evident from a practical point of view that there should be some understandable way of eliciting the parameters of the scalarization and it should also be easy to understand how the solutions obtained as optimal values of the scalarization correspond to the given parameters. For further desirable properties, see the references mentioned above.

Different scalarizations are compared e.g., in [59, 85, 86]. In the following subsections, examples of different scalarizations are given. These are chosen based on author's preferences, their popularity or because they are used in the interactive synchronous NIMBUS method. The interactive NIMBUS method is later described in Section 2.4.1.

### 2.2.1 The Linear Scalarization

Linear scalarization (also called the weighted sum or the weighting problem), see e.g., [55], is perhaps the simplest scalarization available. The linear scalarization

of (MOP) is optimization problem

$$\begin{aligned} \min & \quad \sum_{i=1}^k \lambda_i f_i(x) \\ \text{subject to} & \quad x \in S \end{aligned} \quad (1)$$

with weights of the objectives in a vector  $\lambda \in \{\lambda \in [0, 1]^k : \sum_{i=1}^k \lambda_i = 1\}$  describing relative importance of the objectives.

Each optimal solution to problem (1) is at least weakly Pareto optimal and Pareto optimal, if  $\lambda_i > 0$  for all  $i = 1, \dots, k$  and, thus, the first desirable property is satisfied. It is, however, a well-known drawback that, if a multiobjective optimization problem is nonconvex, then there exist so-called unsupported solutions that do not optimize problem (1) for any  $\lambda \in \{\lambda \in [0, 1]^k : \sum_{i=1}^k \lambda_i = 1\}$ . For details, see e.g., [55].

Furthermore, weights have been criticized for being hard to understand, see e.g., [55]. As argued e.g., in [86], the typical interpretation of weights through corresponding marginal substitution rates ( $m_{ij} = w_j/w_i$ ) does not help, because the concept of marginal substitution itself is the dual of preference. Finally, as illustrated in an example in [82], weights that appear good may produce bad solutions and weights that appear bad may produce good solutions. This behavior of the weights further emphasizes their bad interpretability.

Even though there are many methods for eliciting weights (e.g., swing weighing, see [27]), we find the previous arguments so compelling that we do not use weights in this thesis as sole means of representing decision maker's preferences. In addition, we find the notion of a global linear preference model too restrictive. In the following sections, we discuss less restrictive ways of modeling preferences.

### 2.2.2 The Epsilon Constraint Problem

The epsilon constraint problem [34] is often the second scalarization that people come to think of. The epsilon constraint problem related to multiobjective optimization problem (MOP) is

$$\begin{aligned} \min & \quad f_j(x) \\ \text{subject to} & \quad x \in S \\ & \quad f_i(x) \leq \epsilon_i \text{ for all } i \neq j \end{aligned} \quad (2)$$

for some  $j \in \{1, \dots, k\}$  with epsilon constraints in vector  $\epsilon \in R^k$ .

Each optimal solution to problem (2) is a weakly Pareto optimal solution to problem (MOP). Furthermore, if a solution  $x^*$  is an optimal solution to problem (2) for all  $j = 1, \dots, k$  with  $\epsilon_i = f_i(x^*)$  for all  $i = 1, \dots, k$ , then the solution  $x^*$  is a Pareto optimal solution to problem (MOP). Finally, each Pareto optimal solution to problem (MOP) is an optimal solution to problem (2) with epsilon constraints  $\epsilon_i = f_i(x^*)$  for all  $i = 1, \dots, k$ . Thus, the epsilon constraint problem satisfies the two desirable properties for scalarizations. For details, see e.g., [55].

In addition, the epsilon constraints are easy to understand and it is fairly easy to interpret how the change in the constraints would change the optimal

solution. The epsilon constraint, however, has some drawbacks, also. If the epsilon constraints are too optimistic, problem (2) might not have any solutions. In addition, if the constraints are too pessimistic, the entire possible minimization is done on the  $j^{\text{th}}$  objective, which may not be preferable by the decision maker. Thus, the decision maker must have a good understanding about the problem in order to use the epsilon constraint problem efficiently. This notion was reflected in [84], where it was argued that the preferable objective function values given by the decision maker should be treated as aspirations levels, not as constraints. This idea is pursued in the so-called achievement scalarizing functions [84, 86], which are discussed in the following subsection.

### 2.2.3 Achievement Scalarizing Functions

The study of using aspiration levels in multiobjective optimization led to achievement scalarizing functions (see [83, 84, 86]). Aspiration levels are values of objectives that should desirably be achieved, and they can be represented as a reference point  $\bar{z} \in \mathbb{R}^k$ .

Achievement scalarizing functions (here denoted by functions  $s_{ach} : f(S) \times R \rightarrow \mathbb{R}$ , where the set  $R$  is a function dependent set of possible reference points) are desirably order-consistent, which they further partition into either order-representing or order-approximating with a precision parameter  $\epsilon > 0$  (see [86] for formal definitions) achievement scalarizing functions. The achievement scalarizing problem implied by multiobjective optimization problem (MOP) is

$$\begin{aligned} \min \quad & s_{ach}(f(x), \bar{z}) \\ \text{subject to} \quad & x \in S, \end{aligned} \quad (3)$$

where the reference point is  $\bar{z} \in R$  and the function  $s_{ach}$  is an achievement scalarizing function. Assuming that a vector  $x^*$  is an optimal solution to problem (3) for some  $\bar{z} \in R$ , then, according to [86],  $x^*$  is weakly Pareto optimal to problem (MOP), if the achievement scalarizing function is order-representing, and properly Pareto optimal to problem (MOP), if the achievement scalarizing function is order-approximating. On the other hand, if a solution  $x^*$  is weakly Pareto optimal and the achievement scalarizing function is order-representing, then  $x^*$  is an optimal solution to problem (3) with  $\bar{z} = f(x^*)$  and, finally, if a solution  $x^*$  is  $\epsilon'$ -properly Pareto optimal, and the achievement scalarizing function is order-approximating with adequately small precision parameter  $\epsilon'$ , then  $x^*$  is an optimal solution to problem (3) with  $\bar{z} = f(x^*)$ .

In [86], there exist many order-consistent achievement scalarizing functions and, in addition, some more recent ones have been introduced e.g., in [69]. In this thesis, we use only the order-approximating achievement scalarizing problem mentioned also in [86], which is

$$\begin{aligned} \min \quad & \max_{i=1, \dots, k} \left[ \frac{f_i(x) - \bar{z}_i}{z_i^{\text{nad}} - z_i^{\text{utopia}}} \right] + \rho \sum_{i=1}^k \frac{f_i(x)}{z_i^{\text{nad}} - z_i^{\text{utopia}}} \\ \text{subject to} \quad & x \in S, \end{aligned} \quad (4)$$

where the term  $\rho \sum_{i=1}^k \frac{f_i(x)}{z_i^{\text{nad}} - z_i^{\text{utopia}}}$  is called the augmentation term,  $\rho > 0$  is a small constant, and  $z^{\text{nad}}$  and  $z^{\text{utopia}}$  are the nadir vector and a utopia vector, respectively. This problem is also used as one of the subproblems in the interactive synchronous NIMBUS method.

In the next subsection, we introduce the scalarization that is used in the interactive satisficing trade-off method. This scalarization is closely related to the achievement scalarization function. It was introduced more or less at the same time with the achievement scalarizing problem (4). The basic question when comparing these two scalarizations is how to interpret the objective function values in a reference point and they have taken slightly different approaches.

#### 2.2.4 The STOM Scalarization

The satisficing trade-off method (commonly known as STOM) is an interactive multiobjective optimization method, introduced in [67]. The scalarization used in the STOM method is based on a reference point and, in this thesis, this scalarization is called the STOM scalarization. We will not further discuss the STOM method in this thesis, but refer to [55, 67] for details. The STOM scalarization is important for this thesis, because it is used in the interactive synchronous NIMBUS method.

The basic formulation of STOM scalarization guarantees only weakly Pareto optimal solutions. The augmented version of STOM scalarization

$$\begin{aligned} \min \quad & \max_{i=1, \dots, k} \left[ \frac{f_i(x) - z_i^{\text{utopia}}}{\bar{z}_i - z_i^{\text{utopia}}} \right] + \rho \sum_{i=1}^k \frac{f_i(x)}{\bar{z}_i - z_i^{\text{utopia}}} \\ \text{subject to} \quad & x \in S, \end{aligned} \quad (5)$$

however, guarantees (proper) Pareto optimality of the optimal solutions. Here the constant  $\rho$  is again a small positive constant. There are two differences between the augmented STOM scalarizing function above and the augmented achievement scalarizing function. First, the term under the maximal operator has changed from  $\frac{f_i(x) - \bar{z}_i}{z_i^{\text{nad}} - z_i^{\text{utopia}}}$  to  $\frac{f_i(x) - z_i^{\text{utopia}}}{\bar{z}_i - z_i^{\text{utopia}}}$ . The term in the achievement scalarizing function leads to minimizing the weighted Chebyshev distance from the reference point, when the STOM scalarizing function leads to minimizing the weighted Chebyshev distance from a utopia point and the reference point sets weights for different objectives.

The reason for the augmentation terms in the achievement scalarizing function and the STOM scalarizing function is the same – to remove weakly Pareto optimal solutions from their optimal solutions. Note that the STOM scalarizing function has a slight advantage, since it does not require the computation of the nadir point, which is not easy, as discussed e.g., in [20]. The decision maker must, however, be shown the ideal point, so that he/she does not give reference point values better than that.

As with the achievement scalarizing problem, all the  $\epsilon$ -properly Pareto optimal solutions can be attained as optimal solutions to problem (5). The  $\epsilon$  depends

here on the constant  $\rho$ . With what was discussed above, this means that the STOM scalarization (5) satisfies the first desirable property with normal Pareto optimality and the second one with  $\epsilon$ -proper Pareto optimality.

In the next subsection, we expand the possibilities in ways how the decision maker's preferences can be modeled by introducing two scalarizations used in the interactive synchronous NIMBUS method. The Synchronous NIMBUS method uses four different scalarizations that are the achievement scalarizing function (4), the STOM scalarization (5) and two other scalarizations that are introduced in the next subsection.

### 2.2.5 The Specific Scalarizations in the Synchronous NIMBUS

In this section, we give the two remaining scalarizations of the interactive synchronous NIMBUS method. These classifications also allow the decision maker to classify the objectives into classes depending on preferred changes in their values. In this section, we will only describe the scalarizations and their use in the synchronous NIMBUS method is discussed in Section 2.4.1. These scalarizations are not used in this form in any other method than the synchronous NIMBUS method.

The first one of the remaining scalarizations in the synchronous NIMBUS is

$$\begin{aligned} \min \quad & \max_{i \notin I^{<>}} \left[ \frac{f_i(x) - z_i^{nad}}{z_i^{nad} - \bar{z}_i} \right] + \rho \sum_{i=1}^k \frac{f_i(x)}{z_i^{nad} - \bar{z}_i} \\ \text{subject to} \quad & x \in S, \end{aligned} \quad (6)$$

where  $I^{<>}$  is the (possibly empty) set of indices of objectives classified by the decision maker as ones that can change freely and  $\bar{z}$  is a reference point. This scalarization is based on the scalarization used in the GUESS method [12].

The NIMBUS scalarization of the synchronous NIMBUS is

$$\begin{aligned} \min \quad & \max_{i \in I^{<}, j \in I^{\leq}} \left[ \frac{f_i(x) - z_i^*}{z_i^{nad} - z_i^{\text{utopia}}} \cdot \frac{f_j(x) - z_j}{z_j^{nad} - z_j^{\text{utopia}}} \right] + \rho \sum_{i=1}^k \frac{f_i(x)}{z_i^{nad} - z_i^{\text{utopia}}} \\ \text{subject to} \quad & f_i(x) \leq f_i(x_c) \text{ for all } i \in I^{<} \cup I^{\leq} \cup I^= \\ & f_i(x) \leq \epsilon_i \text{ for all } i \in I^{\geq} \\ & x \in S. \end{aligned} \quad (7)$$

In the above problem, the sets  $I^{<}$ ,  $I^{\leq}$ ,  $I^=$  and  $I^{\geq}$  contain indices of objective functions that the decision maker has classified as those that he/she wants to improve as much as possible, wants to improve to aspiration levels denoted by  $z_i$ , wants to remain unchanged and allows to get worse until bounds denoted by  $\epsilon_i$ .

In both problems,  $\rho > 0$  is a small constant. The augmentation terms differ slightly, but both of them remove weakly Pareto optimal solutions from the sets of optimal solutions to the problems.

With appropriate parameters (i.e., classification, reference point, aspiration levels and bounds, see [60]), optimal solutions to either one of problems (5) or (7) are always Pareto optimal to multiobjective optimization problem (MOP). On the other hand, each  $\epsilon$ -properly Pareto optimal solution is an optimal solution to



either one of the problems with suitable reference point, classification, aspirations and bounds, where  $\epsilon$  depends on the constant  $\rho$ . Thus, desirable properties 1. and 2. are satisfied with scalarizations in problems (6) and (7).

In the next section, we will discuss a posteriori methods. When scalarizations can be used to find a single Pareto optimal solution to a multiobjective optimization problem, a posteriori methods can be used to find a representation of the Pareto optimal solutions to a problem (i.e., a discrete set of Pareto optimal solutions that approximates the set of Pareto optimal solutions). Some a posteriori methods achieve this by solving a number of scalarizations with different parameters and some use other methods.

### 2.3 A Posteriori Methods

The idea in a posteriori methods is to generate a large set of different solutions from which the decision maker can choose. Because every non-Pareto optimal solution is dominated by a Pareto optimal solution, one knows that one may concentrate on producing a large set of Pareto optimal solutions. A possible product of an a posteriori method is illustrated in Figure 3.

In a posteriori methods, where no preference information is available, the aim is to produce a representation that approximates the Pareto front as well as possible. However, defining what it means for a discrete representation of Pareto optimal outcomes to approximate the Pareto front (which is often not discrete for continuous problems) is problematic (see e.g., [26, 90]).

A posteriori methods can be divided into two classes: deterministic (or classical) methods and evolutionary methods (see e.g., [80]). The main approach of deterministic a posteriori methods (although others also exist) is to solve multiple scalarizations in a way that produces the whole Pareto front (if possible) or a good representation on it (see e.g., [55]). Evolutionary multiobjective optimization methods are heuristic search-based methods, where an iterative search pattern tries to find a set that approximates the Pareto front as well as possible and, also, is as diverse as possible (see e.g., [19]).

Surveys of deterministic a posteriori methods can be found e.g., in [55, 75, 80]. In [75], approximation methods (i.e., methods producing more than a discrete set of Pareto optimal outcomes) are also surveyed, but the so-called  $0^{th}$  order approximation methods are deterministic a posteriori methods, as defined in this thesis. Since most deterministic a posteriori methods count on solving multiple scalarizations, their functionality depends greatly on the ability to solve them as single objective optimization problems. In addition, another basic question is how to select the parameters. A good discrete approximation of the parameter set  $P$  in scalarization (SCAL) will not necessarily produce a good representation of the Pareto front.

Evolutionary multiobjective optimization methods are very popular nowadays. Surveys of evolutionary a posteriori methods can be found e.g., in [17,

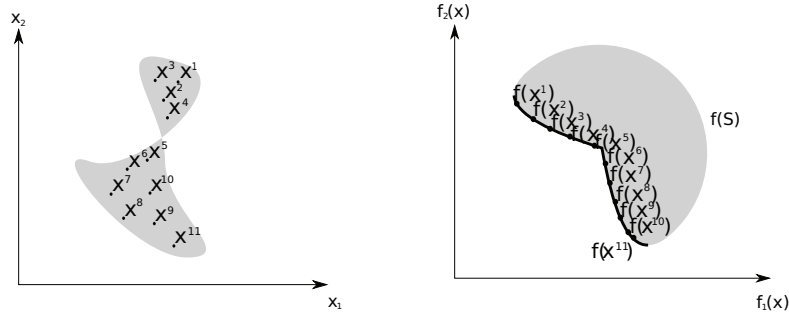


FIGURE 3 A possible product of an a posteriori method in the outcome space and in the feasible set. The actual Pareto front is marked with a black curve.

19]. In [19], a very popular evolutionary multiobjective optimization algorithm NSGA-II is also described. In evolutionary methods, a set of solutions is evaluated at a time and operators, like selection, mutation and crossover, are applied to make the set converge to the Pareto front. Evolutionary methods are often less effective than deterministic methods in finding Pareto optimal solutions to differentiable and convex multiobjective problems, but excel in problems involving disconnectedness or multiple local, but not global, Pareto optima. For a comparison of deterministic methods with evolutionary methods, see e.g., [80].

The use of (both deterministic and evolutionary) a posteriori methods has three main difficulties: (1) It is not evident what is a good representation of the Pareto front, see e.g., [26, 90], (2) producing a good representation of the Pareto front may be time consuming and (3) the decision maker may have difficulties in choosing the best alternative from a large set of Pareto optimal outcomes as argued by [46]. For this reason, we discuss interactive multiobjective optimization methods in the following subsection. The interactive methods avoid the above difficulties by letting the decision maker guide the search for better solutions (according to his/her preferences) and by asking the decision maker only simple, easily understandable questions at a time.

## 2.4 Interactive methods

In interactive methods, the idea is to form a solution pattern, where the decision maker is allowed to guide the search for better solutions (according to his/her preferences), and then repeat this pattern until a solution preferred by the decision maker is found. More specifically, the general work flow is the following: (1) start with a single or a small set of solutions, (2) let the decision maker choose one and specify preferences for improvement, (3) find a new solution or a small set of solutions that match the updated preferences of the decision maker and (4) if the decision maker is not satisfied with the solution(s) found, repeat the

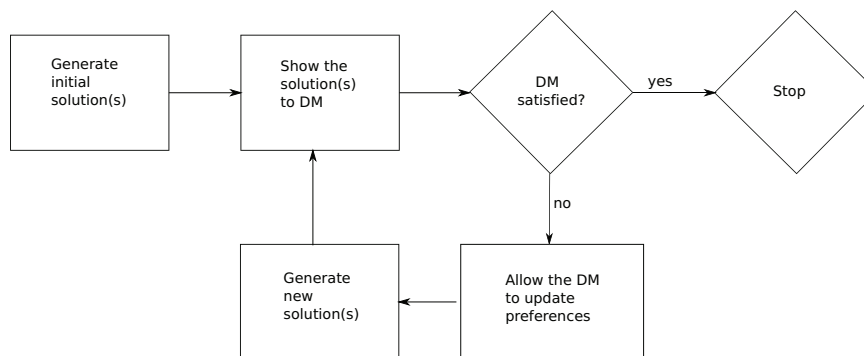


FIGURE 4 The work flow of an interactive method with a decision maker (DM)

steps (2)-(4). This is shown also in Figure 4.

Most interactive methods generate the initial and the new solutions using scalarizations (see Section 2.2). Naturally, different methods use different scalarizations. In addition, interactive methods differ from each other in the way that they elicit preferences from the decision maker. Ways to elicit preferences used in interactive methods are e.g., classification of objectives, asking for a new reference point and asking for (local) trade-off information. Furthermore, interactive methods use different ways to inform the decision maker about the solutions found e.g., visualizations or trade-off information. For more details, see e.g., [51].

The potential of interactive methods is in their support for the decision maker [61]. Because the decision maker can iteratively see the Pareto optimal solutions that have been generated so far based on his/her preferences, the decision maker can learn about the problem and his/her preferences while solving it [9, 61]. In [61], the use of an interactive method is divided into two phases: a learning phase and a decision phase. In the learning phase, the decision maker learns about the problem and finds an interesting region of solutions. In the decision phase, a preferred solution is picked out from the region found in the learning phase. In [61], it is reminded that these phases may also be used iteratively if so desired.

Another benefit of interactive methods is that the decision maker can guide the search for better solutions. Thus, only a small set of Pareto optimal solutions has to be usually examined and this leads to a gain in computational efficiency. This may be crucial for computationally expensive multiobjective optimization problems. In addition, the need to consider only a single or a small set of solutions at a time leads to a lower cognitive load.

There are also some drawbacks to interactive methods. Most importantly, the decision maker has to be willing to invest time in trying to understand the problem and the tools that he/she is given. This may be a problem, because many real-life decision makers may be rather busy. If the multiobjective optimization problem is computationally expensive, iterations of the interactive method may take a long time. According to [45], long iteration time may lead the deci-

sion maker to get tired and prematurely stop the iterative process. Thus, he/she would not get good enough an understanding about the problem in the learning phase and would end up with a less preferred solution. In Chapter 3, we discuss computationally expensive multiobjective optimization problems and ways of resolving the drawbacks of computational cost that are available in the literature. In this thesis, we also develop a method for resolving this problem of computational cost, and this method and the theory leading to it are discussed in the second part of this thesis.

There are numerous interactive methods (e.g., the Step method [10], the interactive surrogate worth trade-off method [15, 16]) in the literature. In this thesis, we will only describe the synchronous NIMBUS method in a greater detail. The NIMBUS method is necessary for this thesis, because it will be used together with our PAINT method in the second part of this thesis. For other interactive methods, we refer to [55] and to references therein.

#### 2.4.1 The Synchronous NIMBUS Method

The synchronous NIMBUS method (introduced in [60]) is an extension of the NIMBUS method, which was introduced in [57]. The word synchronous refers to the fact that four different ways to use the same preference information are used at the same time instead of just one way like in most of the other interactive methods. In this thesis, we sometimes drop the word synchronous and refer to the synchronous NIMBUS method merely as the NIMBUS method.

The NIMBUS method starts with a so-called neutral compromise solution or with a solution given by the decision maker. The NIMBUS method uses classification of objectives as preference information. Given a Pareto optimal solution to a multiobjective optimization problem, the decision maker can classify the objective functions of the current solution into classes  $I^<$ ,  $I^{\leq}$ ,  $I^=$ ,  $I^{\geq}$  and  $I^{>}$ . These classes are defined, respectively, as classes of objective functions that the decision maker wants to improve as much as possible, wants to improve to an aspiration level  $z_i$ , feels acceptable, allows to get worse until a bound  $\epsilon_i$  and allows to change freely for a while. If the sets  $I^{\leq}$  or  $I^{\geq}$  are used, then also the aspiration levels  $z_i$  and the bounds  $\epsilon_i$  are asked from the decision maker. The decision maker must be willing to sacrifice something and, thus, the set of objectives that are allowed to get worse cannot be empty.

The preferences of the decision maker are then transferred into up to four different single objective optimization problems i.e., scalarizations. These are the NIMBUS scalarization (7), scalarization (5) of the satisficing trade-off method, an achievement (scalarizing) function (4) and scalarization (6), which is based on the scalarization in the GUESS method. The last three scalarizations are reference point based, so there is a need to transfer the classification given by the decision maker into a consistent reference point. In [60], such a reference point  $\bar{z}$  is constructed by setting  $\bar{z}_i = z_i^{\text{ideal}}$  for objectives in the class  $I^<$ ,  $\bar{z}_i = z_i$  (where  $z_i$  is the aspiration level defined by the decision maker) for objectives in the class  $I^{\leq}$ ,  $\bar{z}_i = z_i^{\text{current}}$  (where  $z^{\text{current}}$  is the current Pareto optimal outcome) for objectives

in the class  $I^=$  and  $\bar{z}_i = \epsilon_i$  (where  $\epsilon_i$  is the bound defined by the decision maker) for objectives in the class  $I^{\geq}$ , and by setting  $\bar{z}_i = z_i^{\text{nad}}$  for objectives in the class  $I^{<>}$ . Here,  $z^{\text{ideal}}$  is the ideal vector of the multiobjective optimization problem and  $z^{\text{nad}}$  is the nadir vector.

The decision maker can choose how many new Pareto optimal solutions (from one to four) he/she wants to be computed with each classification. The amount of solutions chosen by the decision maker is computed with the scalarizations and these solutions are shown to the decision maker. Solutions selected by the decision maker are saved for possible further inspection. The decision maker has now a chance to compute so-called intermediate solutions between any of the solutions (see e.g., [60] for details) that have been previously saved and these solutions can also be saved. If a solution is found, where the decision maker does not wish to decrease any of the objective function values, the NIMBUS algorithm is stopped and this solution is taken as the final solution. On the other hand, the decision maker has the chance to take any of the saved solutions as the starting point of a new classification and the solution pattern is repeated.

There exist two implementations of the NIMBUS method: the first one is WWW-NIMBUS and the other one is implemented in the IND-NIMBUS<sup>®</sup> software framework. WWW-NIMBUS (described in [58, 60]) is the first interactive multiobjective optimization method operating via the Internet (available at <http://nimbus.mit.jyu.fi/>). WWW-NIMBUS can be used with most web browsers and it does the calculations on a server. The current version of WWW-NIMBUS uses the synchronous version of NIMBUS. The implementation has a set of single objective optimization solvers that can be used to solve the scalarizations. It also implements different styles of visualizing the Pareto optimal solutions. The problems can be inputted into WWW-NIMBUS using standard mathematical operators available in it.

IND-NIMBUS<sup>®</sup> is a multi-platform desktop software framework, currently available for Windows and Linux operating systems, intended to provide flexible tool-set for developing multiobjective optimization methods. The IND-NIMBUS framework has been among others used to implement the synchronous NIMBUS method. It should be noted, that the IND-NIMBUS<sup>®</sup> software has been developed for industrial applications, and it does not include any tools for formulating the problem. Instead, IND-NIMBUS<sup>®</sup> has to be always connected to an external source that models the problem. Figure 5 contains a screen shot of the graphical user interface of the NIMBUS method in the IND-NIMBUS<sup>®</sup> software framework.

The implementation of the NIMBUS method in the IND-NIMBUS<sup>®</sup> software framework has (among others) been successfully applied to multiobjective optimization of simulated moving bed processes in [35], continuous casting of steel in [56], brachytherapy planning in [74], heat exchanger network synthesis in [47]. In all of these cases, IND-NIMBUS<sup>®</sup> has been connected to an external a modeling or simulator tool that computes the objective and constraint function values.

In addition to the synchronous NIMBUS method, the IND-NIMBUS<sup>®</sup> software framework has already been used to implement the Pareto Navigator method (introduced in [25]). In this thesis, we develop a new IND-NIMBUS<sup>®</sup> PAINT

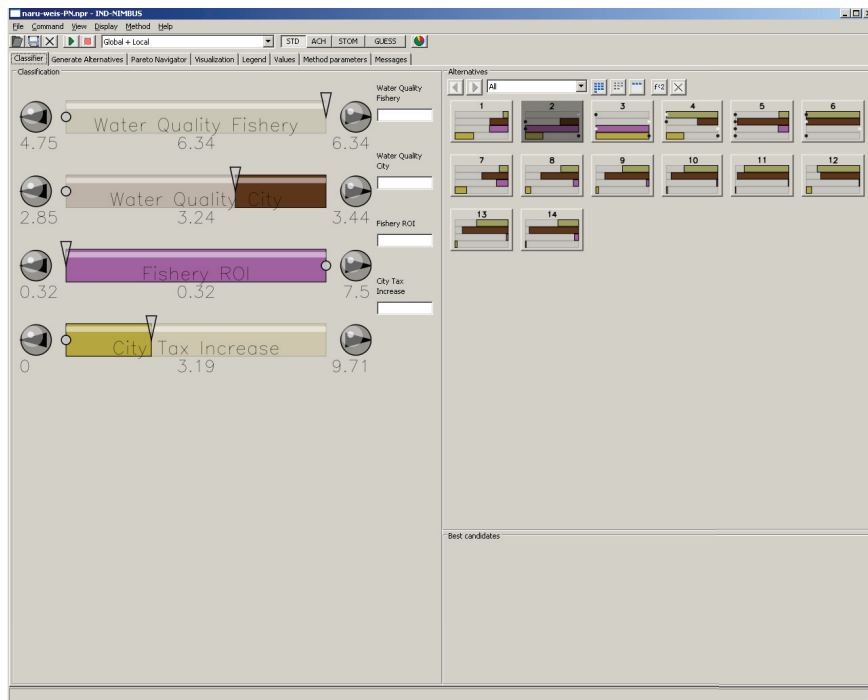


FIGURE 5 A screen shot of the graphical user interface of the NIMBUS method in the IND-NIMBUS<sup>®</sup> software framework

module that combines our Pareto front approximation method PAINT (described in Section 5.3) and the NIMBUS method.

### **3 ON COMPUTATIONALLY EXPENSIVE MULTIOBJECTIVE OPTIMIZATION PROBLEMS**

In this thesis, we say that a multiobjective optimization problem is computationally expensive, if the time taken by evaluating the objective functions is significant, when solving the problem. This naturally leads to complications in solving the multiobjective optimization problem, because the objective functions must usually be evaluated multiple times when solving the problem. With this definition, many real-life optimization problems are computationally expensive (see e.g., [1, 36, 37, 47, 88]).

Perhaps the most common cause of computational cost is the need to do computationally expensive simulations to find the values of the objective functions and simulation-based optimization is discussed in the following section. On the other hand, some objective functions may have a closed form formulation, but the elementary functions needed for it may be so complicated that evaluating them takes time. In addition, the objective functions may even need to be measured (as e.g., in [5]) which surely is time consuming.

#### **3.1 Simulation-Based Optimization**

By a simulator, we mean in this thesis a computer program that can imitate real-life systems or phenomena for given values of decision variables (often called parameters of the simulation). Simulation is used when the real system or phenomenon cannot be studied directly, because they may be inaccessible, or they may be dangerous or unacceptable to engage. It might even be that the system is being designed but has not yet been built. In this thesis, we use (as an example and a proof of concept for our method) the GPS-X simulator that can simulate wastewater treatment plants. Other systems that need to be simulated are e.g., weather (see e.g., [13]) and chemical processes within e.g., a paper making line (see [11]). For further details on simulation, see e.g., [7, 81]

A single run of a simulator only predicts the behavior of the system or the

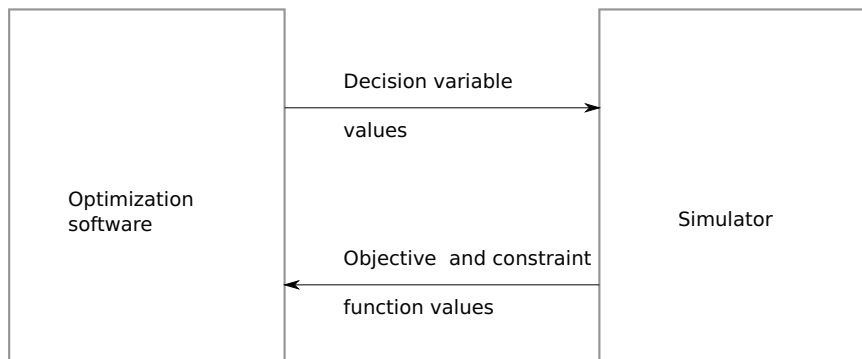


FIGURE 6 The basic functionality of simulation-based optimization

phenomenon for a single value of the decision variables and, thus, simulation is not usually enough as itself. Often, one must find the optimal values of the decision variables and a single run of a simulator does not provide any information about how this can be done. In simulation-based optimization, an optimization algorithm is combined with a simulator to find the optimal values of the decision variables (see e.g., [3, 28, 29]).

Even single objective simulation-based optimization is iterative by nature. An iterative search for an optimal solution starts with a single solution (or possibly a set of solutions) depending on the optimization algorithm. Both the values of the objective functions and the constraints (or directly the feasibility of the current solution(s)), or just one of the above, are then extracted from the output of the simulator. After inspecting the solutions, the optimization algorithm either stops, if an optimal solution (or close enough) has been found, or sends new solution(s) for the simulator. In this way, the simulator and the optimization algorithm alternate until a solution close enough to optimality has been found. Note that, if the simulator only outputs the values of the objectives (and possibly the constraints) and no other assumptions are made on the objective functions, the actual optimality of the solution cannot be known without examining each solution, which is not possible for most problems. The basic functionality of simulation-based optimization is captured in Figure 6. In this Figure, we assume that a single simulator outputs both the values of the objective function(s) and the constraints.

A way to alleviate the computational burden of solving computationally expensive multiobjective optimization problems is to do approximation. In the next section, we identify two different types of approximation that can be used in solving computationally expensive multiobjective optimization problems. We concentrate on approximations that can be used with either a posteriori methods or interactive methods.



### 3.2 Approximation in Computationally Expensive Multiobjective Optimization

In this section, we identify two different types of approximation used in computationally expensive multiobjective optimization in the literature. These are approximating the objective functions or the constraints (whichever is computationally expensive) and approximating the Pareto front. We will first discuss approximating the objective functions or the constraints and then move to Pareto front approximations.

In [40], five levels of approximation, when using a surrogate model of the objective functions with evolutionary algorithms, are listed. Most of these levels are, however, applicable to approximating the objective functions (or the constraints) with any method. According to [40], different possible approximation levels are no approximation at all, fully computational simulations (e.g., computational fluid dynamics simulations instead of the wind tunnel experiments), simplified computational simulations (e.g., Quasi-3D simulations instead of real 3D simulations), functional approximations (also known as meta-models) and (in evolutionary algorithms) evolutionary approximation (e.g., fitness inheritance). Note that simulation is one level of approximation, also. In addition, according to [40], another matter (in addition to choosing the approximation level) in approximating the objective functions or the constraints is that the surrogate model should be most accurate in the areas of interest (i.e., close to the Pareto front in multiobjective optimization) and the model should be improved as much as possible with the given data.

Surrogate models of the objective functions have been used extensively with a posteriori methods. Because the aim of a posteriori methods of multiobjective optimization is to produce a good representation of the Pareto front, as default they need to employ a lot of function evaluations. For this reason, different approaches to reducing the number of evaluations have been suggested. A posteriori methods for computationally expensive multiobjective optimization are important in this thesis, because our PAINT method (introduced in Paper [PIV] and discussed in Section 5.3) can use the output of an a posteriori algorithm and further approximate the Pareto front based on this.

The set-based structure of evolutionary multiobjective optimization methods for computationally expensive multiobjective optimization problems conveniently works in concert with a surrogate model, which can be validated and updated inside the algorithm. In [43], different uses of surrogate models in evolutionary multiobjective optimization are surveyed. For functional approximation, they mention different meta-models including Response Surface Method (see e.g., [65]), Radial Basis Functions and Support Vector Machines (applied e.g., in [66]). A similar, but a little more recent survey is also done in [78].

In deterministic a posteriori methods, a single solution is computed at a time (unlike in the evolutionary methods). Thus, one cannot use similar approaches for validating and updating the surrogate models of the objective functions. De-

terministic methods, however, compensate for this because, in general, they converge to at least locally Pareto optimal solutions with fewer function evaluations than evolutionary methods. Deterministic a posteriori methods using surrogates of the objective functions can be found e.g., in [52, 53, 54, 64, 76, 87].

The choice of a posteriori method using surrogates of the objective or constraint functions for computationally expensive multiobjective optimization problem is greatly problem dependent. That is to say, different methods work better for different problems. It is always beneficial to use some characteristics of the problem, if possible.

Similar approximation issues to those for a posteriori methods (described in [40]) apply also to interactive methods with computationally expensive problems. The approximation level of the objective functions can be any of the ones mentioned above and the problem with surrogate objective functions (approximated e.g., with simulations or meta-models) can be solved with any interactive method. Once the decision maker has found a preferred solution to the problem with surrogate objective functions one can either take that solution as the final solution or take the outcome given by the surrogate objective function as a reference point e.g., for the achievement scalarizing problem (4) and take the solution to that as the final solution. This kind of approaches have been followed e.g., in [68] (using meta-models) and [36] (using simulations).

Like the evolutionary a posteriori methods, also interactive methods could be used with surrogate approximations of the objective functions that could then be updated whenever necessary. This is because of the iterative nature of interactive methods, where, once an interesting solution for the problem using a surrogate approximation has been found, the solution could be evaluated using the actual computationally expensive objective functions. If found necessary, the surrogate approximation can be updated with this new solution after its values for the original objective functions are known. This has, however, (at least to the author's knowledge) been followed only in [68] possibly because evaluating the problem with the original computationally expensive objective functions and updating the surrogate approximations of the objective functions may be very time consuming and the decision maker has to wait through it.

Another direction is approximating the Pareto front. This approach can be seen as following the arguments of [42] that only the values (e.g., the objectives) should be meaningful in decision making – the decision variables should be decided only after a proper outcome has been decided. Thus, it is enough to approximate the outcome space in order to make an informed decision. Furthermore, since the Pareto optimal outcomes dominate all the other outcomes, it is enough to approximate the set of Pareto optimal outcomes i.e., the Pareto front. Once a preferred approximate Pareto optimal outcome has been found, the values of preferred decision variables can be found via e.g., the achievement scalarizing problem (4).

Some Pareto front approximation methods published before the year 2003 are surveyed in [75]. Pareto front approximations not mentioned in [75] are given in [18, 32, 41, 48, 49, 50, 73]. However, most of the approximation methods in the

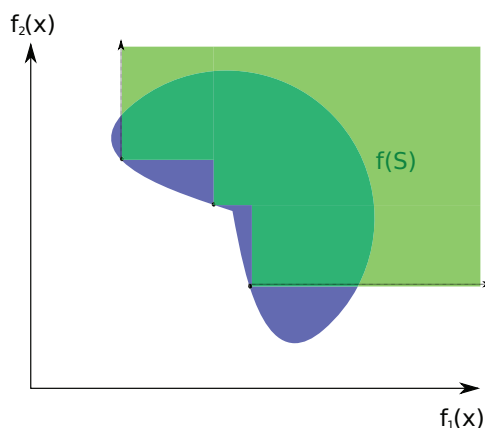


FIGURE 7 The Edgeworth Pareto hull approximation (in green) constructed in the non-convex case by the feasible goals method for three Pareto optimal outcomes (black dots)

literature are only applicable to convex multiobjective optimization problems.

Most of the approximation methods surveyed in [75], also, do not concentrate on the issue of how to make decisions about the original multiobjective optimization problem with the Pareto front approximation (nor how to use either an a posteriori or interactive method with the approximation). To our knowledge, only few papers consider decision making with the Pareto front approximation.

One of only few papers considering decision making with Pareto front approximations is [49]. In [49], an approach that resembles using an a posteriori method with a Pareto front approximation is proposed. In that paper, different approximations for convex and nonconvex problems are constructed and statistical methods are used to inspect the quality of the approximation. After the approximation is constructed, a preferred point is chosen with the help of a visualization technique called Interactive Decision Maps. In this thesis, we are mainly interested in techniques that approximate nonconvex Pareto fronts and, thus, we are going to discuss here only those. In addition, we are only going to discuss the feasible goals method for approximating the Edgeworth Pareto hull, because the reasonable goals method is based on the convex hull, which is also used by other methods discussed later. Technically speaking, the idea in the feasible goals method in nonconvex problems is to approximate the Edgeworth Pareto hull by the set  $\cup_{y \in Y} \{y + z : z \in \mathbb{R}_+^k\}$ . This set is displayed in Figure 7 for three Pareto optimal outcomes in the objective space.

In addition, methods resembling interactive methods have been developed to be used with a Pareto front approximation. In [25], the convex hull of a given set of Pareto optimal outcomes is constructed and a procedure that extends and generalizes the idea of the Pareto Race method for linear problems [44] is used to navigate on the approximation. In addition, in [62, 63], the convex hull of given Pareto optimal outcomes is used and an approach called Pareto Navigation for

exploring the Pareto front approximation is developed.

These multiobjective optimization methods with Pareto front approximations, however, have some drawbacks. Above, only the approximation method of [49] is applicable to nonconvex problems. In addition, decision making with the Pareto front approximation is done with methods that are tailor made for deciding on the constructed Pareto front approximation. Our goal is that the same methods should be used on the decision making with the Pareto front approximations, as one would prefer to apply to the original multiobjective optimization problem. None of the Pareto front approximation methods mentioned in this section are applicable to this with nonconvex multiobjective optimization problems.

The next chapter concludes the first part of this thesis. The purpose of the next chapter is to motivate our research and to give background for it. In the following chapter, we show how none of the Pareto front approximation methods given above apply to approximating the Pareto fronts on nonconvex multiobjective optimization problems in a way that would suit us. After this, we state our research objectives for our Pareto front approximation method PAIN.T.

## 4 TOWARDS THE PAINT METHOD

In this chapter, we motivate the PAINT method by giving a simple example that shows what we would want to be able to do and show how none of the methods available in the literature accomplish what we want. After this, we state our research objectives for the PAINT method.

### 4.1 Motivating the PAINT Method

In this section, we give a (simple) example that demonstrates our motivation for the PAINT method. In practice, the given set of Pareto optimal outcomes would be a lot larger, but the three points given in this section are enough to demonstrate our way of thinking.

Let us assume that we want to solve a computationally expensive, multiobjective optimization problem with continuous variables

$$\begin{aligned} \min & \quad (f_1(x), f_2(x), f_3(x)) \\ \text{subject to} & \quad x \in S. \end{aligned}$$

In addition, let us assume that the feasible set is  $S \subset \mathbb{R}^n$  for some rather large  $n$ , which is often the case for practical multiobjective optimization problems. Furthermore, we assume that we have used a lot of time and effort to produce three Pareto optimal solutions  $x^1, x^2, x^3 \in S$  to the problem. The three Pareto optimal outcomes given by these solutions are  $p^1 = (3, 0, 0)$ ,  $p^2 = (0, 3, 0)$  and  $p^3 = (2, 2, 10)$ , respectively, and these are shown in Figure 8. Unfortunately, this set of solutions does not include a preferable solution to the decision maker. Because we have already done a lot of computation to find these three solutions, we (as analysts) want to use them to better guide the search for further solutions. These Pareto optimal solutions are not yet enough to construct a surrogate approximation of the objective functions because  $n$  is large, and we must pursue other means. As we can ask the preferences of the decision maker, we decide to approximate the possible intermediate outcomes between these Pareto optimal

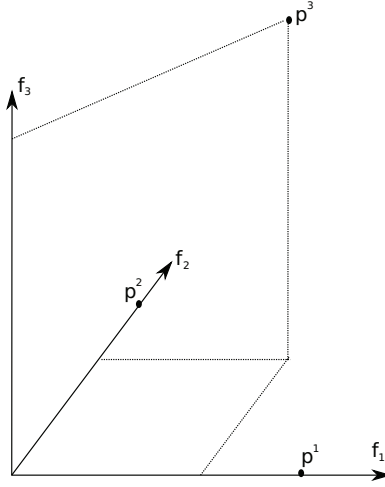


FIGURE 8 The three given Pareto optimal outcomes for our example problem

outcomes and then ask the decision maker to choose a preferred one of the approximate outcomes. This preferred approximate outcome can then be used as a reference point to guide the search for further solutions. In order to accomplish this, two questions have to be answered:

1. How to define and compute the intermediate outcomes?
2. How to find a preferred intermediate outcome once their set is known?

Starting with the first question, we can try to use Pareto front approximation methods surveyed in Section 3.2. We have decided to consider the convex hull of the outcomes (used e.g., in [25, 62, 63]), the Edgeworth Pareto Hull of the outcomes (used in the Feasible Goals method described in [49]) and the response surface methodology (of [32]). We decided to use these approaches, because we could not choose any of the sandwich approximations, as the problem of choosing a preferred approximate outcome in the larger set of the sandwich approximations would be a lot harder problem and we do not know how to do that. In addition, the Pareto front approximation had to be applicable to a given set of Pareto optimal outcomes. We will now study these approximations in detail.

Figure 9 displays the convex hull of the Pareto optimal outcomes. This approximation may seem rather feasible at first. However, one may notice at a closer inspection of the figure that the convex hull contains e.g., the vector  $z = (1.5, 1.5, 0)$ . The vector  $z$  dominates one of the given Pareto optimal outcomes  $p^3 = (2, 2, 10)$ , which implies by the definition of Pareto optimality that the vector  $z$  cannot be an outcome of the multiobjective optimization problem. Thus, the approximation is flawed and misleading. Having not noticed the contradiction between the approximation and the given Pareto optimal outcomes, one may have led the decision maker to believe that the outcome  $(1.5, 1.5, 0)$  is feasible, when in fact we know that it is not. This demonstrates that a convex hull

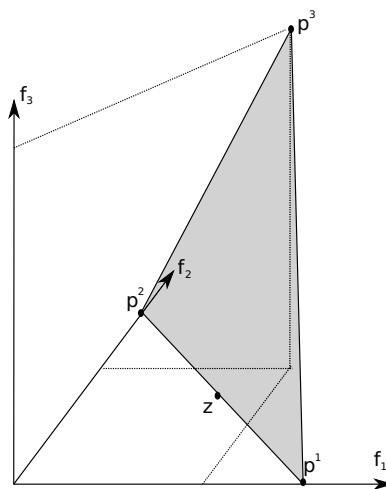


FIGURE 9 A convex hull of the given Pareto optimal outcomes

is not a good approximation for nonconvex problems, which is what this problem clearly is.

Figure 10 shows the approximation of the Edgeworth Pareto Hull produced by the Feasible Goals method implemented in (and visualized by) the Pareto Front Visualization software [49]. This approximation does not have the same drawback as the convex hull approximation and it has indeed been successfully used to approximate Pareto fronts of nonconvex multiobjective optimization problems. However, our criticism to this is that the only nondominated vectors on the approximation are the given Pareto optimal outcomes. When the decision maker wants to choose a vector that is not in the given set of Pareto optimal outcomes, he/she has to choose an area of the outcome space in which the outcomes are dominated according to the approximation. We believe that this leaves too heavy a cognitive load for the decision maker to determine that the approximation may be flawed and to trust that there may be outcomes that are better than what the approximation explicitly shows.

Finally, there is the response surface based methodology used in [32] where a polynomial approximation of the function that describes one of the objectives as a function of the others is formed and then the graph of the constructed function restricted to the convex hull of the outcomes is taken as the approximation. Unfortunately, this approach may imply the same phenomenon as the convex hull approximation: Assume that we are to describe the third objective as a function  $g : \mathbb{R}^2 \rightarrow \mathbb{R}$  of the first and the second objective. Now, by setting  $g(z_1, z_2) = 10(z_1 + z_2) - 30$ , it holds that  $g(0, 3) = g(3, 0) = 0$  and  $g(2, 2) = 10$ . Thus, the polynomial  $g$  describes the third objective as a function of the two others. By restricting the function  $g$  to the convex hull of the outcomes, the graph is exactly the convex hull shown in Figure 9. Thus, the same conclusions as in the convex hull case apply.

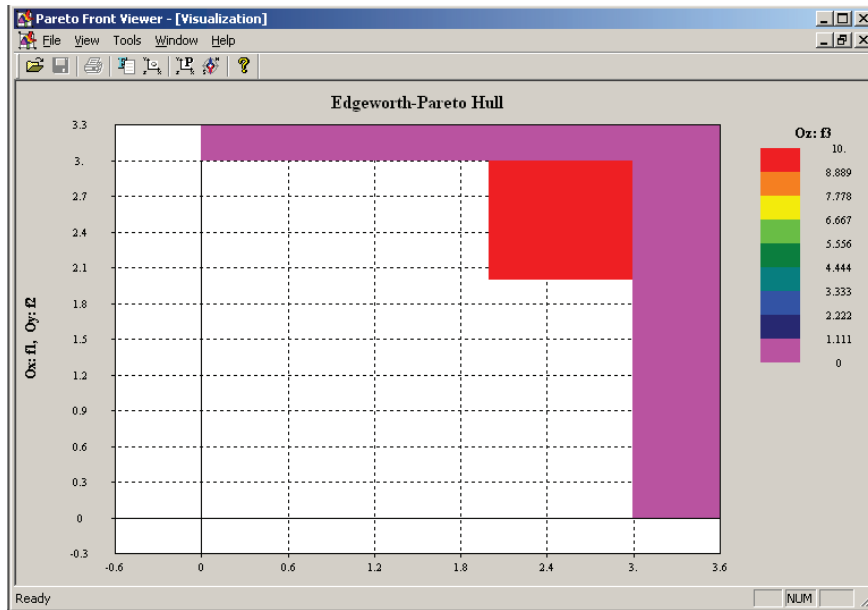


FIGURE 10 An Edgeworth Pareto hull approximation produced by the Pareto Front Visualization software (see [49]) using the feasible goals method

The above observations led us to consider desirable properties of a set that could be used to approximate the Pareto front based on a given set of Pareto optimal outcomes. A fundamental property of the Pareto front itself is that there are no two vectors on it so that one dominates the other. Thus, a potential Pareto front approximation would be a set with this property that also includes the given Pareto optimal outcomes. This led us to the definition of inherent nondominance. In the next chapter, this concept is defined in mathematical terms and the most important theorems given in Paper [PI] are reviewed. These theorems show that an inherently nondominated Pareto front approximation avoids the problems of the approximations shown above. The PAINT method (introduced in Paper [PIV] and described in Section 5.3 of this thesis) then constructs an inherently nondominated Pareto front approximation. The Pareto front approximation constructed with the PAINT method for the above problem is shown in Figure 11. We must emphasize that the Pareto front approximation constructed with PAINT is not always merely a union of line segments, but could contain higher dimensional objects, also. The fact that the Pareto front approximation in Figure 11 consists of line segments is because of the characteristics of our example problem and the locations of the given outcomes.

The question of choosing a preferred approximate solution has been neglected e.g., in many of the papers surveyed in [75]. If the approximation is a finite set, then one may use any method of multiattribute decision analysis e.g., Multi-Attribute Value Analysis (see e.g., [27]) or MACBETH (see [6]). When the set of intermediate solutions is infinite or even innumerable, (which is a natural



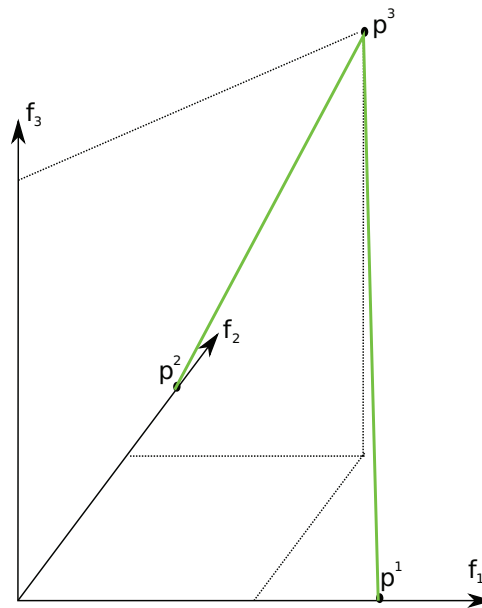


FIGURE 11 The Pareto front approximation constructed with the PAINT method (the green line segments)

structure for such a set, if the variables of the problem are continuous), it becomes much harder to choose a preferred approximate outcome.

Inherent nondominance can be used to derive a surrogate problem, which is computationally less expensive than the original one. This surrogate problem is rather simple, but because of the inherent nondominance, it has some interesting features. The most important of these features is that the Pareto front of the surrogate problem is exactly the inherently nondominated Pareto front approximation. The surrogate problem can be solved with any interactive multiobjective optimization method e.g., NIMBUS (discussed in Section 2.4.1). For further details on the surrogate problem, see Section 5.4. This solves the problem of finding a preferred outcome on the Pareto front approximation and, thus, we can use an inherently nondominated Pareto front approximation (displayed in Figure 11) to computationally inexpensively guide the search for further more preferred solutions to the computationally expensive multiobjective optimization problem.

## 4.2 Research Objectives

In this section, we give our research objectives that were motivated by the example in the previous section. These were the following:

1. The Pareto front approximation developed should be applicable to approximating Pareto fronts of nonconvex problems based on any given (finite) set

of Pareto optimal outcomes.

2. Motivated by the Pareto Navigator method (introduced in [25]), the Pareto front approximation should be applicable with interactive methods to solve computationally expensive multiobjective optimization problems.
3. The Pareto front approximation method should be easily implementable.
4. The usefulness of the Pareto front approximation in solving computationally expensive multiobjective optimization problems should be established with examples.

The applicability of the Pareto front approximation to nonconvex problems is the main motivation behind our method, because we wanted to generalize and extend the Pareto Navigator method. In addition, the method should not have any requirements for the given set of Pareto optimal outcomes that are the starting point of the approximation. As the Pareto front approximation does not have any requirements, any a posteriori method can be used to generate this set. The PAINT method satisfies these requirements, as we demonstrate later in this thesis.

The use of the Pareto front approximation with an interactive method to solve computationally expensive problems has a couple of issues that need to be solved. First, as interactive methods can be used to solve computationally expensive multiobjective optimization problems (not find preferable approximate outcomes in sets), the Pareto front approximation should imply a surrogate problem for the original problem. Second, the surrogate problem should be computationally expensive so that it is worthwhile to solve the surrogate instead of the original problem. The surrogate problem implied by an inherently nondominated Pareto front approximation constructed with the PAINT method satisfies these requirements.

In real-life examples, the numbers of given Pareto optimal outcomes and objectives may be quite large. Thus, the PAINT method must be easily implementable on a computer. For that reason, we also consider numerical and algorithmic issues related to the PAINT method.

Finally, the usefulness of the PAINT method should be established with examples. In this thesis, we demonstrate how our approach applies to two examples. These examples are a three objective heat exchanger network synthesis problem and a five objective problem of designing and operating a wastewater treatment plant.

## 5 THE PAINT METHOD

This chapter starts the second part of this thesis. In this part, we present a mathematical consideration of the results leading to the PAINT method and a mathematical introduction to the PAINT method.

Since this chapter starts a new more mathematical treatment of the topics, we need some more definitions. These are given in the following section.

### 5.1 Prerequisites for the PAINT Method

In addition to the definitions given in Chapter 2.1, we need more definitions for introducing the PAINT method. These are given in this section.

First, we need some definitions from computational geometry (see e.g., [23, 33]). An  $a$ -polytope in  $\mathbb{R}^k$  is the convex hull of  $a + 1$  points in  $\mathbb{R}^k$ . An  $a$ -polytope is said to be defined by points  $z^1, \dots, z^{a+1} \in \mathbb{R}^k$ , if it is the convex hull of those points, and then it is denoted by  $\mathcal{P}(z^1, \dots, z^{a+1})$ . A vertex of a polytope  $K$  is a point  $x \in K$  for which it holds that  $\lambda z + (1 - \lambda)y = x$ ,  $\lambda \in (0, 1)$  and  $z, y \in K$  imply  $x = y = z$ . The set of vertices of a polytope  $K$  is denoted by  $\text{vert}(K)$ . A face of a polytope  $K$  is another polytope  $K'$  so that  $K' = \emptyset$ ,  $K' = K$  or there exist vectors  $z^1, z^2 \in \mathbb{R}^k$  so that  $K' = \{z^1 + h : h \in \mathbb{R}^k, h^T z^2 = 0\} \cap K$ . A polytope  $\mathcal{P}(z^1, \dots, z^{a+1})$  is called a simplex if the vectors  $z^1, \dots, z^{a+1}$  are affinely independent. Figure 12 shows a 3-polytope, its face and its vertex.

Collections of polytopes are sets whose elements are polytopes and they are denoted by calligraphic letters  $\mathcal{A}, \mathcal{B}, \mathcal{K}, \dots$ . The body of a collection of polytopes  $\mathcal{A}$  is  $\text{body}(\mathcal{A}) = \cup_{K \in \mathcal{A}} K$ . A (polyhedral) complex  $\mathcal{K}$  is a special type of collection of polytopes so that (a) if it holds that a polytope  $K \in \mathcal{K}$  and another polytope  $K'$  is a face of  $K$  then it must hold that  $K' \in \mathcal{K}$ , and (b) if it holds that polytopes  $K^1, K^2 \in \mathcal{K}$  then the set  $K^1 \cap K^2$  must be a possibly empty face of both polytopes  $K^1$  and  $K^2$ . A triangulation of a finite set  $P \subset \mathbb{R}^k$  is a complex  $\mathcal{T}$  so that the body of  $\mathcal{T}$  is the convex hull of the set  $P$  and the set of vertices in  $\mathcal{T}$  is  $P$ . A polytope  $\mathcal{P}(z^1, \dots, z^k)$  with  $z^1, \dots, z^k \in P$  is called Delaunay (in  $P$ ) if there exists

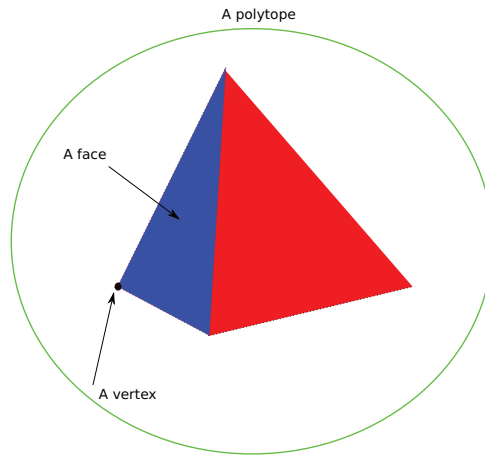


FIGURE 12 A 3-polytope, its face and its vertex

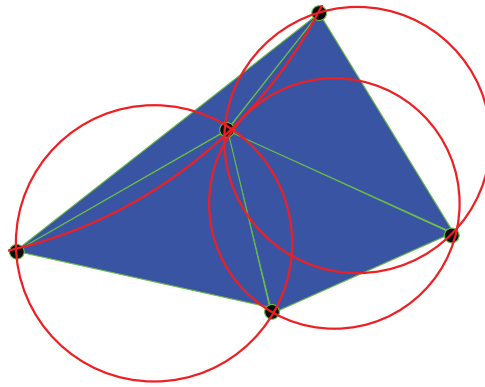


FIGURE 13 A Delaunay triangulation of 5 points in  $\mathbb{R}^2$  with the circumscribing balls for the 2-polytopes outlined in red

a circumscribing ball  $B \in \mathbb{R}^k$  with  $\text{cl}(B) \cap P = \{z^1, \dots, z^{a+1}\}$  (here  $\text{cl}(B)$  denotes the closure of  $B$ ) and  $B \cap P = \emptyset$ . A complex  $\mathcal{D}$  is a Delaunay triangulation of  $P$  if  $\mathcal{D}$  is a triangulation of  $P$  so that every polytope in  $\mathcal{D}$  is Delaunay. A polyhedral complex is called a simplicial complex, if all the polytopes are simplices. Figure 13 shows a Delaunay triangulation of 5 points in  $\mathbb{R}^2$ . The circumscribing balls are drawn for the 2-polytopes in red.

A requirement for a representation  $p^1, \dots, p^a \in PF$  of the Pareto front (given e.g., in [90]) is that the points are mutually nondominated, i.e., there does not exist  $i, j \in \{1, \dots, a\}$  such that  $p^i$  dominates  $p^j$ . The inherent nondominance property of a set (defined later in this chapter) then generalizes this definition.

In the PAINT method, we need to consider relations between sets with respect to Pareto dominance and, thus, we define (Pareto) dominance between two sets in the objective space (in addition to dominance between two vectors in the

objective space). A set  $A^1 \subset \mathbb{R}^k$  is said to dominate another set  $A^2 \subset \mathbb{R}^k$ , if there exist vectors  $z^1 \in A^1$  and  $z^2 \in A^2$  so that the vector  $z^1$  dominates the vector  $z^2$ . A point  $z$  is said to dominate or be dominated by a set, if the singleton  $\{z\}$  dominates or is dominated by the set.

Some matrix notations are used in introducing the PAINT algorithm. The entry of a matrix  $A \in \mathbb{R}^{a \times b}$  on the row  $i$  and the column  $j$  is denoted by  $A_{i,j}$ . For two matrices,  $A \in \mathbb{R}^{a,b}$  and  $B \in \mathbb{R}^{a,c}$ , the notation  $C = [A, B]$  denotes a matrix

$$C = \begin{bmatrix} A_{1,1} & \dots & A_{1,a} & B_{1,1} & \dots & B_{1,c} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ A_{b,1} & \dots & A_{b,a} & B_{b,1} & \dots & B_{b,c} \end{bmatrix}.$$

## 5.2 Inherent Nondominance

This section summarizes the definitions and theorems related to inherent nondominance given in [PI]. A fundamental concept in this research is inherent nondominance.

**Definition 1.** A set  $A \subset \mathbb{R}^k$  is inherently nondominated, if there does not exist vectors  $a, b \in A$  such that  $a \leq b$ .

An equivalent formulation for Definition 1 would be that all the finite subsets of  $A$  are mutually nondominated. Thus, inherent nondominance can be seen as a generalization of mutual nondominance.

The following definition attaches an inherently nondominated set to a known set of Pareto optimal outcomes  $P$ . It says that an inherently nondominated set is called a Pareto front approximation if it contains the given set of Pareto optimal outcomes.

**Definition 2.** Let  $P \subset \mathbb{R}^k$  be a finite set of Pareto optimal outcomes. An inherently nondominated set  $A$  is called an inherently nondominated Pareto front approximation based on the set  $P$ , if  $P \subset A$ .

An inherently nondominated set is connected in a natural way to a multiobjective optimization problem. This problem is called the multiobjective optimization problem implied by the approximation.

**Definition 3.** Let  $A$  be an inherently nondominated Pareto front approximation. Then the multiobjective optimization problem

$$\begin{array}{ll} \min & (z_1, \dots, z_k) \\ \text{subject to} & (z_1, \dots, z_k) \in A \end{array}$$

is called the multiobjective optimization problem implied by the approximation  $A$ .

Results of the most important theorems concerning inherent nondominance are summarized below. Their interpretations are given in more detail in Paper [PI], which also includes their proofs.

1. The Pareto front  $PF$  of a multiobjective optimization problem is an inherently nondominated Pareto front approximation based on  $P$  for any finite subset  $P \subset PF$ .
2. Let  $A \subset \mathbb{R}^k$  be a set. Then the set  $A$  is inherently nondominated if and only if  $PF(A) = A$ .
3. If a set  $A \subset \mathbb{R}^k$  is an inherently nondominated Pareto front approximation based on  $P$ , then

$$P \subset PF(A).$$

4. The given set of Pareto optimal outcomes  $P$  is an inherently nondominated Pareto front approximation based on  $P$ .

An interpretation of result 1. above is that an inherently nondominated Pareto front approximation is well defined in a way that the Pareto front itself satisfies this property. It would be undesirable if this would not be the case. Result 2. gives an equivalent definition for inherent nondominance and result 3. is a corollary of result 2. that shows that the given set of Pareto optimal outcomes  $P$  is in the set of Pareto optimal vectors in the inherently nondominated Pareto front approximation  $A$ . Finally, result 4. proves the existence of inherently nondominated Pareto front approximations. Because of the simplicity of such an approximation, the given set of Pareto optimal outcomes  $P$  is called the trivial inherently nondominated Pareto front approximation.

Now, we have demonstrated the desirable properties of an inherently nondominated Pareto front approximation. The following section includes a mathematical idea of constructing an inherently nondominated Pareto front approximation based on any given set of Pareto optimal outcomes.

### 5.2.1 An Inherently Nondominated Subcomplex of the Delaunay Triangulation

In Paper [PII], it was suggested that similar ideas that are used in computational geometry (see e.g., [23]) should be used to construct an inherently nondominated Pareto front approximation from a given set of Pareto optimal outcomes. These ideas were to use the Delaunay triangulation of the given outcomes as a basis and then remove inappropriate polytopes from it according to some rules that are based on inherent nondominance. This idea is similar to that of [4] in approximating a general nonconvex set, where distance is used as a rule for removing polytopes from the Delaunay triangulation. As our requirement for the Pareto front approximation is inherent nondominance, distance cannot be used in our case.

The rules (given in [PII]) that govern the removal of polytopes are the following. From now on, let  $P$  be the set of  $m$  given Pareto optimal outcomes. The Pareto front approximation is constructed based on this set. Assuming that for all  $a = 0, \dots, m - 1$ , the  $a$ -polytopes in the complex  $\mathcal{D}$  are  $K^{a,1}, \dots, K^{a,t^a}$ . For all  $a = m - 1, m - 2, \dots, 1$  and for all  $j = 1, \dots, t^a$ , it holds that the polytope  $K^{a,j}$  is removed, if either

- (R1) there exists a Pareto optimal outcome  $p \in P$  that dominates or is dominated by the polytope  $K^{a,j}$  or the polytope  $K^{a,j}$  dominates itself  
OR  
(R2) there exists a  $b$ -polytope  $K^{b,j'} \in \mathcal{D}$  with either  $b > a$  or  $b = a$  and  $j' < j$  that is not removed and that dominates, or is dominated by, the polytope  $K^{a,j}$ .

The result of applying the rules to a Delaunay triangulation is a set of polytopes, which forms an inherently nondominated Pareto front approximation.

As the above rules include rather complicated rules and the Delaunay triangulation may contain a large number of polytopes, it is not clear whether they can be implemented efficiently on a computer. The PAINT method, introduced in the following section and in a greater detail in Paper [PIV], constructs an inherently nondominated Pareto front approximation that follows rules (R1) and (R2).

### 5.3 The PAINT Method

In this section, we introduce the PAINT method that implements rules (R1) and (R2). The purpose of this section is to give an idea of the method and Paper [PIV] contains a more rigorous treatment of the method.

#### 5.3.1 The Algorithm

In Paper [PIV], the PAINT algorithm is introduced and, in this introduction, it is given in Algorithm 1. In the algorithm, the Delaunay triangulation and the inherently nondominated Pareto front approximation are represented as matrices  $D$  and  $A$ , respectively. In those matrices, each row represents the vertices of a polytope with each entry referring to an outcome in  $P$ . For example, a row with entries 1, 2 and 3 represents a polytope defined by outcomes  $p^1, p^2$  and  $p^3$  in  $P$ . Polytopes defined with fewer outcomes than there are columns in  $D$  or  $A$  are handled by repeating the same outcome multiple times. Furthermore, indices  $a$  and  $b$  stand for the number of polytopes in the Delaunay triangulation and the maximal number of outcomes defining a polytope in the Delaunay triangulation, respectively.

The input of the algorithm is the given set of Pareto optimal outcomes  $P$ . In line 2 of the algorithm, a Delaunay triangulation of the set  $P$  is constructed and represented as a matrix  $D$ . In line 3 of Algorithm 1, the rows of matrix  $D$  are ordered in a descending order with respect to the number of outcomes defining

---

**Algorithm 1** PAINT method: Construction of the inherently nondominated Pareto front approximation

---

```

1: Read the Pareto optimal outcomes  $P = \{p^1, p^2, \dots, p^m\} \subset \mathbb{R}^k$ .
2:  $D \leftarrow$  the Delaunay triangulation of  $P$ 
3: Sort the rows of  $D$  in descending order w.r.t. number of different entries in
   the row
4:  $a, b \leftarrow$  the number of rows and columns of  $D$ , respectively,  $d \leftarrow 0$ 
5: for  $i = 1$  to  $a$  do
6:   if the polytope given by vertices  $p^{D_{i,1}}, p^{D_{i,2}}, \dots, p^{D_{i,b}}$  is not inherently non-
     dominated then
7:      $d \leftarrow d + 1$  and interchange rows  $i$  and  $d$  of the matrix  $D$ 
8:   else
9:      $deleted \leftarrow \text{false}$ 
10:    for  $j = 1$  to  $m$  do
11:      if the polytope given by vertices  $p^{D_{i,1}}, p^{D_{i,2}}, \dots, p^{D_{i,b}}$  dominates or is
        dominated by outcome  $p^j$  then
12:         $d \leftarrow d + 1$  and interchange rows  $i$  and  $d$  of the matrix  $D$ 
13:         $deleted \leftarrow \text{true}$ 
14:        Break
15:      end if
16:    end for
17:    if not  $deleted$  then
18:      for  $l = d + 1$  to  $i - 1$  do
19:        if the polytope given by vertices  $p^{D_{i,1}}, p^{D_{i,2}}, \dots, p^{D_{i,b}}$  dominates or
          is dominated by the polytope given by vertices  $p^{D_{l,1}}, p^{D_{l,2}}, \dots, p^{D_{l,b}}$ 
          then
20:           $d \leftarrow d + 1$  and interchange rows  $i$  and  $d$  of the matrix  $D$ 
21:          Break
22:        end if
23:      end for
24:    end if
25:  end if
26: end for
27:  $A \leftarrow \begin{bmatrix} D_{d+1,1} & \dots & D_{d+1,b} \\ \vdots & \vdots & \vdots \\ D_{a,1} & \dots & D_{a,b} \end{bmatrix}$ 
28: return  $A$ 

```

---



the polytope that is represented by the row. In lines 6, 11 and 19 of the algorithm, there are three different if-conditions. The purpose of the conditions in lines 6 and 11 is to make sure that the final approximation follows rule (R1), and the purpose of the condition in line 19 is to make sure that the final approximation follows rule (R2).

All the if-conditions (in lines 6, 11 and 19) reduce to checking dominance between two polytopes, which is the condition in line 19: (a) inherent nondominance of a polytope in line 6 can be reduced to checking whether a polytope dominates itself, because in Paper [PIII] it was proven that a polytope is inherently nondominated if and only if it does not dominate itself, and (b) dominance between an outcome and a polytope in line 11 is actually dominance between two polytopes because a singleton containing a vector in  $\mathbb{R}^k$  is by definition a polytope. Thus, it is adequate to build mathematical tools for determining dominance between two polytopes.

Let a matrix

$$B = \begin{bmatrix} p_1^{D_{r_1,1}} & p_1^{D_{r_1,2}} & \dots & p_1^{D_{r_1,b}} & -p_1^{D_{r_2,1}} & \dots & -p_1^{D_{r_2,b}} \\ p_2^{D_{r_1,1}} & p_2^{D_{r_1,2}} & \dots & p_2^{D_{r_1,b}} & -p_2^{D_{r_2,1}} & \dots & -p_2^{D_{r_2,b}} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ p_k^{D_{r_1,1}} & p_k^{D_{r_1,2}} & \dots & p_k^{D_{r_1,b}} & -p_k^{D_{r_2,1}} & \dots & -p_k^{D_{r_2,b}} \end{bmatrix}.$$

In Paper [PIII], it was proven that if a polytope  $K^1$  is given by row  $r_1$  in the matrix  $D$  (representing the Delaunay triangulation) and another polytope  $K^2$  is given by row  $r_2$  in  $D$ , then the polytope  $K^1$  dominates the polytope  $K^2$  if and only if either the optimal value of problem

$$\begin{aligned} \min \quad & t \\ \text{subject to} \quad & [B, -\mathbf{1}] \begin{bmatrix} \lambda \\ \mu \\ t \end{bmatrix} \leq \begin{bmatrix} 0 \\ \vdots \\ 0 \end{bmatrix} \\ & \sum_{j=1}^b \lambda_j = 1, \sum_{j=1}^b \mu_j = 1, \end{aligned} \tag{8}$$

where  $\lambda \in [0, 1]^b$ ,  $\mu \in [0, 1]^b$ ,  $t \in \mathbb{R}$ ,  $-\mathbf{1} = \begin{bmatrix} -1 \\ \vdots \\ -1 \end{bmatrix}$

is less than zero or the optimal value of problem (8) is zero and the optimal value

of problem

$$\begin{aligned}
& \min && g \begin{bmatrix} \lambda \\ \mu \end{bmatrix} \\
& \text{subject to} && B \begin{bmatrix} \lambda \\ \mu \end{bmatrix} \leq \begin{bmatrix} 0 \\ \vdots \\ 0 \end{bmatrix} \\
& \text{where} && \sum_{j=1}^b \lambda_j = 1, \sum_{j=1}^b \mu_j = 1, \\
& && g = [p_1^{D_{r_1,1}} + \dots + p_k^{D_{r_1,1}}, \dots, p_1^{D_{r_1,b}} + \dots + p_k^{D_{r_1,b}} \\
& && \quad - p_1^{D_{r_2,1}} - \dots - p_k^{D_{r_2,1}}, \dots, -p_1^{D_{r_2,b}} - \dots - p_k^{D_{r_2,b}}] \\
& && \lambda \in [0, 1]^b, \mu \in [0, 1]^b
\end{aligned} \tag{9}$$

is less than zero. Consequently, all the if-conditions in lines 6, 11 and 19 can be checked by solving linear optimization problems (8) and (9).

This concludes discussion of the PAINT algorithm. Paper [PIV] contains a more thorough treatment of the topics. In the next section, we demonstrate how PAINT method applies to a simple multiobjective optimization problem.

### 5.3.2 PAINT with the Four-Objective DTLZ2 Test Problem

In this section, we give an example of Pareto front approximation with the PAINT method and describe how we have implemented each step of Algorithm 1. The multiobjective optimization problem that we are going to investigate is the four-objective DTLZ2 test problem with  $n$  variables, introduced in [21]. The problem can be formulated as

$$\begin{aligned}
& \min && (f_1(x), f_2(x), f_3(x), f_4(x)) \\
& \text{subject to} && 0 \leq x_i \leq 1 \text{ for all } i = 1, \dots, n \\
& \text{where} && g(x) = \sum_{i=4}^n (x_i - 0.5)^2 \\
& && f_1(x) = (1 + g(x)) \cos(x_1\pi/2) \cos(x_2\pi/2) \cos(x_3\pi/2) \\
& && f_2(x) = (1 + g(x)) \cos(x_1\pi/2) \cos(x_2\pi/2) \sin(x_3\pi/2) \\
& && f_3(x) = (1 + g(x)) \cos(x_1\pi/2) \sin(x_2\pi/2) \\
& && f_4(x) = (1 + g(x)) \sin(x_1\pi/2).
\end{aligned}$$

The Pareto front of the four-objective DTLZ2 test problem consists of a subset of the unit sphere that is in the positive orthant of  $\mathbb{R}^4$ . It is easy to see that the Pareto optimal solutions  $x^*$  to the DTLZ2 problem have  $x_i^* = 0.5$  for all  $i = 4, \dots, n$ . The DTLZ2 problem is not computationally expensive, but we can use it here to demonstrate the PAINT method's usefulness and operation in approximating Pareto fronts of nonconvex four-objective optimization problems.

Because the DTLZ2-problem is mathematically very simple, one can find a set of Pareto optimal solutions to it by setting the components from four to  $n$  to 0.5 and randomly generating values between zero and one for the three first components. By using this method, we generated a set of 20 Pareto optimal solutions to this problem. The objective function values associated to these solutions are given in Table 1. These were taken as the given set of Pareto optimal solutions  $P$ , which was inputted into the PAINT algorithm.

TABLE 1 The given set of Pareto optimal outcomes  $P$  for the four-objective DTLZ2 test problem

Outcome	$f_1$	$f_2$	$f_3$	$f_4$
$p^1$	0.1712	0.7377	0.6467	0.0909
$p^2$	0.4399	0.2015	0.4218	0.7668
$p^3$	0.2457	0.1934	0.6819	0.6613
$p^4$	0.7375	0.3556	0.2103	0.5343
$p^5$	0.2430	0.6618	0.6582	0.2641
$p^6$	0.4781	0.6984	0.0493	0.5304
$p^7$	0.0952	0.4074	0.5114	0.7506
$p^8$	0.5689	0.3861	0.7032	0.1810
$p^9$	0.1823	0.0621	0.7022	0.6854
$p^{10}$	0.3304	0.5462	0.5204	0.5672
$p^{11}$	0.3074	0.7199	0.2407	0.5738
$p^{12}$	0.5578	0.3268	0.3376	0.6842
$p^{13}$	0.3685	0.5472	0.4703	0.5861
$p^{14}$	0.3598	0.4594	0.6085	0.5378
$p^{15}$	0.5401	0.1739	0.6530	0.5016
$p^{16}$	0.0395	0.2779	0.7468	0.6030
$p^{17}$	0.7778	0.4705	0.4110	0.0682
$p^{18}$	0.5801	0.6938	0.1218	0.4089
$p^{19}$	0.6200	0.2287	0.4903	0.5682
$p^{20}$	0.5681	0.3916	0.6646	0.2868

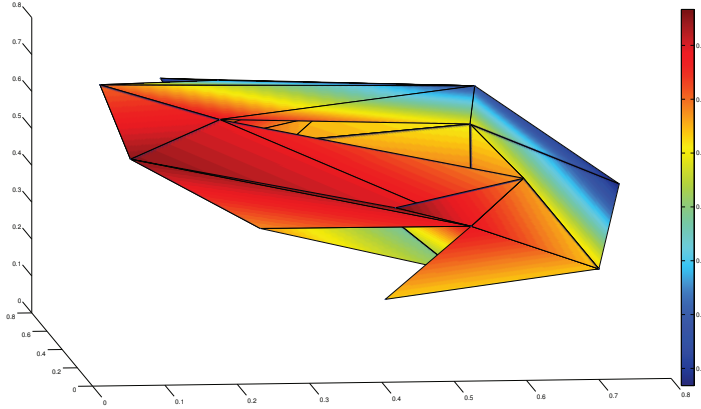


FIGURE 14 A projection of an inherently nondominated Pareto front approximation for the 4-objective DTLZ2 test problem. The values of the fourth objective are marked with color.

On line 2 of Algorithm 1, the structure of the Delaunay triangulation of  $P$  is put into a matrix  $D$ . The Delaunay triangulation is constructed with the Qhull implementation of the Quickhull algorithm [8]. The Qhull software outputs the full-dimensional (in this case the four-dimensional) polytopes of the Delaunay triangulation. The lower dimensional faces of these polytopes are then constructed by a simple Octave (see [22]) code. If the Pareto optimal outcomes  $P$  are in general position (as defined e.g., in [23]), the faces are the polytopes defined by the subsets of the set of vertices of the full-dimensional polytopes. As described in [24], the general position of points can be achieved by perturbing them slightly. This idea is used in the Qhull software, also.

After constructing the Delaunay triangulation (containing 368 polytopes) as the matrix  $D$ , we started removing inappropriate polytopes from it on lines five to 25 of the algorithm. In this example, 38.8% of the polytopes were removed due to rule (R1): 19.8% of the polytopes were removed on line 6 because they were not inherently nondominated and 19.0% of the polytopes were removed on line 11, because they were inherently nondominated, but dominated or were dominated by an outcome in  $P$ . Furthermore, 34.2% of the polytopes that were not removed due to rule (R1) were removed on line 19 due to rule (R2), because they dominated, or were dominated by, a higher dimensional polytope. Running the PAINT method took approximately 1.5 minutes, in this case.

The inherently nondominated Pareto front approximation given by PAINT contained 148 polytopes. The body of the inherently nondominated Pareto front approximation can be seen in Figure 14. In the figure, the approximation has been projected to  $\mathbb{R}^3$  by the projection function  $p : \mathbb{R}^4 \rightarrow \mathbb{R}^3, (x_1, x_2, x_3, x_4) \mapsto (x_1, x_2, x_3)$  and the fourth objective is marked with color.

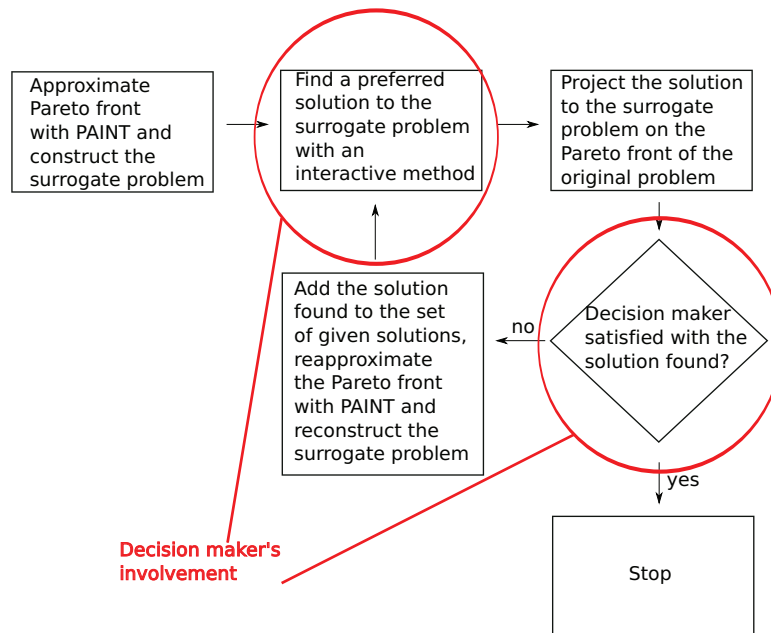


FIGURE 15 A flowchart of the decision making process with PAINT based on a given set of Pareto optimal outcomes

## 5.4 Decision Making and the Surrogate Problem Implied by the Approximation

As it was motivated in Section 4.1, once the Pareto front approximation is constructed the decision maker must be supported in finding a preferred vector on the approximation (i.e., preferred approximate Pareto optimal outcome). With the PAINT method, this is done through the mixed integer linear surrogate problem introduced in Paper [PIV]. Another issue is that, once a preferred approximate Pareto optimal outcome has been found, one must find the closest Pareto optimal solution to the original problem. In this section, we discuss both of these issues.

Decision making with PAINT is described in Figure 15. First, a Pareto front approximation is constructed with the PAINT algorithm 1 based on a given set of Pareto optimal outcomes. As described in Section 5.3, this yields a collection of polytopes that together form a Pareto front approximation.

Once the Pareto front approximation has been constructed, the surrogate problem implied by the approximation is formulated. The Pareto front of the surrogate problem is exactly the Pareto front approximation and, thus, a preferred solution to the surrogate problem yields a preferred approximate outcome on the Pareto front approximation. In Paper [PIV], it was shown that, when the Pareto front approximation is given by the PAINT method as a collection of poly-

topes  $\mathcal{A} = \{C_1, \dots, C_a\}$  with polytopes  $C_j = \mathcal{P}(p^{A_{j,1}}, \dots, p^{A_{j,b}})$ , where the matrix  $A \in \mathbb{R}^{a \times b}$  is constructed with the PAINT method and the given Pareto optimal outcomes are  $p^1, \dots, p^m$ , then the surrogate problem can be formulated as a multiobjective mixed integer linear problem

$$\begin{aligned}
& \min && (z_1, \dots, z_k) \\
& \text{subject to} && \sum_{j=1}^a \sum_{l=1}^b \lambda_{j,l} = 1 \\
& && \sum_{l=1}^b \lambda_{j,l} \leq y_j, \text{ for all } j = 1, \dots, a \\
& && \sum_{j=1}^a y_j = 1 \\
& \text{where} && \lambda \in [0, 1]^{a \times b} \\
& && y \in \{0, 1\}^a \\
& && z_i = \sum_{j=1}^a \sum_{l=1}^b \lambda_{j,l} p_i^{A_{l,j}} \text{ for all } i = 1, \dots, k.
\end{aligned} \tag{10}$$

Once the surrogate problem has been formulated, the decision maker gets involved. Benefit of problem (10) is that it can be used with low computational cost with interactive methods like NIMBUS, whose scalarizations preserve linearity. When this is the case, the scalarizations are also mixed integer linear problems that can be solved efficiently with e.g., CPLEX (see [39]) or GLPK (see [31]).

Once the computationally inexpensive mixed integer linear surrogate problem (10) has been solved, an actual Pareto optimal solution (to the original, computationally expensive multiobjective optimization problem) corresponding to it must be found. With the PAINT method, this is done with the achievement scalarizing problem (4). The preferred Pareto optimal outcome to the surrogate problem is taken as the reference point and the scalarization of the original problem is solved. Depending on the original problem, this part can be computationally expensive, but fortunately, no input is needed from the decision maker and, thus, he/she can concentrate on other tasks while the analyst finds an optimal solution to the achievement scalarizing problem. Because of the properties of the achievement scalarizing problem (4), this solution is Pareto optimal to the original problem and satisfies the aspirations of the decision described by the reference point (as discussed in [84]).

If the decision maker is satisfied with the Pareto optimal solution found in the previous step, we can naturally stop, since a preferred solution to the computationally expensive multiobjective optimization problem has been found. On the other hand, if the decision maker is not satisfied with the actual Pareto optimal solution, but he/she was satisfied with the approximate Pareto optimal solution (i.e., the solution to the surrogate problem), we find that our Pareto front approximation was not accurate enough near the Pareto optimal solution found. This flaw can be corrected by adding the Pareto optimal solution (to the original problem) to the given set of Pareto optimal outcomes and recomputing the Pareto front approximation with PAINT. This yields a Pareto front approximation that is more accurate near the Pareto optimal outcome found with the achievement scalarizing problem, because that outcome is included in the new approximation.

Once the new surrogate problem has been formulated, an interactive method can be used again to find a preferred solution to the new surrogate problem. This

iteration can be repeated as long as the decision maker is satisfied with the Pareto optimal solution found. It must, however, be noted that solving the achievement scalarizing problem (formulated for the original computationally expensive multiobjective problem) and recomputing the Pareto front approximation may be time consuming, and, thus, they must be avoided if possible. Naturally, to avoid spending time in solving the achievement scalarizing problem, one can first find the point that minimizes the achievement scalarizing problem in the given set of Pareto optimal outcomes (i.e., solves problem (4) with the feasible set  $S = P$ , the objective function  $f$  as the identity mapping and the reference point as the preferred approximate outcome) and then judge whether solving the computationally expensive problem is actually needed or if this point is already preferred by the decision maker. The above does not take a long time, because it only involves computing the value of a simple function for points in a relatively small set  $P$  and then selecting the smallest one of those values.

The ideas described in this section are implemented in a new IND-NIMBUS<sup>®</sup> PAINT module. The PAINT module (described in Paper [PV]) makes possible the use of an approximation (constructed with the PAINT method) with the NIMBUS method using a single graphical interface that is similar to the graphical interface of the NIMBUS method, implemented in the the IND-NIMBUS<sup>®</sup> software framework. The main screen of the IND-NIMBUS<sup>®</sup> PAINT module is shown in Figure 16.

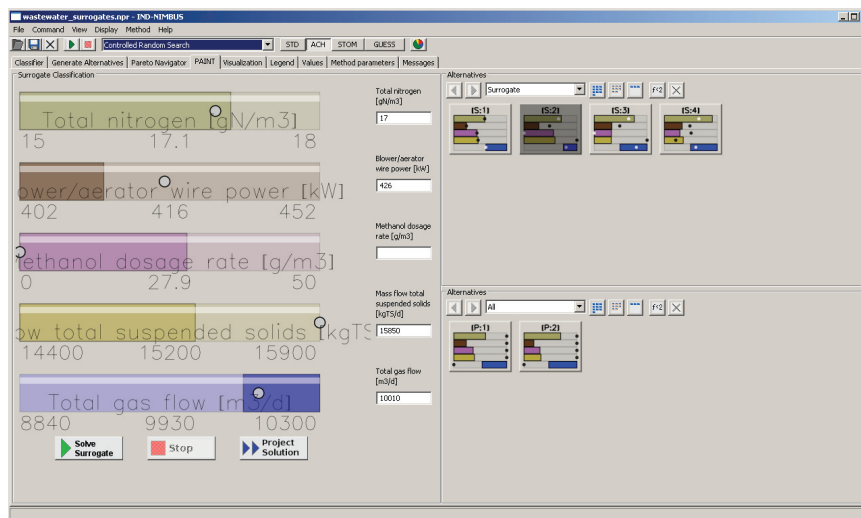


FIGURE 16 The IND-NIMBUS<sup>®</sup> PAINT module

With the PAINT module, the decision maker can give his/her preferences concerning the surrogate problem and, thus, find a preferred solution to the surrogate problem. The preferred solution to the surrogate problem can also be projected on the Pareto front of the original problem within the same interface (using the Project Solution button near the bottom of the screen). The approxi-

mate Pareto optimal solutions and the actual Pareto optimal solutions that have been found are visualized on the right of the screen. Further description of the PAIN module can be found in Paper [PV], which also includes an application of it to solving a computationally expensive multiobjective optimization problem of designing and operating wastewater treatment plant.



## 6 APPLICATIONS

In this Chapter, we describe how the applicability of the PAINT method has already been studied with two multiobjective optimization problems. The first problem was studied in Paper [PIII] and it was a multiobjective heat exchanger network synthesis problem. This problem had been studied also in [47]. The purpose of treating this problem in Paper [PIII] was to act merely as a proof of concept of decision making on an inherently nondominated Pareto front approximation. The number of given Pareto optimal outcomes used for approximating the Pareto front was considerably smaller than it would be in a real application. The number of given outcomes was small to reduce the complexity of the approximation. The PAINT method had not been developed, when Paper [PIII] was being written, and the polytopes had to be removed manually using directly the rules given in [PII].

The second example was studied in Paper [PV] and it was a multiobjective optimization problem of designing and operating a wastewater treatment plant. This problem had been previously studied in [77]. In Paper [PV], the PAINT method is applied to a real life problem for the first time. In addition, our treatment of this problem was in Paper [PV] much more rigorous than that of the heat exchanger problem in Paper [PIII]. In the wastewater treatment problem, we had a real decision maker Mr. Kristian Sahlstedt from Pöyry Environment Ltd who was willing to invest his time in studying the problem with our method. In the wastewater treatment problem, the new IND-NIMBUS<sup>®</sup> PAINT module (briefly discussed earlier in Section 5.4) was also used. With the PAINT method and the interactive NIMBUS method, we were able to find good compromise solutions to the wastewater treatment problem with low computational cost.

### 6.1 Heat Exchanger Network Synthesis

According to [47], the objective of research in heat exchanger network synthesis is to design a heat exchanger network that minimizes the total annualized

cost, given sets of hot streams, cold streams, hot utilities and cold utilities. Each hot and cold stream has a specific heat capacity flow rate, a start and a target temperature. For a more detailed description of the problem, see [47] and the references therein. Different optimization based solution approaches are given e.g., in [14, 47, 70]. Even though the problem has multiple conflicting objectives, the conflicting objectives have been explicitly handled only in [47].

A way to model the heat exchanger network synthesis, called the SynHeat model, has been introduced in [89]. Based on this model, a four-objective optimization problem to minimize the cold utility consumption, hot utility consumption, number of heat exchanger units and total heat exchanger surface area was introduced in [47]. The problem was modeled with GAMS (see [30]) and solved with the NIMBUS method that was also implemented on GAMS.

In Paper [PIII], a simplification of the heat exchanger network synthesis problem of [47] was treated, where the cold utility was dropped from the set of objectives. The cold utility could be dropped because it was found that it correlated with the hot utility and, thus, it was enough to consider only one of them. In that paper, nine Pareto optimal outcomes were taken as the given set of Pareto optimal outcomes. This set of outcomes acted as a basis for the Pareto front approximation. These Pareto optimal outcomes were the Pareto optimal outcomes found in [47] with the NIMBUS method. Our aim was to see whether we could gain any more insight about the problem by approximating the Pareto front and then exploring the approximation by NIMBUS. Naturally, in a rigorous treatment of the problem the set of Pareto optimal outcomes would be larger. As the purpose of Paper [PIII] was to act merely as a proof of concept of decision making on an inherently nondominated Pareto front approximation, the aim was for simplicity of the treatment and, thus, only the nine Pareto optimal outcomes were chosen.

As the given set of Pareto optimal outcomes was so small, the inherently nondominated Pareto front approximation could be constructed manually with the ideas of Paper [PII], i.e., without the PAINT method introduced in Paper [PIV]. After the Pareto front approximation had been constructed, a surrogate problem was formulated. The surrogate problem was a nonlinear mixed integer problem, because the mixed integer linear surrogate problem (10) was not developed before Paper [PIV]. The surrogate problem was, however, simple enough that the scalarizations were solvable with the single objective solvers available within WWW-NIMBUS, which was chosen as the implementation of the NIMBUS method. The NIMBUS method was chosen, because of its success in solving real-life multiobjective optimization problems (see Section 2.4.1).

The decision maker was able to find a preferred solution to surrogate problem with WWW-NIMBUS. The preferred approximate Pareto optimal outcome was then projected on the Pareto front of the original problem with the help of the achievement scalarizing problem (4). The Pareto optimal outcome obtained was fairly close to the approximate one and the decision maker was satisfied with it.

In conclusion, the decision maker was able to explore the Pareto front ap-

proximation through the surrogate problem. The exploration of the Pareto front approximation was cognitively easy to the decision maker with the help of the NIMBUS method. The decision maker was able to find a preferred solution to the original problem in a computationally inexpensive way.

## 6.2 Designing and Operating a Wastewater Treatment Plant

Designing and operating a wastewater treatment plant is a complex problem with multiple conflicting objectives. In [77], a five-objective optimization problem of designing and operating wastewater treatment plant was examined. The approach of [77] was to use the interactive NIMBUS method (and more specifically its implementation IND-NIMBUS<sup>®</sup>) to design and find operating settings for a wastewater treatment plant using activated sludge process. The problem was an extension of the three-objective problem treated in [36], where only a part of the plant was considered.

Figure 17 describes the schematic layout of the wastewater treatment plant that was designed in [77]. The wastewater treatment begins with grit removal. After the grit removal, solids are separated by a gravitational settling. Raw and mixed sludge removed from the primary settlers is fermented in a separate reactor and partly recycled back to the water line to provide readily biodegradable carbon source for denitrification. The bioreactor consists of four anoxic zones, three aerobic zones and one deoxygenation zone. Nitrate-rich activated sludge is recycled from zone 8 of the bioreactor to zone 1. Return sludge and primary effluent are directed to zone 1. Methanol is injected to zone 2 to support denitrification. Excess sludge is pumped from zone 8 of the bioreactor to the beginning of the water process, from which it is removed in the primary settlers together with raw sludge. Raw and mixed sludge is thickened gravitationally into approximately 4.5% total solids prior to anaerobic digestion. Anaerobic digestion produces biogas and the produced biogas can be converted into electrical or thermal energy. The digested sludge is dewatered by centrifuges into approximately 28% total solids. The reject water from sludge treatment is pumped to the beginning of the plant. For a treatment of state-of-the-art practices in wastewater treatment, see e.g., [71].

The objectives of the five-objective optimization problem of [77] were amount of nitrogen in the effluent, consumption of energy in the bioreactor, methanol consumption, excess sludge production and biogas production. The first one is the main goal of the activated sludge process and the four others are connected to the operational costs. The multiobjective optimization problem allowed the simultaneous consideration of different aspects related to the performance of the plant (through nitrogen removal rate) and different aspects of the operational costs. Naturally, the last objective was maximized and the others were minimized. The decision variables were percentage of sludge pumped into fermentation from the primary gravitational settling, amount of excess sludge removed from aeration,

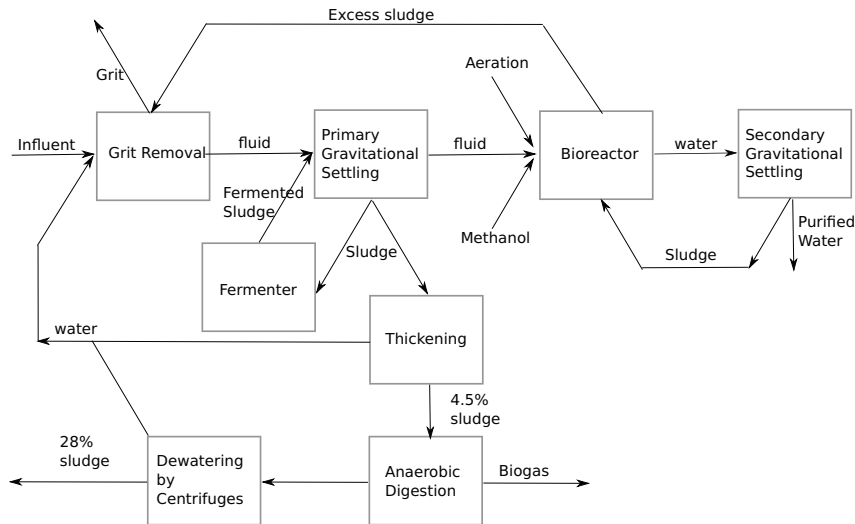


FIGURE 17 A schematic layout of the wastewater treatment plant

$O_2$  concentration in the last aerobic zone of the bioreactor and methanol dosage. Like the model used e.g., in [36], this model was based on findings made in Pöyry Engineering Ltd.

Each simulation of the wastewater treatment plant of [77] takes about 11 seconds on the GPS-X simulator. This makes the problem computationally expensive. In addition, one can notice from the Pareto optimal outcomes computed for the problem that the problem is nonconvex. During the analysis in [77], 200 simulations were run to optimize each of the subproblems of the synchronous NIMBUS method with the Controlled Random Search algorithm (see [72]), which is implemented within the IND-NIMBUS<sup>®</sup> software framework. This means that each iteration of the synchronous NIMBUS method took more than half an hour even with only one scalarization. Even though, interesting compromise solutions to the problem were found in [77], the computational time of iterations was an inconvenience to the decision maker Mr. Kristian Sahlstedt (according to personal communications with the authors of [77]).

Because of the computational cost of the wastewater treatment problem of [77], it was worthwhile to revisit this problem using PAINTE. This was done in Paper [PV]. In order to study, how the introduction of the PAINTE method changes the use of the interactive method, all the remaining circumstances were kept the same in solving the problem. I.e., the synchronous NIMBUS method was used in decision making and the decision maker was Mr. Kristian Sahlstedt, as was in [77]. The PAINTE method was combined with the NIMBUS method using the IND-NIMBUS<sup>®</sup> PAINTE module that was also introduced in Paper [PV]. Because the only thing that changed was the introduction of the PAINTE method into the decision making process, Mr. Sahlstedt was able to compare the experiences of

using the NIMBUS method with and without the PAINTE method. In the following, the decision making process of Paper [PV] is described and conclusions are drawn.

The Pareto front approximation with the PAINTE method always starts with computing the given set of Pareto optimal outcomes. A set of 197 mutually nondominated solutions to the problem was produced using the UPS-EMO evolutionary algorithm of [2]. The mutually nondominated solutions were locally improved using an achievement scalarizing problem of [83] and Matlab `fmincon` function with finite differences approximated gradients. The result was 195 mutually nondominated solutions, because some of the solutions became dominated by others. The implied 195 outcomes were taken as the given set of Pareto optimal outcomes for the PAINTE method.

Next, the Pareto front was approximated with PAINTE based on the given set of Pareto optimal outcomes. The approximation consisted of 4272 polytopes with the given set of Pareto optimal outcomes as their vertices. As described in Paper [PV], the Pareto front approximation implied a mixed integer linear surrogate problem (10) with 3040 continuous variables and 608 binary variables.

The mixed integer linear surrogate problem was inputted into the IND-NIMBUS<sup>®</sup> PAINTE module. The module allowed the decision maker to investigate the approximate Pareto optimal outcomes (i.e., the outcomes of the surrogate problem) and to find the closest actual Pareto optimal solution to the approximate outcome whenever he so wished. The computationally inexpensive investigation of approximate Pareto optimal outcomes through the surrogate problem removed the inconvenience that the decision maker would have had to wait while new solutions based on his/her preferences are generated for investigation.

Mr. Kristian Sahlstedt felt that using the PAINTE and NIMBUS methods together was a definite improvement to using the NIMBUS method directly with the original computationally expensive problem. In addition, he was able to gain more understanding about the problem, because he was more willing to experiment with the surrogate problem, since it was computationally inexpensive. Finally, Mr. Sahlstedt found a preferred solution to the wastewater treatment problem by projecting a solution to the surrogate problem to the Pareto front of the original problem. Paper [PV] contains a more thorough description of the decision making procedure and the tools that were used.

## 7 AUTHOR'S CONTRIBUTION

The idea to approximate the Pareto front came from my supervisor Professor Kaisa Miettinen. In communications with Prof. Miettinen, our research objectives were established.

After having established the goal of my research, I started doing literature review. The aim of this was to find out the challenges in approximating the Pareto fronts of nonconvex multiobjective optimization problems and to see if any of the existing Pareto front approximation methods can handle these challenges. Furthermore, I wanted to see how they fail if that is the case.

During the literature review, I constructed the motivational example given in Section 4.1. With this motivation, I decided to approach the Pareto front approximation problem from a different direction. I did not directly try to come up with a Pareto front approximation method, but instead developed the inherent nondominance concept. The inherent nondominance of a Pareto front approximation guarantees that the Pareto front approximation will not fail e.g., with the motivational example in Section 4.1 in a way that I had found most of the Pareto front approximations to fail.

At this point, Prof. Miettinen and I decided that I would present the idea in the MCDM2009 conference and the conference proceedings. This paper became Paper [PI] that is included in this thesis. In writing the paper, Prof. Miettinen and I had the privilege of having Prof. Margaret M. Wiecek from Clemson University, U.S.A, as our co-author. I was the main author of Paper [PI] and the primary source of new ideas. The co-authors Professors Miettinen and Wiecek contributed with their guidance, helpful comments, ideas and feed-back. As this was my first scientific paper, the role of my co-authors was rather large in writing the paper.

Paper [PII], containing the mathematical basis for the Pareto front approximation, is again mainly written by me. Before starting the writing process, I had already developed an idea about the basis for the Pareto front approximation. Professors Miettinen and Wiecek played an active role by always challenging me to think everything again and to write a better scientific paper. They encouraged me always to find new ideas from a variety of areas of literature. Their encouragement led me to modify the idea and bring new aspects from compu-

tational geometry to the paper. However, the ideas always came from me, and my co-authors' role was to question my ideas and to use their knowledge and experience to guide me if I had come up with alternative research directions.

In Paper [PIII], only Prof. Miettinen collaborated with me. Prof. Miettinen had the idea of applying the ideas of Paper [PII] to a multiobjective optimization problem treated in [47]. In addition, Prof. Miettinen contributed with her knowledge on the problem. I computed the inherently nondominated Pareto front approximation for the problem, did the analysis of the Pareto front approximation with the WWW-NIMBUS software and did most of the writing of the paper. Prof. Miettinen instructed me and provided support during the writing process.

For Paper [PIV], introducing the PAINTE method (that implements the ideas from previous papers) and describing implementation issues, Prof. Wiecek rejoined Prof. Miettinen and me in doing the research and writing the paper. I was again the main author and the main source of ideas. My independence in writing this paper even grew, because I had learned from the previous papers.

Paper [PV], describing an application of the PAINTE method to designing a wastewater treatment plant and introducing the IND-NIMBUS<sup>®</sup> PAINTE module, was written by my co-author Mr. Vesa Ojalehto and me. In this paper, I was responsible for providing the theoretical background on the PAINTE method, and mostly writing the paper. Mr. Ojalehto was on his behalf responsible for implementing the PAINTE module. During the writing of this paper, Dr. Jussi Hakanen provided his expertise in the application and the simulator. Mr. Kristian Sahlstedt acted as our decision maker. In addition, Prof. Miettinen provided us with support during the writing process.

## 8 CONCLUSIONS AND FURTHER RESEARCH

In this thesis, we have introduced the PAIN<sub>T</sub> (PAREto front IN<sub>T</sub>erpolation) method for interpolating between a set of Pareto optimal outcomes. The method is especially useful in solving computationally expensive multiobjective optimization problems, because the Pareto front approximation constructed with the PAIN<sub>T</sub> method implies a computationally inexpensive mixed integer linear surrogate problem for the original problem.

The PAIN<sub>T</sub> method is based on the concept of inherent nondominance, also introduced in this thesis. Inherent nondominance is a property of a set that we have shown to be beneficial for a Pareto front approximation.

In addition, we have shown in this thesis that ideas taken from computational geometry (see e.g., [23]) can be adapted to approximating the Pareto front. In this way, we have opened new doors for further research, as ideas for Pareto front approximations have previously mainly come from mathematical analysis and optimization.

We have also developed a new IND-NIMBUS<sup>®</sup> PAIN<sub>T</sub> module that combines the PAIN<sub>T</sub> method and the interactive NIMBUS method. This module has been applied to solve a computationally expensive five-objective problem of designing and operating a wastewater treatment plant. We have shown that the PAIN<sub>T</sub> method and the new module are effective tools in helping the decision maker find a preferred solution with an interactive method. The PAIN<sub>T</sub> method and the IND-NIMBUS<sup>®</sup> PAIN<sub>T</sub> module should prove useful in helping interactive multiobjective optimization find new application areas in the future.

As further research on this subject, we are going to investigate different applications of the PAIN<sub>T</sub> method. The single real application with a real decision maker (i.e., the wastewater treatment plant problem) that we have had so far is insufficient for determining the strengths and weaknesses of the method. In addition, different decision makers may have different viewpoints on approximate outcomes provided by the PAIN<sub>T</sub> method and this must also be investigated.

We have already identified three weaknesses of the PAIN<sub>T</sub> method that are possible directions for further research. First, the Pareto front approximation constructed with the PAIN<sub>T</sub> method does not provide any information about its



preimage in the decision space, which may be an issue in problems, where the decision variables are also meaningful to the decision maker. Second, the PAIN method interpolates between any set of outcomes whenever the interpolation is inherently nondominated. Thus, the PAIN method is not capable of detecting any disconnectedness in the Pareto front, which naturally is a problem. Third, the PAIN method's computational cost grows exponentially with the number of objectives and more efficient ways of implementing it are needed. Currently, a problem with five objectives and about 200 Pareto optimal outcomes in the given set of Pareto optimal outcomes is at the upper limit of what the PAIN method can handle. For this kind of situations, running the algorithm of the PAIN method takes about 3-4 hours depending on the geometry of the Pareto front and, naturally, the computer that the PAIN method is run on. Fortunately, the decision maker does not have to wait while the approximation is computed but he/she gets involved only after this step, as described also in Figure 15.

## 9 ERRATA

In Paper [PIII], the inherently nondominated Pareto front approximation was constructed without the PAINT method, but directly with rules (R1) and (R2) given in Paper [PII]. Without an algorithm, the construction of the Pareto front approximation was susceptible to errors and this is exactly what happened. The Pareto front approximation given in Paper [PIII] is inherently nondominated, but it does not follow rules (R1) and (R2). The inherently nondominated Pareto front approximation that follows rules (R1) and (R2) contains polytopes given in Table 2 and shown in Figure 18.

However, since the Pareto front approximation in Paper [PIII] is an inherently nondominated Pareto front approximation, the conclusions drawn in Paper [PI] hold. Thus, the analysis following the construction of the Pareto front approximation is justified. However, since the inherently nondominated Pareto front approximation in Figure 18 does not include the hole that was in the Pareto front approximation of Paper [PIII] (compare with Figure 1 in Paper [PIII]) it would have had more approximate outcomes to choose on it. Thus, the decision maker's preferences may have been better met when using the NIMBUS method to solve the surrogate problem that would have in that case had more feasible solutions. The author of this thesis thinks that the fact that there was a mistake in trying to construct the Pareto front approximation directly with rules (R1) and (R2) further emphasizes the need for the PAINT method that was then developed in Paper [PIV].

TABLE 2 Polytopes of the inherently nondominated Pareto front approximation that follows rules (R1) and (R2) for the heat exchanger network synthesis problem of Paper [PIII]

Polytope	Vertices	Polytope	Vertices	Polytope	Vertices
$p^1$	$p^1$	$p^2$	$p^2$	$p^3$	$p^3$
$p^4$	$p^4$	$p^5$	$p^5$	$p^6$	$p^6$
$p^7$	$p^7$	$p^8$	$p^8$	$p^9$	$p^9$
$p^{10}$	$p^1 p^3$	$p^{11}$	$p^1 p^6$	$p^{12}$	$p^1 p^9$
$p^{13}$	$p^2 p^4$	$p^{14}$	$p^2 p^7$	$p^{15}$	$p^2 p^8$
$p^{16}$	$p^3 p^4$	$p^{17}$	$p^3 p^7$	$p^{18}$	$p^3 p^9$
$p^{19}$	$p^4 p^7$	$p^{20}$	$p^4 p^8$	$p^{21}$	$p^4 p^9$
$p^{22}$	$p^6 p^8$	$p^{23}$	$p^6 p^9$	$p^{24}$	$p^8 p^9$
$p^{25}$	$p^1 p^3 p^9$	$p^{26}$	$p^1 p^6 p^9$	$p^{27}$	$p^2 p^4 p^7$
$p^{28}$	$p^2 p^4 p^8$	$p^{29}$	$p^3 p^4 p^7$	$p^{30}$	$p^3 p^4 p^9$
$p^{31}$	$p^4 p^8 p^9 p^4$	$p^{32}$	$p^6 p^8 p^9 p^6$		

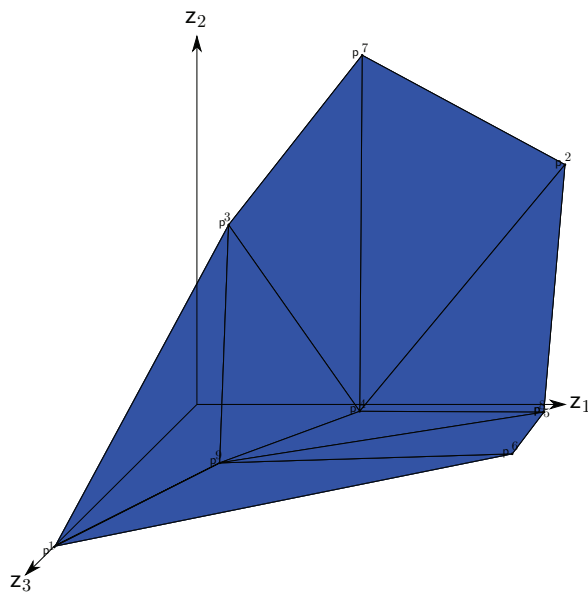


FIGURE 18 The inherently nondominated Pareto front approximation that follows rules (R1) and (R2) for the heat exchanger network synthesis problem of Paper [PIII]

## YHTEENVETO (FINNISH SUMMARY)

Tässä väitöskirjassa, jonka otsikko on "Interpolointimenetelmä epäkonvekseen monitavoiteoptimointiin", käsitellään laskennallisesti vaativia monitavoiteoptimointitehtäviä. Nämä ovat optimointitehtäviä, joissa on monia tavoitteita ja ainakin yhden objektifunktion arvon selvittäminen vaatii paljon aikaa. Tämä on tilanne esimerkiksi silloin kun objektifunktioiden laskemiseen täytyy käyttää raskaita tietokonesimulaatioita.

Monitavoiteoptimointitehtävien ratkaisemisessa niin kutsuttu analyytikko (monitavoiteoptimointimenetelmien asiantuntija) auttaa päätöksentekijää (sovel-lusalan asiantuntija) löytämään parhaan mahdollisen kompromissin tehtävälle. Interaktiiviset monitavoiteoptimointimenetelmät on nähty potentiaalisina, koska ne auttavat päätöksentekijää oppimaan tehtävästä ja tavoitteistaan. Interaktiivisissa menetelmissä päätöksentekijä voi iteratiivisesti etsiä parasta ratkaisua ja mahdollisesti muuttaa preferenssejään. Lisäksi, koska päätöksentekijä voi ohjata interaktiivista menetelmää haluamilleen päätösavaruuden alueille, täytyy yleensä varsin harvoja ratkaisuja tutkia halutun ratkaisun löytämiseksi.

Laskennallisesti vaativat monitavoiteoptimointitehtävät asettavat haasteita kuitenkin myös interaktiivisille menetelmille. Koska interaktiivisia menetelmiä käyttäessä täytyy uusi ratkaisu löytää jokaisella iteraatiolla, voi niiden käyttäminen tulla erittäin hitaaksi. Tämä voi johtaa päätöksentekijän tuskastumiseen tai hän voi jopa menettää uskonsa menetelmään. Erityisesti tätä ongelmaa on tutkitu tässä väitöskirjassa.

Interaktiivisten menetelmien iteraatioiden nopeuttamiseksi on tässä väitöskirjassa kehitetty PAINT-menetelmä, joka interpoloi tunnettujen ratkaisujen välillä. PAINT-menetelmän tuottamalla interpolaatiolla voidaan liikkua käyttäen mitä tahansa interaktiivista menetelmää, koska tämä interpolaatio voidaan esittää laskennallisesti vähemmän vaativana lineaarisena sekalukutehtävänä. Näin päätöksentekijä voi tutkia interpoloituja ratkaisuja ilman laskennallista vaativuutta, vaikka alkuperäinen tehtävä olisikin sellainen.

Tämä väitöskirja sisältää myös kaksi sovellusta, joiden tarkoitus osoittaa PAINT menetelmän hyödyt. Ensimmäinen sovellus näyttää, että interpolointi on käytännöllinen lähestymistapa laskennallisesti raskaan lämmönvaihtimen monitavoitteisessa suunnittelussa. Toisessa sovelluksessa suunnitellaan jätevedenpuhdistamo. Tässä sovelluksessa käytetään myös kehittämäämme IND-NIMBUS<sup>®</sup> PAINT lisäosaa, joka yhdistää PAINT-menetelmän ja interaktiivisen NIMBUS menetelmän IND-NIMBUS<sup>®</sup> ohjelmistokehikossa. Samalla tutkitaan päätöksentekijän toimintaa tämän ongelman ratkaisemisessa PAINT menetelmän avulla.

Väitöskirjan lähestymistapa laskennallisesti vaativien epäkonveksien monitavoitteisten optimointitehtävien ratkaisemiseen on uusi. Väitöskirja avaa siten uusia sovelluksia interaktiiviselle monitavoitteiselle optimoinnille tehtäviin, joita on aikaisemmin pidetty niille liian vaikeina.

## REFERENCES

- [1] T. Aittokoski and K. Miettinen. Cost Effective Simulation-Based Multiobjective Optimization in the Performance of an Internal Combustion Engine. *Engineering Optimization*, 40:593–612, 2008.
- [2] T. Aittokoski and K. Miettinen. Efficient Evolutionary Approach to Approximate the Pareto-Optimal Set in Multiobjective Optimization, UPS-EMOA. *Optimization Methods and Software*, 25:841 – 858, 2010.
- [3] S. Andradottir. Simulation Optimization. In J. Banks, editor, *Handbooks of Simulation: Principles, Methodology, Advances, Applications, and Practice*. John Wiley & Sons, New York, 1998.
- [4] D. Attali, H. Edelsbrunner, J. Harer, and Y. Mileyko. Alpha-Beta Witness Complexes. In Frank Dehne, Jörg-Rüdiger Sack, and Norbert Zeh, editors, *Algorithms and Data Structures*, pages 386–397. Springer-Verlag Berlin, Heidelberg, 2007.
- [5] T. Bäck, J. Knowles, and O. M. Shir. Experimental Optimization by Evolutionary Algorithms. In *Proceedings of the 12th Annual Conference Companion on Genetic and Evolutionary Computation*, pages 2897–2916. ACM, New York, 2010.
- [6] C. A. Bana e Costa, J.-M. de Corte, and J.-C. Vansnick. On the Mathematical Foundations of MACBETH. In J. Figueira, S. Greco, and M. Ehrgott, editors, *Multiple Criteria Decision Analysis: State of the Art Surveys*. Springer-Verlag, New York, 2005.
- [7] J. Banks. *Handbook of Simulation: Principles, Methodology, Advances, Applications, and Practice*. John Wiley & Sons, New York, 1998.
- [8] C. B. Barber, D. P. Dobkin, and H. Huhdanpää. The Quickhull Algorithm for Convex Hulls. *ACM Transactions on Mathematical Software*, 22:469–483, 1996.
- [9] V. Belton, J. Branke, P. Eskelinen, S. Greco, J. Molina, F. Ruiz, and R. Slowinski. Interactive Multiobjective Optimization from a Learning Perspective. In J. Branke, K. Deb, K. Miettinen, and K. Slowinski, editors, *Multiobjective Optimization: Interactive and Evolutionary Approaches*. Springer-Verlag Berlin, Heidelberg, 2008.
- [10] R. Benayon, J. de Montgolfier, J. Tergny, and O. Laritchev. Linear Programming with Multiple Objective Functions: Step Method. *Mathematical Programming*, 1:366–375, 1971.
- [11] L. T. Biegler. Chemical Process Simulation. *Chemical Engineering Progress*, 85:50–61, 1989.

- [12] J.T. Buchanan. A Naive Approach for Solving MCDM Problems: the GUESS Method. *Journal of the Operational Research Society*, 48:202–206, 1997.
- [13] J. R. Carley, B. R. J. Schwedler, M. E. Baldwin, and R. J. Trapp. A Proposed Model-Based Methodology for Feature-Specific Prediction for High-Impact Weather. *Weather Forecasting*, 26:243–249, 2011.
- [14] J. Cerda and A. W. Westerberg. Synthesizing Heat Exchanger Networks Having Restricted Stream/Stream Matches Using Transportation. *Chemical Engineering Science*, 38:1723–1740, 1983.
- [15] V. Chankong and Y. Y. Haimes. Interactive Surrogate Worth Trade-Off (ISWT) Method for Multiobjective Decision-Making. In S. Zionts, editor, *Multiple Criteria Problem Solving*, pages 42–67. Springer-Verlag Berlin, Heidelberg, 1978.
- [16] V. Chankong and Y. Y. Haimes. *Multiobjective Decision Making: Theory and Methodology*. North-Holland, Amsterdam, 1983.
- [17] C. A. Coello Coello, D. A. Van Veldhuizen, and G. B. Lamont. *Evolutionary Algorithms for Solving Multi-Objective Problems*. Springer, New York, 2007.
- [18] D. Craft, T. Halabi, H. Shih, and T. Bortfeld. Approximating Convex Pareto Surfaces in Multiobjective Radiotherapy Planning. *Physics in Medicine and Biology*, 33:3399–3407, 2006.
- [19] K. Deb. *Multi-Objective Optimization Using Evolutionary Algorithms*. John Wiley & Sons, Chichester, 2001.
- [20] K. Deb, K. Miettinen, and D. Sharma. A Hybrid Integrated Multi-Objective Optimization Procedure for Estimating Nadir Point. In M. Ehrgott, C. Fonseca, X. Gandibleux, J.-K. Hao, and M. Sevaux, editors, *Evolutionary Multi-Criterion Optimization*, pages 569–583. Springer-Verlag Berlin, Heidelberg, 2009.
- [21] K. Deb, L. Thiele, M. Laumanns, and E. Zitzler. Scalable Multi-Objective Optimization Test Problems. *IEEE International Conference on E-Commerce Technology*, 1:825–830, 2002.
- [22] J. W. Eaton. *GNU Octave Manual*. Network Theory Limited, 2002.
- [23] H. Edelsbrunner. Shape Reconstruction with Delaunay Complex. In C. L. Lucchesi and A. V. Moura, editors, *LATIN'98: Theoretical Informatics*, pages 119–132. Springer-Verlag Berlin, Heidelberg, 1998.
- [24] H. Edelsbrunner and E. P. Mücke. Simulation of Simplicity: a Technique to Cope with Degenerate Cases in Geometric Algorithms. *ACM Transactions on Graphics*, 9:66–104, 1990.

- [25] P. Eskelinen, K. Miettinen, K. Klamroth, and J. Hakanen. Pareto Navigator for Interactive Nonlinear Multiobjective Optimization. *OR Spectrum*, 32:211–227, 2010.
- [26] S. Faulkenberg and M. M. Wiecek. On the Quality of Discrete Representations in Multiple Objective Programming. *Optimization and Engineering*, 11:423–440, 2010.
- [27] S. French, J. Maule, and N. Papamichail. *Decision Behaviour, Analysis and Support*. Cambridge University Press, Cambridge, 2009.
- [28] M. Fu. Optimization via Simulation: A Review. *Annals of Operations Research*, 53:199–248, 1994.
- [29] M. Fu. Optimization for Simulation: Theory vs. Practice. *INFORMS Journal on Computing*, 14:195–215, 2002.
- [30] GAMS (General Algebraic Modeling System) Home Page, Last accessed: November 11, 2011. url = <http://www.gams.com/>.
- [31] GLPK (GNU Linear Programming Kit) Home Page, Last accessed: November 11, 2011. url = <http://www.gnu.org/s/glpk/>.
- [32] T. Goel, R. Vaidyanathan, R. T. Haftka, W. Shyy, N. V. Queipo, and K. Tucker. Response Surface Approximation of Pareto Optimal Front in Multi-Objective Optimization. *Computer Methods in Applied Mechanics and Engineering*, 196:879–893, 2007.
- [33] B. Grünbaum. *Convex Polytopes*. Interscience Publishers, London, 1967.
- [34] Y. Y. Haimes, L. S. Lasdon, and D. A. Wismer. On a Bicriterion Formulation of the Problems of Integrated System Identification and System Optimization. *IEEE Transactions on Systems, Man, and Cybernetics*, 1:296–297, 1971.
- [35] J. Hakanen, Y. Kawajiri, K. Miettinen, and L.T. Biegler. Multi-Objective Optimization for Simulated Moving Bed Processes. *Control and Cybernetics*, 36:282–320, 2007.
- [36] J. Hakanen, K. Miettinen, and K. Sahlstedt. Wastewater Treatment: New Insight Provided by Interactive Multiobjective Optimization. *Decision Support Systems*, 51:328–337, 2011.
- [37] M. Hasenjäger and B. Sendhoff. Crawling Along the Pareto Front: Tales From the Practice. In *The 2005 IEEE Congress on Evolutionary Computation (IEEE CEC 2005)*, pages 174–181, Piscataway, NJ, 2005. IEEE Press.
- [38] C. Hwang and A. S. M. Masud. *Multiple Objective Decision Making - Methods and Applications: a State-of-the-Art Survey*. Springer-Verlag Berlin, Heidelberg, 1979.

- [39] IBM ILOG CPLEX Optimization Studio Home Page, Last accessed: November 11, 2011. url = <http://www-01.ibm.com/software/integration/optimization/cplex-optimization-studio/>.
- [40] Y. Jin. A Comprehensive Survey of Fitness Approximation in Evolutionary Computation. *Soft Computing*, 9:3–12, 2005.
- [41] G. K. Kamenev. Study of an Adaptive Single-Phase Method for Approximating the Multidimensional Pareto Frontier in Nonlinear Systems. *Computational Mathematics and Mathematical Physics*, 49:2103–2113, 2009.
- [42] R. L. Keeney. *Value-Focused Thinking: a Path to Creative Decisionmaking*. Harvard University Press, Harvard, 1996.
- [43] J. Knowles and H. Nakayama. Meta-Modeling in Multiobjective Optimization. In J. Branke, K. Deb, K. Miettinen, and R. Slowinski, editors, *Multiobjective Optimization: Interactive and Evolutionary Approaches*, pages 245–284. Springer-Verlag Berlin, Heidelberg, 2008.
- [44] P. Korhonen and J. Wallenius. A Pareto Race. *Naval Research Logistics*, 35:615–623, 1988.
- [45] P. Korhonen and J. Wallenius. Behavioural Issues in MCDM: Neglected Research Questions. *Journal of Multi-Criteria Decision Analysis*, 5:178–182, 1996.
- [46] O. I. Larichev. Cognitive Validity in Design of Decision-Aiding Techniques. *Journal of Multi-Criteria Decision Analysis*, 3:127 – 138, 1992.
- [47] T. Laukkanen, T.-M. Tveit, V. Ojalehto, K. Miettinen, and C.-J. Fogelholm. An Interactive Multi-Objective Approach to Heat Exchanger Network Synthesis. *Computers & Chemical Engineering*, 34:943–952, 2010.
- [48] T. Lehtonen and P. Pulkkinen. A Simple Method for Approximating a General Pareto Surface. *Journal of Multi-Criteria Decision Analysis*, 17:115–124, 2010.
- [49] A. V. Lotov, V. A. Bushenkov, and G. A. Kamenev. *Interactive Decision Maps*. Kluwer Academic Publishers, Boston, 2004.
- [50] A. Lovison. Singular Continuation: Generating Piecewise Linear Approximations to Pareto Sets via Global Analysis. *SIAM Journal on Optimization*, 21:463–490, 2011.
- [51] M. Luque, F. Ruiz, and K. Miettinen. Global Formulation for Interactive Multiobjective Optimization. *OR Spectrum*, 33:27–48, 2011.
- [52] A. Messac, A. Ismail-Yahaya, and C.A. Mattson. The Normalized Normal Constraint method for Generating the Pareto Frontier. *Structural and Multidisciplinary Optimization*, 25:86–98, 2003.



- [53] A. Messac and C. A. Mattson. Normal Constraint Method with Guarantee of Even Representation of Complete Pareto Frontier. *American Institute of Aeronautics and Astronautics*, 42:2101–2111, 2004.
- [54] A. Messac and A. A. Mullur. A Computationally Efficient Metamodeling Approach for Expensive Multiobjective Optimization. *Optimization and Engineering*, 9:37–67, 2008.
- [55] K. Miettinen. *Nonlinear Multiobjective Optimization*. Kluwer Academic Publishers, Boston, 1999.
- [56] K. Miettinen. Using Interactive Multiobjective Optimization in Continuous Casting of Steel. *Materials and Manufacturing Processes*, 22:585–593, 2007.
- [57] K. Miettinen and M. Mäkelä. Interactive Bundle-based Method for Nondifferentiable Multiobjective Optimization: NIMBUS. *Optimization*, 34:231–246, 1995.
- [58] K. Miettinen and M. Mäkelä. Interactive Multiobjective Optimization System WWW-NIMBUS on the Internet. *Computers & Operations Research*, 27:709–723, 2000.
- [59] K. Miettinen and M. M. Mäkelä. On Scalarizing Functions in Multiobjective Optimization. *OR Spectrum*, 24:193–213, 2002.
- [60] K. Miettinen and M. M. Mäkelä. Synchronous Approach in Interactive Multiobjective Optimization. *European Journal of Operational Research*, 170:909–922, 2006.
- [61] K. Miettinen, F. Ruiz, and A. P. Wierzbicki. Introduction to Multiobjective Optimization: Interactive Approaches. In J. Branke, K. Deb, K. Miettinen, and R. Slowinski, editors, *Multiobjective Optimization: Interactive and Evolutionary Approaches*, pages 27–57. Springer-Verlag Berlin, Heidelberg, 2008.
- [62] M. Monz. *Pareto Navigation - Algorithmic Foundation of Interactive Multi-criteria IMRT Planning*. PhD thesis, University of Kaiserslautern, 2006.
- [63] M. Monz, K. H. Kufer, T. R. Bortfeld, and C. Thieke. Pareto Navigation - Algorithmic Foundation of Interactive Multi-Criteria IMRT Planning. *Physics in Medicine and Biology*, 53:985–998, 2008.
- [64] A. A. Mullur and A. Messac. Extended Radial Basis Functions: More Flexible and Effective Metamodeling. *American Institute of Aeronautics and Astronautics*, 43:1306–1315, 2005.
- [65] R.H. Myers and D.C. Montgomery. *Response Surface Methodology: Process and Product Optimization using Designed Experiments*. John Wiley & Sons, Chichester, 1995.

- [66] H. Nakayama, M. Arakawa, and R Sasaki. Simulation Based Optimization for Unknown Objective Functions. *Optimization and Engineering*, 3:201–214, 2002.
- [67] H. Nakayama and Y. Sawaragi. Satisficing Trade-Off for Multiobjective Programming. In M. Grauer and A. P. Wierzbicki, editors, *Interactive Decision Analysis*, pages 113–122. Springer-Verlag Berlin, Heidelberg, 1984.
- [68] H. Nakayama and Y. Yun. Support Vector Regression Based on Goal Programming and Multi-Objective Programming. In *Proceedings of the 2006 International Joint Conference on Neural Networks*, 2006. Available at [http://ieeexplore.ieee.org/xpls/abs\\_all.jsp?arnumber=1716232](http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1716232), Last accessed: November 11, 2011.
- [69] Y. Nikulin, K. Miettinen, and M. Mäkelä. A New Achievement Scalarizing Function based on Parameterization in Multiobjective Optimization. *OR Spectrum*, To Appear. DOI: 10.1007/s00291-010-0224-1.
- [70] S. A. Papoulias and I. E. Grossmann. A Structural Optimization Approach to Process Synthesis - II: Heat Recovery Networks. *Computers & Chemical Engineering*, 7:707–721, 1983.
- [71] H. M. Phillips, K. E. Sahlstedt, K. Frank, J. Bratby, W. Brennan, S. Rogowski, D. Pier, W. Anderson, M. Mulas, J. B. Copp, and N. Shirodkar. Wastewater Treatment Modelling in Practice: a Collaborative Discussion of the State of the Art. *Water Science and Technology*, 59:695–704, 2009.
- [72] W. L. Price. Global Optimization by Controlled Random Search. *Journal of Optimization Theory and Applications*, 40:333–348, 1983. 10.1007/BF00933504.
- [73] G. Rennen, E. R. van Dam, and D. den Hertog. Enhancement of Sandwich Algorithms for Approximating Higher-Dimensional Convex Pareto Sets. *INFORMS Journal on Computing*, To Appear. DOI: 10.1287/ijoc.1100.0419.
- [74] H. Ruotsalainen, K. Miettinen, J.-E. Palmgren, and T. Lahtinen. Interactive Multiobjective Optimization for Anatomy based Three-Dimensional HDR Brachytherapy. *Physics in Medicine and Biology*, 55:4703–4719, 2010.
- [75] S. Ruzika and M. M. Wiecek. Approximation Methods in Multiobjective Programming. *Journal of Optimization Theory and Applications*, 126:473–501, 2005.
- [76] J.-H. Ryu, S. Kim, and H. Wan. Pareto Front Approximation with Adaptive Weighted Sum Method in Multiobjective Simulation Optimization. In M. D. Rossetti, R. R. Hill, A. Dunkin, and R. G. Ingalls, editors, *Proceedings of the 2009 Winter Simulation Conference*, pages 623 – 633. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc., 2009.

- [77] K. Sahlstedt, J. Hakanen, and K. Miettinen. Interactive Multiobjective Optimization in Wastewater Treatment Plant Operation and Design. In *Proceedings of the ECWATECH-2010, the IWA Specialist Conference 'Water and Wastewater Treatment Plants in Towns and Communities of the XXI Century: Technologies, Design and Operation'*, CD-ROM, 2010.
- [78] L. V. Santana-Quintero, A. A. Montano, and C. A. Coello Coello. A Review of Techniques for Handling Expensive Functions in Evolutionary Multi-Objective Optimization. In Y. Tenne and C.-K. Goh, editors, *Computational Intelligence in Expensive Optimization Problems: Adaptation, Learning, and Optimization*, pages 29–59. Springer-Verlag Berlin, Heidelberg, 2010.
- [79] Y. Sawaragi, H. Nakayama, and T. Tanino. *Theory of Multiobjective Optimization*. Academic Press, Orlando, 1985.
- [80] P. K. Shukla and K. Deb. On Finding Multiple Pareto-Optimal Solutions Using Classical and Evolutionary Generating Methods. *European Journal of Operational Research*, 181:1630 – 1652, 2007.
- [81] J. A. Sokolowski and C. M. Banks, editors. *Principles of Modeling and Simulation: A Multidisciplinary Approach*. John Wiley & Sons, New Jersey, 2009.
- [82] R. E. Steuer. *Multiple Criteria Optimization: Theory, Computation and Application*. John Wiley & Sons, New York, 1986.
- [83] A. P. Wierzbicki. The Use of Reference Objectives in Multiobjective Optimization. In G. Fandel and T. Gal, editors, *Multiple Criteria Decision Making Theory and Applications*, pages 468–486. Springer-Verlag Berlin, Heidelberg, 1980.
- [84] A. P. Wierzbicki. A Mathematical Basis for Satisficing Decision Making. *Mathematical Modelling*, 3:391–405, 1982.
- [85] A. P. Wierzbicki. A Methodological Approach to Comparing Parametric Characterizations of Efficient Solutions. In G. Fandel, M. Grauer, A. Kurzhanski, and A. P. Wierzbicki, editors, *Large-Scale Modelling and Interactive Decision Analysis*. Springer-Verlag Berlin, Heidelberg, 1986.
- [86] A. P. Wierzbicki. On the Completeness and Constructiveness of Parametric Characterizations to Vector Optimization Problems. *OR Spectrum*, 8:73–87, 1986.
- [87] B. Wilson, D. Cappelleri, T. W. Simpson, and M. Frecker. Efficient Pareto Frontier Exploration Using Surrogate Approximations. *Optimization and Engineering*, 2:31–50, 2001.
- [88] L. Xu, T. Reinikainen, W. Ren, B. P. Wang, Z. Han, and D. Agonafer. A Simulation-Based Multi-Objective Design Optimization of Electronic Packages Under Thermal Cycling and Bending. *Microelectronics Reliability*, 44:1977 – 1983, 2004.

- [89] T. F. Yee and I. E. Grossmann. Simultaneous Optimization of Models for Heat Integration - II. Heat Exchanger Network Synthesis. *Computers and Chemical Engineering*, 14:1165–1184, 1990.
- [90] E. Zitzler, J. Knowles, and L. Thiele. Quality Assessment of Pareto Set Approximations. In J. Branke, K. Deb, K. Miettinen, and R. Slowinski, editors, *Multiobjective Optimization*, pages 373–404. Springer-Verlag, Berlin, Heidelberg, 2008.

## ORIGINAL PAPERS

PI

### DECISION MAKING ON PARETO FRONT APPROXIMATIONS WITH INHERENT NON-DOMINANCE

by

Markus Hartikainen · Kaisa Miettinen · Margaret M. Wiecek 2011

in Y. Shi and S. Wang and G. Kou and J. Wallenius (editors) *New State of MCDM  
in the 21st Century, Selected Papers of the 20th International Conference on Multiple  
Criteria Decision Making 2009*, pages 35–46

Reproduced with kind permission of Springer Science+Business Media.

# Decision Making on Pareto Front Approximations with Inherent Nondominance

Markus Hartikainen, Kaisa Miettinen, and Margaret M. Wiecek

**Abstract** Approximating the Pareto fronts of nonlinear multiobjective optimization problems is considered and a property called inherent nondominance is proposed for such approximations. It is shown that an approximation having the above property can be explored by interactively solving a multiobjective optimization problem related to it. This exploration can be performed with available interactive multiobjective optimization methods. The ideas presented are especially useful in solving computationally expensive multiobjective optimization problems with costly function value evaluations.

## 1 Introduction

In multiobjective optimization (MO), the problem is to find a preferred solution in the presence of several conflicting objectives (see e.g., [7, 17, 23]). In such problems, there is no single well-defined optimal solution but several mathematically equally good solutions, so-called Pareto optimal (PO) solutions, can be identified. To be able to find the most preferred among them as the final solution usually additional preference information is needed from a human decision maker (DM), who knows the problem domain.

According to [7], MO methods can be divided into no-preference methods, a priori methods, a posteriori methods and interactive methods. For details of these different methods see e.g., [17]. Interactive methods can be seen as the most prominent - as argued in [20] - because the DM can learn about the problem as e.g., in [4] and adjust his/her goals accordingly. Because only such PO solutions are generated that are interesting to the DM, interactive methods are generally computationally much less expensive than, for example, a posteriori methods, for which many PO

---

Markus Hartikainen · Kaisa Miettinen · Margaret M. Wiecek\*

Department of Mathematical Information Technology, University of Jyväskylä, Finland

\* On sabbatical leave from Clemson University, South Carolina, USA

solutions must be computed in order for the DM to get enough alternatives for making an informed decision.

If a multiobjective optimization problem (MOP) is computationally very demanding i.e., function evaluations take a lot of time, even the classical interactive methods can have problems; the DM may get frustrated in waiting while new PO solutions that reflect his/her preferences are computed. This high computational demand occurs often, for example, with the so-called black-box MOPs, whose objective functions are not explicitly known as algebraic functions of decision variables but need to be e.g., simulated. Approaches based on evolutionary multiobjective optimization methods have been proposed e.g. in [11, 14] for finding PO solutions for computationally expensive MOPs. Another way to handle such MOPs is to use design of experiments for constructing surrogate approximations of the objective functions and then solving the surrogate MOP [26, 25]. However, according to [9], the use of surrogate approximations loses its computational benefits as the number of objectives increases. The approaches based on evolutionary algorithms have their issues too; in order to produce a dense Pareto set representation, a large number of solutions must be computed, which is time-consuming or sometimes even impossible. On the other hand, the choice of the final outcome from a large number of PO outcomes is cognitively very demanding as noted in [13].

Instead of estimating each objective function with a surrogate, one can estimate the set of PO outcomes (also called the Pareto front (PF)). In [4], an interactive method called Pareto Navigator is proposed, in which a PF approximation for a convex MOP is constructed based on a relatively small set of PO outcomes. Having constructed the approximation, the preferences of the DM can be reflected in real time by moving on the approximation in a way that resembles the visual interactive Pareto Race method for linear problems [12]. When a preferred point on the approximation has been found, the closest actual PO outcome can be found by means of the achievement scalarizing function [24].

The construction of a PF approximation (that allows consideration of the objective vectors between known PO outcomes) and its exploration with MO methods is a novel approach to computationally demanding MOPs and has some potential benefits. When compared to a posteriori methods, some computational expense can be avoided because a smaller number of PO outcomes is sufficient. Since the DM is allowed to explore the approximation by iteratively expressing his/her preferences, the cognitive load of going through a large set of PO outcomes becomes lighter. The DM can also learn about the problem in the process. Finally, since the construction of the PF approximation does not have any special requirements for the initial PO outcomes, they may be generated with any method for finding PO outcomes, and because the DM is not involved in this, the possible computational cost is not a problem.

The Pareto Navigator method is, however, only applicable to convex MOPs, because of the type of approximation it uses. The ultimate goal behind our study is to develop a MO method that uses the Pareto Navigator's idea of navigating on the approximation for general continuous nonlinear MOPs. The aim of this paper is to concentrate on the issue of what kind of approximation can be used for the above

mentioned exploration. Because interactive MO methods have been proven effective in finding a preferred point on the PF, we want to employ them also for finding a preferred point on the approximation. The process of finding a preferred point on a PF approximation (by using interactive MO methods or other methods) is referred to in this paper as *decision making on the approximation*.

There are numerous methods for approximating the PF of a MOP. A survey [22] summarizes methods published before 2003. More recent approximation methods for nonlinear MOPs are given e.g., in [3, 5, 15, 16]. However, the uses of the PF approximations in decision making have not received the attention that they deserve. Among the few exceptions are [2, 4, 8, 10, 15, 18, 21]. None of these, however, aims at using general interactive MO methods in exploring the PF approximation. The approach of this paper is, thus, different from all the above; we aim at characterizing an approximation for a specific use of exploration with interactive MO methods.

The paper is structured as follows. Some terminology and notation is set in Section 2. We introduce a new property called inherent nondominance (IND) for a PF approximation and prove its relevant properties in Section 3. In Section 4, decision making on the approximation is discussed and an example is given. We also propose an approach to exploring an IND PF approximation by solving a special type MOP. Finally, we draw conclusions in Section 5.

## 2 Notation and definitions

In this paper we study multiobjective optimization problems (MOPs)

$$\min_{\text{s.t. } x \in X} (f_1(x), \dots, f_k(x)), \quad (1)$$

where  $f_i : X \rightarrow \mathbb{R}$  is a function for all  $i \in \{1, \dots, k\}$ ,  $X \subset \mathbb{R}^n$  and  $k, n \in \mathbb{N}$ . We also define a vector valued function  $f : X \rightarrow \mathbb{R}^k$ ,  $f(x) = (f_1(x), \dots, f_k(x))^T$  for all  $x \in X$ .

The set  $X$  is called the feasible decision set of (1) and every  $x \in X$  is called a (feasible) decision. The set  $f(X)$  is the feasible outcome set of (1) and every  $z \in f(X)$  is called a (feasible) outcome. The set  $\mathbb{R}^k$  is called the objective space of (1) and every point  $z \in \mathbb{R}^k$  is called an objective vector. With the above notation, the MOP (1) can be formulated as

$$\min_{\text{s.t. } z \in f(X)} z. \quad (2)$$

For any objective vectors  $z^1, z^2 \in \mathbb{R}^k$  notation  $z^1 \leq z^2$  means that  $z_i^1 \leq z_i^2$  for all  $i = 1, \dots, k$  and  $z^1 \neq z^2$ . If  $z^1 \leq z^2$  it is said that  $z^1$  dominates  $z^2$ . If  $B \subset \mathbb{R}^k$  is a set, then a point  $b \in B$  is Pareto optimal (PO) in  $B$ , if there does not exist a point  $\tilde{b} \in B$  that dominates  $b$ . An outcome  $z \in f(X)$  is PO for (1), if  $z$  is PO in  $f(X)$ . Finally, the Pareto front (PF) of  $B$ ,  $PF(B)$ , is the set of PO points in  $B$  and the PF of a MOP,  $PF$ , is the set of PO outcomes of the MOP. A feasible decision  $x \in X$  is a PO solution to (1), if  $f(x)$  is a PO outcome.



We assume a rational decision maker (DM): he/she always prefers a PO objective vector over a dominated one in any set of possible objective vectors. In other words, for every set  $A \subset \mathbb{R}^k$  the preferred objective vectors of the DM belong to  $PF(A)$  and, furthermore, the DM is not interested in  $A \setminus PF(A)$  if he/she is aware of  $PF(A)$ .

The starting point of this study is as follows. Assume that we are given a computationally demanding MOP with the above notation. This problem is referred to as the *initial MOP*. If not otherwise defined, the notations PO and PF always refer to this problem. We also assume that we are given a finite set of PO outcomes  $P$ . This set is referred to as the *initial set of PO outcomes*. As in Pareto Navigator, the way this set has been generated is not restricted in any way.

### 3 Inherent nondominance and inherently nondominated approximation of the PF

In this section we introduce the concept of inherent nondominance and other related concepts. We also present the relevant properties characteristic for an inherently nondominated approximation of the PF. These properties will be further interpreted in Section 4.

**Definition 1.** We say that a set  $A \subset \mathbb{R}^k$  is *inherently nondominated (IND)*, if for any two points  $a^1, a^2 \in A$  neither  $a^1 \leq a^2$  nor  $a^2 \leq a^1$ .

**Definition 2.** A set  $A \subset \mathbb{R}^k$  is an *inherently nondominated PF approximation (based on  $P$ )* if  $A$  is IND and  $P \subset A$ .

According to the following theorem, the actual PF satisfies the IND condition. The actual PF should be suitable as an approximation of itself, because naturally it is the best approximation that one can hope for.

**Theorem 1.** *The PF is an IND PF approximation based on  $P$ .*

*Proof.* Since  $P$  is a set of PO outcomes, then  $P \subset PF$ . The property that  $PF$  is IND is trivial, since if  $z^1, z^2 \in f(X)$  so that  $z^1 \leq z^2$ , then  $z^2 \notin PF$ .  $\square$

The following theorem gives a sufficient and necessary condition for a set to be IND and could, thus, have been chosen as the definition for an IND set. We, however, chose the definition given above.

**Theorem 2.** *Let  $A \subset \mathbb{R}^k$  be a set. Then the set  $A$  is IND if and only if  $PF(A) = A$ .*

*Proof.* " $\Rightarrow$ ": Naturally  $PF(A) \subset A$ . To show that  $A \subset PF(A)$ , assume the contrary that there exists a point  $a \in A \setminus PF(A)$ . This implies a point  $b \in A$  so that  $b \leq a$ . But this is a contradiction, since  $A$  is IND.

" $\Leftarrow$ ": To show that  $A$  is IND assume the contrary that there exist distinct points  $a, b \in A$  so that  $a \leq b$ . This would imply that  $b \notin PF(A)$  and would thus be a contradiction.  $\square$

The following corollary follows directly from Theorem 2. It is, however, considerably important, since according to it an IND approximation *extends*  $P$  *without losing Pareto optimality*.

**Corollary 1.** *If a set  $A \subset \mathbb{R}^k$  is an IND PF approximation based on  $P$ , then*

$$P \subset PF(A).$$

*Proof.* Now  $PF(A) = A \supset P$ , because of Theorem 2 and Definition 2.  $\square$

The above properties of an IND approximation are rather straightforward. Theorem 3 establishes the existence of IND PF approximations.

**Theorem 3.** *The set of initial PO outcomes  $P$  is an IND PF approximation based on  $P$ .*

*Proof.* Because outcomes in  $P$  are PO, they do not dominate each other. Also  $P \subset P$  trivially.  $\square$

The IND approximation  $P$  is not, however, interesting, since it does not contain any points outside the set of initial PO outcomes  $P$ . We want to construct IND approximations  $A$  that extend  $P$ , but we must skip the development of algorithms producing such IND approximations because of space limitations. Our general idea is to start with a large set  $B$ ,  $P \subset B \subset \mathbb{R}^k$ , and then remove from  $B$  such points  $b \in B \setminus P$  that violate the IND condition. By Theorem 3, this procedure produces an IND approximation. Also by Zorn's lemma, there exists at least one minimal (by inclusion) set  $\tilde{B} \subset B \setminus P$  such that  $B \setminus \tilde{B}$  is an IND PF approximation.

*Example 1.* Let the set of initial PO points be  $P = \{p^1, p^2, p^3, p^4\}$ , where  $p^1 = (1, 0, 0)$ ,  $p^2 = (0, 1, 0)$ ,  $p^3 = (0, 0, 1)$  and  $p^4 = (2/5, 2/5, 2/5)$ . We compare two piecewise linear PF approximations  $A^1 = \text{conv}(P)$  and  $A^2 = \text{conv}(p^1, p^2, p^4) \cup \text{conv}(p^1, p^3, p^4) \cup \text{conv}(p^2, p^3, p^4)$ , where  $\text{conv}(C)$  is the convex hull of a set  $C$ . These are shown in Figure 1. The convex hull of all outcomes in  $P$  is used e.g., in [4, 21] for approximating the outcome set  $f(X)$  for convex MOPs and has thus been used for getting also a PF approximation. Approximation  $A^2$  is IND, which can be shown analytically. Approximation  $A^1$  is, however, not IND because the point  $(1/3, 1/3, 1/3) \in A^1$  dominates the initial PO outcome  $p^4$  that is also in  $A^1$ . For this reason, approximation  $A^1$  is not able to capture the non-convex shape implied by the initial set of PO outcomes and, unlike the IND approximation  $A^2$ , gives too optimistic estimates.

## 4 Decision making on the approximation

In this section we discuss decision making on the approximation. The underlying reasoning is that once the PF approximation with desired properties has been created, one can navigate on it with significantly lower computational cost than on the

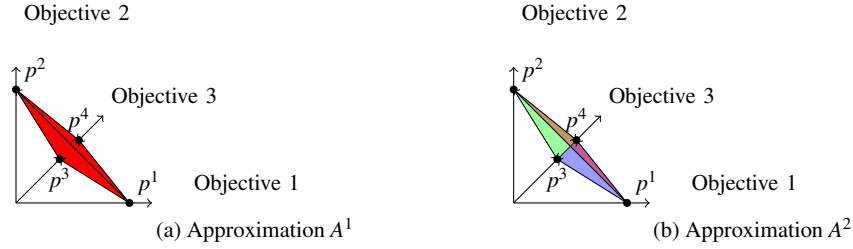


Fig. 1: Two piecewise linear PF approximations constructed with the same initial set of PO outcomes. The approximation  $A^1$  is the convex hull of all points and the approximation  $A^2$  is a union of three triangles.

PF of the original problem. Once a desired objective vector on the approximation has been found, one can generate the closest corresponding PO solution as in Pareto Navigator.

The problem of finding a preferred point on a PF approximation is similar to finding a preferred outcome of a MOP; like the PF, its approximation  $A$  may have many mathematically equally good objective vectors and, thus, there is a need for decision making even on the PF approximation. An important aspect of decision making on an approximation is the question of how well the preferences of the DM on the approximation reflect the preferences on the real PF.

Based on a limited knowledge of the PF in a form of the initial set of PO outcomes  $P$ , the accuracy of the approximation cannot naturally be guaranteed outside the initial PO outcomes. What can, however, be done is that a rational DM is given the opportunity to choose any of the initial PO points in  $P$  while making rational decisions on the approximation. An undesirable occurrence happens e.g., in non-IND approximation  $A^1$  in Example 1, where a rational DM will not be interested in the initial PO outcome  $p^4$ , because this point is not PO on the approximation.

Assuming that the PF approximation discussed above is IND, then we have the following: First of all, all points in the initial set of PO outcomes are now plausible decisions for a rational DM because  $P \subset PF(A)$  by Corollary 1. Second, since  $PF(A) = A$  by Theorem 2, the diversity of the PO objective vectors on the approximation can be examined directly from  $A$ . Since the DM is rational, then the diversity of  $PF(A)$ , not that of  $A$ , is interesting to him/her. If the PF approximation  $A$  is non-IND, these are not the same things.

We are left with the question of how to help the DM find a preferred point on a PF approximation. This is another instance where the concept of an IND approximation helps. According to Theorem 4, an IND approximation is the PF of a MOP that also has the set  $P$  on its PF.

**Theorem 4.** *If a set  $A \subset \mathbb{R}^k$  is an IND PF approximation based on  $P$ , then there exists a MOP with notation (1) so that  $P \subset PF(f(X))$  and  $PF(f(X)) = A$ .*

*Proof.* Assume that  $f = I$  (the identity mapping),  $X = A$  and  $n = k$  in (1). Then  $PF(f(X)) = PF(X) = PF(A) = A$ , where the last equation results from Theorem 2. Also by definition  $P \subset PF(f(X)) = A$ .  $\square$

The MOP, whose existence is guaranteed by Theorem 4, can be seen as a surrogate MOP for the initial MOP. We define this MOP for a general set, and not just for an IND approximation, and also give it a name.

**Definition 3.** For any nonempty set  $A \subset \mathbb{R}^k$ , the MOP

$$\begin{aligned} \min \quad & z \\ \text{s.t. } & z \in A \end{aligned} \quad (3)$$

is called the *MOP implied by A*. The formulation above is analogous to formulation (2) but instead of the set of outcomes  $f(X)$ , an arbitrary set  $A$  in the objective space  $\mathbb{R}^k$  is used.

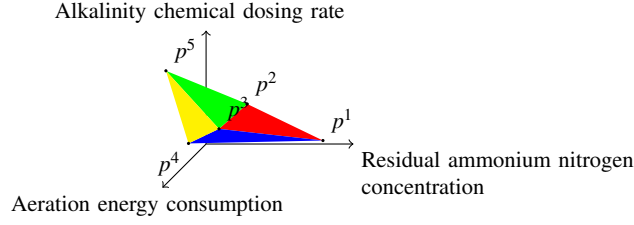
When trying to find a preferred point on an IND PF approximation, we thus propose to solve the MOP implied by  $A$  and take the final solution of the former as a preferred point on the approximation  $A$ . This approach has a couple of apparent benefits: All methods developed for solving MOPs are now applicable for decision making on the approximation and there is no need to develop new ones for this. Especially we can now use interactive MO methods in solving MOP (3). Also, since the PF of the MOP implied by the approximation is explicitly known, the computational expense should be very low while using MO methods in solving this MOP. Furthermore, since all the outcomes of the MOP are actually already on the PF of this MOP, there is no need to be concerned about the convergence to the PF, which also saves computational time. These aspects are clarified below with an example.

*Example 2.* In [6], a computationally expensive, black-box problem of wastewater treatment planning involving three objective functions is solved with the interactive multiobjective method NIMBUS [19, 17]. In the optimization process, 11 PO outcomes including the final preferred outcome  $(0.72, 332, 524)^T$  were computed. Here we want to produce a PF approximation in the neighborhood of the final outcome and then to formulate a MOP implied by this approximation in a form that could be solved with interactive MO methods. We use the already generated PO outcomes to construct this approximation but we want to emphasize that any appropriate method could have been used to generate the set  $P$ . We choose a subset of known PO outcomes (and thus approximate only a part of the PF) for the simplicity of the presentation.

The final outcome and four outcomes closest to it are chosen as the initial set of PO outcomes  $P$ . Thus, we have  $P = \{p^1, p^2, p^3, p^4, p^5\}$ , where  $p^1 = (1.7, 326, 506)$ ,  $p^2 = (1.1, 336, 515)$ ,  $p^3 = (0.9, 333, 519)$ ,  $p^4 = (0.7, 332, 524)$ ,  $p^5 = (0.5, 347, 528)$ . The set

$$A = \text{conv}(p^1, p^2, p^3) \cup \text{conv}(p^1, p^3, p^4) \cup \text{conv}(p^3, p^4, p^5) \cup \text{conv}(p^2, p^3, p^5)$$

shown in Figure 2 is an IND PF approximation and this can be checked analytically.

Fig. 2: The IND PF approximation  $A$ .

The IND PF approximation  $A$  can be parametrized with a slack variable  $t \in \{0, 1\}^3 \times [0, 1]^2$  and, using this parametrization, the MOP implied by  $A$  can be formulated as

$$\begin{aligned}
 & \min z(t) \\
 & \text{s.t. } z(t) = t_1(t_4 p^1 + t_5 p^2 + (1 - t_4 - t_5)p^3) \\
 & \quad + t_2(t_4 p^1 + t_5 p^3) + (1 - t_4 - t_5)p^4 \\
 & \quad + t_3(t_4 p^2 + t_5 p^3 + (1 - t_4 - t_5)p^5) \\
 & \quad + (1 - t_1 - t_2 - t_3)(t_4 p^3 + t_5 p^4 + (1 - t_4 - t_5)p^5) \\
 & \quad t = (t_1, \dots, t_5) \in \{0, 1\}^3 \times [0, 1]^2 \\
 & \quad 0 \leq 1 - t_1 - t_2 - t_3 \text{ and } 0 \leq 1 - t_4 - t_5.
 \end{aligned} \tag{4}$$

The MOP (4) includes 2 continuous and 3 discrete decision variables, 2 linear inequalities concerning them and also some box constraints. The decision variable  $t$  does not have any meaning to the DM, but is merely a product of the parametrization of  $A$ . However, the objective function values in the vector  $z(t) \in A$  have a meaning to the DM for each feasible  $t$  and based on these values he/she can express preferences. Exploration of the PF approximation  $A$  means varying the decision variable  $t$  based on the preferences of the DM and then showing him/her the resulted objective function values.

MOP (4) can be inputted e.g., to the WWW-NIMBUS implementation of the NIMBUS method [1]. Solving it is computationally inexpensive and the DM can get feedback to his/her preferences quickly unlike with the initial computationally expensive MOP. Thus, the DM could be more inclined to see more new outcomes and this should help him/her obtain a better understanding of the problem.

The ideas presented in this paper are of general nature, but can be seen to be related to some specific MO methods found in the literature. The Pareto Navigator method [4] can be interpreted as solving a MOP implied by the convex cone of the Pareto optimal solutions. The surface of this convex cone approximates the PF since for Pareto Navigator the problem is assumed to be convex. The methods of [15], on the other hand, can be seen as using a visual technique called the Interactive Decision Map for solving the MOP implied by an approximation (that is constructed by methods mentioned therein) of the feasible outcome set. However, since the ap-

proximations used in these methods are not necessarily IND approximations, the benefits mentioned in this section do not apply.

## 5 Conclusion

We have proposed a new approach to solving computationally expensive multiobjective optimization problems. In our approach, we first approximate the Pareto front and then enable the decision maker to explore the approximation with his/her preferences. We have also proposed *inherent nondominance*, a new property for Pareto front approximations which ensures desirable properties for approximations so that exploration can take place.

The utilization of proposed approximation approach requires an algorithm to construct an inherently nondominated Pareto front approximation from a given set of Pareto optimal outcomes. Furthermore, we must study which interactive multiobjective optimization methods are best suited for solving the multiobjective optimization problem implied by the approximation. These issues will be considered in future papers.

**Acknowledgements** The research was partly supported by the Academy of Finland, grant number 128495.

## References

1. WWW-NIMBUS System for Multiobjective Optimization, Last accessed: April 15, 2010. <http://nimbus.mit.jyu.fi/>.
2. H. Ackermann, A. Newman, H. Röglin, and B. Vöcking. Decision-making Based on Approximate and Smoothed Pareto Curves. *Theoretical Computer Science*, 378:253–270, 2007.
3. V. E. Bezerkin, G. K. Kamenev, and A. V. Lotov. Hybrid Adaptive Methods for Approximating a Nonconvex Multidimensional Pareto Frontier. *Computational Mathematics and Mathematical Physics*, 46:1918–1931, 2006.
4. P. Eskelinen, K. Miettinen, K. Klamroth, and J. Hakanen. Pareto Navigator for Interactive Nonlinear Multiobjective Optimization. *OR Spectrum*, 23:211–227, 2010.
5. T. Goel, R. Vaidyanathan, R. T. Haftka, W. Shyy, N. V. Queipo, and K. Tucker. Response Surface Approximation of Pareto Optimal Front in Multi-Objective Optimization. *Computer Methods in Applied Mechanics and Engineering*, 196:879–893, 2007.
6. J. Hakanen, K. Sahlsted, and K. Miettinen. Simulation-Based Interactive Multiobjective Optimization in Wastewater Treatment. In *Proceedings of the International Conference on Engineering Optimization EngOpt 2008*, Rio de Janeiro, Brazil, 2008. [http://www.engopt.org/08/nukleo/pdfs/0205\\_engopt\\_paper\\_hakanen.pdf](http://www.engopt.org/08/nukleo/pdfs/0205_engopt_paper_hakanen.pdf) [Last accessed: 19.11.2009].
7. C. Hwang and A. S. M. Masud. *Multiple Objective Decision Making - Methods and Applications: a State-of-the-Art Survey*. Springer, Berlin, 1979.
8. I. Kaliszewski. *Quantitative Pareto Analysis by Cone Separation Technique*. Kluwer Academic Publishers, Dordrecht, 1994.

9. M. K. Karakasis and K. C. Giannakoglou. On the Use of Surrogate Evaluation Models in Multi-Objective Evolutionary Algorithms. In P. Neittaanmäki, T. Rossi, S. Korotov, E. Onate, J. Periaux, and D. Knörzer, editors, *ECCOMAS 2004: Proceedings*, Jyväskylä, Finland, 2004. <http://www.mit.jyu.fi/eccomas2004/proceedings/pdf/616.pdf>, [Last accessed: 19.11.2009].
10. K. Klamroth and K. Miettinen. Integrating Approximation and Interactive Decision Making in Multicriteria Optimization. *Operations Research*, 56:222–234, 2008.
11. J. Knowles. ParEGO: A Hybrid Algorithm with On-Line Landscape Approximation for Expensive Multiobjective Optimization Problems. *IEEE Transactions on Evolutionary Computation*, 10:50 – 66, 2006.
12. P. Korhonen and J. Wallenius. A Pareto Race. *Naval Research Logistics*, 35:615–623, 1988.
13. O. I. Larichev. Cognitive Validity in Design of Decision-Aiding Techniques. *Journal of Multi-Criteria Decision Analysis*, 3:127 – 138, 1992.
14. W. Liu, Q. Zhang, E. Tsang, and B. Virginas. Tchebycheff Approximation in Gaussian Process Model Composition for Multi-Objective Expensive Black Box. In *2008 IEEE Congress on Evolutionary Computation, CEC 2008*, pages 3060–3065, Hong Kong, China, 2008.
15. A. V. Lotov, V. A. Bushenkov, and G. A. Kamenev. *Interactive Decision Maps*. Kluwer Academic Publishers, Boston, 2004.
16. J. Martin, C. Bielza, and D. R. Insua. Approximating Nondominated Sets in Continuous Multiobjective Optimization Problems. *Naval Research Logistics*, 52:469–480, 2005.
17. K. Miettinen. *Nonlinear Multiobjective Optimization*. Kluwer Academic Publishers, Boston, 1999.
18. K. Miettinen, A. V. Lotov, G. K. Kamenev, and V. E. Bezerkin. Integration of Two Multiobjective Optimization Methods for Nonlinear Problems. *Optimization Methods and Software*, 18:63–80, 2003.
19. K. Miettinen and M. M. Mäkelä. Synchronous Approach in Interactive Multiobjective Optimization. *European Journal of Operational Research*, 170:909–922, 2006.
20. K. Miettinen, F. Ruiz, and A. Wierzbicki. Introduction to Multiobjective Optimization: Interactive Approaches. In J. Branke, K. Deb, K. Miettinen, and R. Slowinski, editors, *Multiobjective Optimization: Interactive and Evolutionary Approaches*, pages 27–57. Springer, Berlin, 2008.
21. M. Monz, K. H. Kufer, T. R. Bortfeld, and C. Thieke. Pareto Navigation — Algorithmic Foundation of Interactive Multi-Criteria IMRT Planning. *Physics in Medicine and Biology*, 53:985–998, 2008.
22. S. Ruzika and M. M. Wiecek. Approximation Methods in Multiobjective Programming. *Journal of Optimization Theory and Applications*, 126:473–501, 2005.
23. Y. Sawaragi, H. Nakayama, and T. Tanino. *Theory of Multiobjective Optimization*. Academic Press, Orlando, 1985.
24. A. Wierzbicki. On the Completeness and Constructiveness of Parametric Characterizations to Vector Optimization Problems. *OR Spectrum*, 8:73–87, 1986.
25. B. Wilson, D. Cappelleri, T. W. Simpson, and M. Frecker. Efficient Pareto Frontier Exploration using Surrogate Approximations. *Optimization and Engineering*, 2:31–50, 2001.
26. M. Yoon, H. Nakayama, and Y. Yun. *Sequential Approximate Multiobjective Optimization Using Computational Intelligence*. Springer, Berlin Heidelberg, 2009.

**PII**

**CONSTRUCTING A PARETO FRONT APPROXIMATION FOR  
DECISION MAKING**

by

Markus Hartikainen · Kaisa Miettinen · Margaret M. Wiecek 2011

*Mathematical Methods of Operations Research* 73, pages 209–234

Reproduced with kind permission of Springer Science+Business Media.



# Constructing a Pareto Front Approximation for Decision Making

Markus Hartikainen · Kaisa Miettinen · Margaret M. Wiecek

Received: date / Accepted: date

**Abstract** An approach to constructing a Pareto front approximation to computationally expensive multiobjective optimization problems is developed. The approximation is constructed as a sub-complex of a Delaunay triangulation of a finite set of Pareto optimal outcomes to the problem. The approach is based on the concept of inherent nondominance. Rules for checking the inherent nondominance of complexes are developed and applying the rules is demonstrated with examples. The quality of the approximation is quantified with error estimates. Due to its properties, the Pareto front approximation works as a surrogate to the original problem for decision making with interactive methods.

**Keywords** Multiobjective Optimization · Multiple Criteria Decision Making · Pareto Optimality · Interactive Decision Making · Interpolation · Delaunay Triangulation

## 1 Introduction

In practical decision problems, there are often multiple conflicting objectives that need to be optimized at the same time (Keeney and Raiffa, 1993). These conflicting objectives are often handled with ad hoc aggregation or by converting all but one objective into constraints. Multiobjective optimization is a systematic approach to optimizing multiple conflicting objectives (see e.g., (Miettinen, 1999)).

Different multiobjective optimization methods can be classified into four classes with respect to the role of a decision maker (an expert in the application area who is at liberty to make decisions concerning the problem) in the optimization process

---

This research was partly supported by the Academy of Finland grant number 128495.

Markus Hartikainen · Kaisa Miettinen · Margaret M. Wiecek\*  
Department of Mathematical Information Technology,  
University of Jyväskylä, Finland

\*On sabbatical leave from Clemson University, Clemson, SC, USA

E-mail: markus.e.hartikainen@jyu.fi · kaisa.miettinen@jyu.fi · wmalgor@clemson.edu

(Miettinen, 1999; Sawaragi *et al.*, 1985). In no-preference methods, no preference information is used. In a priori methods, the problem is optimized with respect to a given preference information and then the decision maker is provided with a single solution. In a posteriori methods, the decision maker is provided with a set of solutions from which he/she can choose a preferred one. In interactive methods, an iterative procedure is used to explore different solutions to the problem and the decision maker is allowed guide the exploration. All these methods find their use in different situations, but we have concentrated on interactive methods because they allow the decision maker to learn about the problem while solving it (Miettinen *et al.*, 2008). In all methods, an important concept is Pareto optimality. A solution to a multiobjective optimization problem is Pareto optimal, if none of the objectives can be improved without impairing some other(s). An outcome is a vector containing the values of objectives as its components, and an outcome is Pareto optimal if it is given by a Pareto optimal solution. The preferences are in interactive methods typically (see (Luque *et al.*, 2011)) given in the form of desired values, changes or proportions of the values of objectives in Pareto optimal outcomes (see e.g., the NIMBUS method (Miettinen and Mäkelä, 1995, 2000, 2006) and the Tchebycheff method (Steuer and Choo, 1983; Steuer, 1986, 1989) to name some).

In addition to having conflicting objectives, many practical decision problems are also computationally expensive (see e.g., (Hasenjäger and Sendhoff, 2005; Laukkanen *et al.*, 2010)). The computational cost results from the lack of closed-form description of the objective functions and the need to employ simulation, which may take even hours (see e.g., (Hasenjäger and Sendhoff, 2005)). Computationally expensive problems are naturally very hard to solve and even interactive methods – that are often praised for their computational efficiency when compared to a posteriori methods (Miettinen, 1999) – may be inefficient. This is because the time grows between the moments when the decision maker expresses his/her preferences and when he/she can see the effects of those preferences in generated outcomes. The time grows because in most interactive methods the multiobjective optimization problem is in each iteration solved with respect to the updated preferences of the decision maker. This naturally slows down the solution process and makes it harder for the decision maker to learn about the problem and may even make him/her reluctant to explore different solutions to the problem.

We have developed an approach to help interactive methods deal with computationally expensive problems. The approach is based on replacing the original problem with a surrogate problem which approximates the set of Pareto optimal outcomes (often also called the Pareto front). Following this idea, we proposed in (Hartikainen *et al.*, 2011) that one should construct (as defined therein) an inherently nondominated Pareto front approximation based on a small set of known Pareto optimal outcomes and use this approximation with the interactive method instead of the original problem. The reasoning behind an inherently nondominated Pareto front approximation is that we construct a set in the space of outcomes that contains the known Pareto optimal outcomes as its subset. In this way, we can examine also other possible Pareto optimal outcomes that are not contained in the set of known Pareto optimal outcomes. In (Hartikainen *et al.*, 2011), we argued that an inherently nondominated Pareto front approximation does not include objective function values that would mislead the de-

---

cision maker. To be more specific, when approximating the Pareto front of a nonconvex multiobjective optimization problem, convex approximations (e.g., convex hulls or ellipses) may include vectors that are strictly better in all components than a known Pareto optimal outcome that was used to construct this approximation. These vectors are infeasible for the problem by definition of Pareto optimality. One of the benefits of an inherently nondominated approximation is that it avoids this behavior.

An inherently nondominated Pareto front approximation gives rise to a multiobjective optimization problem that can in a meaningful way be seen as a surrogate for the original problem. The Pareto front of the surrogate problem equals the inherently nondominated approximation and solving this problem interactively can be seen as exploring the approximation with interactive methods. This surrogate problem has a closed form formulation and, thus, it is computationally much less expensive than the original problem. The construction of the approximation unavoidably requires computation, but it can be done before involving the decision maker. After the approximation has been constructed, the decision maker can use the interactive method of his/her choice with the surrogate problem without computational delay. When a preferred outcome for the surrogate problem is found, it can then be projected onto the actual Pareto front of the original problem with e.g., an achievement scalarizing function (Wierzbicki, 1986).

Approaches similar to ours, where a Pareto front approximation is used in decision making, are given in (Eskelinen *et al.*, 2010; Lotov *et al.*, 2004; Monz, 2006). Eskelinen *et al.* (2010) and Monz (2006) use custom-made iterative procedures for finding a preferred element on the approximation, while we develop an approximation that can be used with almost any interactive method. The benefit of being able to use various interactive methods is that we can support diverse decision makers who may prefer different methods. In addition, approaches in (Eskelinen *et al.*, 2010; Monz, 2006) are only applicable to convex multiobjective optimization problems, while ours can handle both convex and nonconvex problems. The approach in (Lotov *et al.*, 2004) differs from ours, because it relies on a visualization technique called Interactive Decision Maps to find a preferred element on the approximation.

Methods for constructing Pareto front approximations (published before the year 2003) are surveyed in (Ruzika and Wiecek, 2005) and more recent ones include (Bezerkin *et al.*, 2006; Efremov and Kamenev, 2009; Goel *et al.*, 2007; Martin *et al.*, 2005). Some of these methods are not applicable for our needs, because they cannot create a Pareto front approximation for nonconvex multiobjective optimization problems. Other methods cannot operate based on a given set of Pareto optimal outcomes, but assume that some specific a posteriori method has been used to generate the outcomes. The value added by being able to use a given set of Pareto optimal outcomes is that one can use Pareto optimal outcomes generated with any a posteriori method (see e.g., (Miettinen, 1999)) including evolutionary multiobjective optimization methods (see e.g., (Deb, 2001; Coello Coello *et al.*, 2007)). Furthermore, none of the methods in the literature guarantee producing an inherently nondominated Pareto front approximation.

In this paper, we lay theoretical foundations for constructing an inherently nondominated Pareto front approximation for (convex or nonconvex) multiobjective optimization problems with continuous objectives. The proposed approximation im-

proves on the weaknesses of existing methods discussed above. Technically, our approximation is based on intelligent interpolation between a set of known Pareto optimal outcomes. The interpolants are taken from the polytopes of a Delaunay triangulation (see e.g., (Fortune, 1997)) of the set of known Pareto optimal outcomes. The idea for our approximation comes from computational geometry (for a general overview see e.g., (Goodman and O'Rourke, 1997)), where different subcomplexes of Delaunay triangulations have been extensively used in shape reconstruction (Edelsbrunner, 1998). We construct a subcomplex that satisfies some properties that are based on inherent nondominance so that the final approximation is inherently nondominated. The Delaunay triangulation is of combinatorial nature. Here we do not deal with the implementation of our approximation but concentrate on the theoretical aspects.

The rest of this paper is structured as follows. Section 2 concentrates on the notations and basic definitions used in this paper. The most relevant concepts from computational geometry are presented in Section 3 and combined with the inherent nondominance property in Section 4. Section 5 proposes an approach to constructing an inherently nondominated Pareto front approximation as a sub-complex of a Delaunay triangulation while Section 6 demonstrates the proposed approximation with two examples. Properties of the approximation for two special types of multiobjective optimization problems are examined in Section 7 and error estimates for the approximation are derived in Section 8. Finally, Section 9 concludes with some remarks on decision making with the constructed Pareto front approximation.

## 2 Notation and Definitions

We consider multiobjective optimization problems

$$\begin{aligned} \min_{x \in S} (f_1(x), \dots, f_k(x)), \end{aligned} \quad (1)$$

where  $f_i : S \rightarrow \mathbb{R}$  is for all  $i \in \{1, \dots, k\}$  a real-valued function and  $S \subset \mathbb{R}^n$ . We define a vector-valued function  $f : S \rightarrow \mathbb{R}^k$ ,  $f(x) = (f_1(x), \dots, f_k(x))^T$  for all  $x \in S$ . Because we aim to interpolate between the known Pareto optimal outcomes, we assume that all objectives are measured in continuous scales. This is the case for example when the set  $S$  is connected and the functions  $f_i$  are continuous.

The set  $S$  is called the feasible decision set of the multiobjective optimization problem and every vector  $x \in S$  is called a (feasible) decision. The set  $f(S)$  is the feasible outcome set of the multiobjective optimization problem and every vector  $z \in f(S)$  is called a (feasible) outcome.

For a set  $K \subset \mathbb{R}^k$  the boundary of  $K$  is denoted by  $\text{bnd}(K)$  and the interior is denoted by  $\text{int}(K)$ . The closure of the set  $K$  is denoted by  $\text{cl}(K)$ . The convex hull of  $K$  is denoted by  $\text{conv}(K)$ . For two sets  $K^1, K^2 \subset \mathbb{R}^k$  the notation  $K^1 \subset K^2$  means that if a vector  $s \in K^1$  then  $s \in K^2$ . Notation  $K^1 \subsetneq K^2$  means that  $K^1 \subset K^2$  but  $K^1 \neq K^2$ .

For two vectors  $z^1, z^2 \in \mathbb{R}^k$ , notation  $z^1 \leq z^2$  means that  $z_i^1 \leq z_i^2$  for all  $i = 1, \dots, k$  and  $z^1 \neq z^2$ . If  $z^1 \leq z^2$  then  $z^1$  is said to dominate  $z^2$  or  $z^2$  is said to be dominated by  $z^1$ . A vector  $b \in B$  is Pareto optimal in a set  $B \subset \mathbb{R}^k$  if there does not exist a vector  $b' \in B$  that dominates  $b$ . An outcome  $z \in f(S)$  is Pareto optimal for multiobjective

optimization problem (1) if the vector  $z$  is Pareto optimal in the set  $f(S)$ . Finally, the Pareto front of the set  $B$ , denoted by  $\text{PF}(B)$ , is the set of Pareto optimal vectors in the set  $B$  and the Pareto front of (1), denoted by  $\text{PF}$ , is the set of Pareto optimal outcomes of the multiobjective optimization problem. A feasible decision  $x \in S$  is a Pareto optimal solution to the multiobjective optimization problem if  $f(x)$  is a Pareto optimal outcome for the multiobjective optimization problem.

In (Hartikainen *et al.*, 2011), some properties and definitions concerning Pareto front approximations are given. They are summarized below.

**Definition 1** A set  $A \subset \mathbb{R}^k$  is *inherently nondominated* if there do not exist vectors  $a, b \in A$  so that  $a \leq b$ .

**Definition 2** Let  $P \subset \mathbb{R}^k$  be a finite set of Pareto optimal outcomes. A set  $A \subset \mathbb{R}^k$  is a *inherently nondominated Pareto front approximation (based on the set  $P$ )* if the set  $A$  is inherently nondominated and  $P \subset A$ .

**Definition 3** Let  $B \subset \mathbb{R}^k$  be a set and let the set  $P \subset \mathbb{R}^k$  be as above. A set  $A \subset B$  is a  *$B$ -maximal inherently nondominated Pareto front approximation (based on the set  $P$ )* if the set  $A$  is an inherently nondominated Pareto front approximation based on  $P$  and for all vectors  $b \in B \setminus A$  the set  $A \cup \{b\}$  is not inherently nondominated.

As shown in (Hartikainen *et al.*, 2011), an inherently nondominated Pareto front approximation based on  $P$  is the Pareto front of a multiobjective optimization problem that has outcomes  $P$  on its Pareto front. In this way, an inherently Pareto front approximation can be seen as a surrogate for the actual Pareto front.

The starting point of this study is as follows. We assume that we are given a computationally expensive multiobjective optimization problem. This problem is referred to as the *initial multiobjective optimization problem*. We also assume that we are given a finite set of  $m$  Pareto optimal outcomes  $P = \{p^1, \dots, p^m\}$ . This set is referred to as the *initial set of Pareto optimal outcomes* or the *known Pareto optimal outcomes*. This set may have been generated with any appropriate method in the literature e.g., an evolutionary multiobjective optimization method (Deb, 2001) or any of the  $0^{\text{th}}$ -order approximation methods mentioned in (Ruzika and Wiecek, 2005). In this paper, we use the initial set of Pareto optimal outcomes for constructing an inherently nondominated Pareto front approximation that intelligently interpolates between the known Pareto optimal outcomes.

### 3 Background on Polytopes, Complexes and (Delaunay) Triangulations

The Pareto front approximation constructed in this paper is a complex i.e., a collection of polytopes with certain properties. In the first part of this section, we review all the properties of polytopes and complexes that are needed in this paper. For the proofs and further properties we refer to (Grünbaum, 1967) and references therein. We mostly follow the notation and definitions established in (Grünbaum, 1967). The only difference is that here the prefix "a-" before the words "polytope", "face" or "complex" refers to the number of vertices minus one and not to the dimension. This

is because in our setting it is easier to count the number of vertices of the polytope than to compute the dimension of the polytope.

The rest of this section deals with triangulations of a finite vector set. A triangulation is a complex with certain additional properties. Especially, we discuss the Delaunay triangulation, which is a well-known triangulation (see e.g., (Edelsbrunner and Shah, 1994; Fortune, 1997)). Some reasons for the fame of the Delaunay triangulation are that it is useful in shape reconstruction, as argued in (Boissonnant, 1984), and it can be computed in  $\mathbb{R}^k$  by computing the convex hull of a set in  $\mathbb{R}^{k+1}$ , as shown in (Edelsbrunner, 1987). In our approximation, the complex that approximates the Pareto front is a sub-complex of a Delaunay triangulation of the initial set of Pareto optimal outcomes  $P$ . Previously Delaunay triangulations have been used mostly in finite element methods (see e.g., (George and Borouchaki, 1998; Shenton and Cendes, 1985)) and for reconstructing different solids from a set of vectors (see e.g., (Edelsbrunner, 1998)). In (Schandl *et al.*, 2002), the Delaunay triangulation is used for initializing an algorithm for Pareto front approximation. Their aim is to construct a piecewise linear approximation by means of block norms whose level sets provide a polyhedral structure of the approximation. However, the resulting approximation has features different from ours and does not allow the decision maker to use interactive multiobjective optimization methods.

An alternative comprehensive treatment of the topics contained in this section can be found in (Fukuda, 2004). That article also contains references to more detailed treatments of these and further topics.

**Definition 4** Let  $a \in \mathbb{N}$ . The set

$$\mathcal{P}(z^1, \dots, z^{a+1}) = \text{conv}(\{z^1, \dots, z^{a+1}\})$$

with vectors  $z^1, \dots, z^{a+1} \in \mathbb{R}^k$  is called a (convex  $a$ -)polytope. The empty set is a  $(-1)$ -polytope. The polytope  $\mathcal{P}(z^1, \dots, z^{a+1})$  is said to be determined by vectors  $z^1, \dots, z^{a+1}$ .

The following propositions and definitions summarize the properties of polytopes. The proofs can be found in (Grünbaum, 1967).

**Proposition 1** Let  $a, b \in \mathbb{N}$  and consider vectors  $z^1, \dots, z^{a+1}, \hat{z}^1, \dots, \hat{z}^{b+1} \in \mathbb{R}^k$ . If  $\{z^1, \dots, z^{a+1}\} \subset \{\hat{z}^1, \dots, \hat{z}^{b+1}\}$ , then  $\mathcal{P}(z^1, \dots, z^{a+1}) \subset \mathcal{P}(\hat{z}^1, \dots, \hat{z}^{b+1})$ .

**Definition 5** A vertex of a polytope is a vector  $x \in K$  so that if  $y, z \in K$ ,  $0 < \lambda < 1$  and  $x = \lambda y + (1 - \lambda)z$ , then  $x = y = z$ . The set of all vertices is denoted by  $\text{vert}(K)$ .

**Definition 6** Let  $u \in \mathbb{R}^k$  be a vector,  $\alpha \in \mathbb{R}$  be a scalar and  $K \subset \mathbb{R}^k$  be a polytope. The hyperplane

$$H = \{x \in \mathbb{R}^k : x^T u = \alpha\}$$

is a supporting hyperplane of the polytope  $K$  if  $K \cap H \neq \emptyset$  and either  $K \subset \{x \in \mathbb{R}^k : x^T u \leq \alpha\}$  or  $K \subset \{x \in \mathbb{R}^k : x^T u \geq \alpha\}$ .

**Definition 7** Let  $K \subset \mathbb{R}^k$  be a polytope. A set  $F \subset \mathbb{R}^k$  is called a face of the polytope  $K$  if  $F = \emptyset$ ,  $F = K$  or if there exists a supporting hyperplane  $H$  of the polytope  $K$  so that  $F = K \cap H$ . The sets  $\emptyset$  and  $K$  are called improper faces of the polytope  $K$  and the other faces are called proper faces of  $K$ . A face  $F$  that is also an  $a$ -polytope is called an  $a$ -face.

*Example 1* If  $K = \mathcal{P}(z^1, z^2) \subset \mathbb{R}^3$  is a 1-polytope with  $z^1 \neq z^2$ , then the set of vertices is  $\text{vert}(K) = \{z^1, z^2\}$  and the faces are  $\emptyset$ ,  $\{z^1\}$ ,  $\{z^2\}$  and  $\mathcal{P}(z^1, z^2)$ .

**Proposition 2** *A face of a polytope is also a polytope.*

**Proposition 3** *If  $K \subset \mathbb{R}^k$  is a polytope, then the boundary of the polytope is  $\text{bnd}(K) = \cup_{F \in \mathcal{F}} F$ , where  $\mathcal{F}$  is a collection containing all the proper faces of  $K$ .*

**Proposition 4** *Let  $K^1, K^2, K^3 \subset \mathbb{R}^k$  be polytopes so that the polytope  $K^2$  is a face of the polytope  $K^1$  and the polytope  $K^3$  is a face of the polytope  $K^2$ . Then the polytope  $K^3$  is a face of the polytope  $K^1$ .*

**Definition 8** The dimension of a polytope  $K \subset \mathbb{R}^k$  is the minimal dimension of a linear subspace  $L \subset \mathbb{R}^k$  so that  $K \subset z + L$  with some  $z \in \mathbb{R}^k$ .

**Proposition 5** *The dimension of an  $a$ -polytope  $K \subset \mathbb{R}^k$  is at most  $a$ .*

A simplex is a special type of polytope. In a way, a simplex can be seen as the simplest type of all polytopes, as argued in (Grünbaum, 1967). In order to define a simplex, we need to define affine independence of vectors in  $\mathbb{R}^k$ .

**Definition 9** Vectors  $v^1, \dots, v^s \in \mathbb{R}^k$  are affinely independent if the vectors  $v^2 - v^1, \dots, v^s - v^1$  are linearly independent.

**Definition 10** An  $a$ -polytope  $K = \mathcal{P}(z^1, \dots, z^{a+1})$  is called an  $a$ -simplex (or merely a simplex) if the vectors  $z^1, \dots, z^{a+1}$  are affinely independent.

By Proposition 5, the dimension of an  $a$ -polytope is at most  $a$ . We can say even more about simplices.

**Proposition 6** *The dimension of an  $a$ -simplex is  $a$ .*

Another important property of a simplex is that its faces are also simplices and that any proper subset of vertices determines a proper face of the simplex. Notice that the latter is not true all for polytopes, because e.g., the vertices  $(0, 0), (1, 1)$  of a polytope  $\mathcal{P}((0, 0), (1, 0), (0, 1), (1, 1)) \subset \mathbb{R}^2$  do not determine a face of the polytope.

**Proposition 7** *Let  $a, b \in \mathbb{N}$  so that  $a \geq b$ . All the  $b$ -faces of an  $a$ -simplex are  $b$ -simplices and any  $b + 1$  vertices of a simplex determine a  $b$ -face of the simplex.*

A complex is a collection of polytopes with certain properties. The complex is used to describe polyhedral sets beyond polytopes.

**Definition 11** A collection  $\mathcal{K}$  of polytopes in  $\mathbb{R}^k$  is called a (polyhedral) complex provided that

- (1) every face of a polytope in  $\mathcal{K}$  is also a polytope in  $\mathcal{K}$  and
- (2) the intersection of any two polytopes in  $\mathcal{K}$  is a face of each of them.

A complex  $\mathcal{K}$  is called an  $a$ -complex, if there exists an  $a$ -polytope  $K \in \mathcal{K}$  and there is no  $b$ -polytope  $K' \in \mathcal{K}$  with  $b > a$ .

The body of a complex contains all the vectors in all the polytopes in the complex. The body is sometimes also called the underlying space (see e.g., (Vegter, 1997)).

**Definition 12** The body of a complex  $\mathcal{K}$  is  $\text{body}(\mathcal{K}) = \cup_{K \in \mathcal{K}} K$ .

Now we are ready to define a triangulation of a finite set  $P \subset \mathbb{R}^k$ . Informally, the triangulation of a set can be seen as a subdivision of the convex hull of the set.

**Definition 13** A triangulation of a finite set  $P \subset \mathbb{R}^k$  is a complex  $\mathcal{K}$  so that the body of the complex is  $\text{body}(\mathcal{K}) = \text{conv}(P)$  and the set of vertices of the polytopes in the collection  $\mathcal{K}$  is the set  $P$ .

In this paper, we use the Delaunay triangulation. There are different definitions for the Delaunay triangulation. Following (Edelsbrunner and Shah, 1994), we use the following definition.

**Definition 14** A triangulation  $\mathcal{D}$  of a set  $P$  is a Delaunay triangulation if for every polytope  $\mathcal{P}(p^1, \dots, p^{a+1}) \in \mathcal{D}$  there exists an open ball  $B$  so that  $B \cap P = \emptyset$  and  $\text{cl}(B) \cap P = \{p^1, \dots, p^{a+1}\}$ .

In an informal fashion, one may say that the Delaunay triangulation contains all the polytopes defined by neighboring vectors. This is formalized by the link between the Delaunay triangulation and the Voronoi diagram (see e.g., (Edelsbrunner, 1987)).

An important concept concerning Delaunay triangulations and their construction is so-called general position of vectors (see e.g., (Rajan, 1994)). In this paper we do not assume general position, because often it is not valid for outcomes of a multi-objective optimization problem. See e.g., the first example in Section 6, where the outcomes in  $P$  are not in general position.

Delaunay triangulations can be constructed with methods proposed in many papers e.g., (Edelsbrunner, 1987; Rajan, 1994). For this reason this is not a topic of this paper, but we assume that some existing method has been used to construct a Delaunay triangulation of the initial set of Pareto optimal outcomes  $P$ .

In this section, we have introduced the main concepts from computational geometry that are used in this paper. In the next section, we further develop these concepts for our purposes.

#### 4 Inherently Nondominated Polytopes and Complexes

In this section, we discuss inherently nondominated polytopes and complexes. The definition of an inherently nondominated polytope follows the original definition of an inherently nondominated set established in Definition 1 and in Definition 15 inherent nondominance is generalized for complexes.



According to Proposition 5, the dimension of an  $a$ -polytope is at most  $a$ . According to the following theorem, irrespective of the number of vertices, the dimension of an inherently nondominated polytope is at most the number of objectives minus one.

**Theorem 1** *The dimension of an inherently nondominated polytope  $K \subset \mathbb{R}^k$  is at most  $k - 1$ .*

*Proof* Assume that  $K = \mathcal{P}(z^1, \dots, z^{a+1})$  with  $a \geq k$ , because if it held that  $a < k$  then the claim would be trivially true by Proposition 5. It can be seen that  $K \subset z^1 + L$  with  $L = \{l \in \mathbb{R}^k : l = \sum_{i=2}^{a+1} \lambda^i (z^i - z^1), \lambda^i \in \mathbb{R} \text{ for all } i = 2, \dots, a+1\}$ , because every vector  $s \in K$  can be written as  $s = z^1 + \sum_{i=2}^{a+1} \lambda^i (z^i - z^1)$  with  $\lambda^i \in \mathbb{R}$  for all  $i = 2, \dots, a+1$ . Thus, it remains to prove that  $\dim(L) \leq k - 1$ . To prove this assume the contrary i.e.,  $\dim(L) = k$ . If  $\dim(L) = k$ , then there are linearly independent vectors  $v^1, \dots, v^k \in \{z^2 - z^1, \dots, z^{a+1} - z^1\}$ . Choose  $z = \sum_{i=1}^k (1/k)v^i$ . By Propositions 7 and 3 and because the polytope  $\mathcal{P}(v^1, \dots, v^k)$  is now a simplex, it holds that  $z \in \text{int}(\mathcal{P}(v^1, \dots, v^k))$  and by Proposition 1 it holds that  $\text{int}(\mathcal{P}(v^1, \dots, v^k)) \subset \text{int}(\mathcal{P}(z^2 - z^1, \dots, z^{a+1} - z^1))$ . Therefore,  $z^1 + z \in \text{int}(K)$  and there exists a real number  $r > 0$  so that  $B(z^1 + z, r) \subset K$ , where  $B(z^1 + z, r)$  is a ball with radius  $r$  centered at  $z^1 + z$ , and, especially, it holds that  $z^1 + z - (r/2, \dots, r/2) \in K$ . This is a contradiction with the fact that the polytope  $K$  is inherently nondominated, since it holds that  $z^1 + z - (r/2, \dots, r/2) < z^1 + z$ . This completes the proof.  $\square$

**Corollary 1** *A  $k$ -simplex is not inherently nondominated.*

*Proof* This follows from Proposition 6 and Theorem 1.  $\square$

An inherently nondominated polytope is always a union of polytopes with fewer vertices than objectives. This is given by the following theorem.

**Theorem 2** *If a polytope  $K = \mathcal{P}(z^1, \dots, z^{a+1}) \subset \mathbb{R}^k$  is inherently nondominated, then there exist  $a_i$ -polytopes  $K^i \subset \mathbb{R}^k$ ,  $i = 1, \dots, t$ , so that  $K = \cup_{i=1}^t K^i$  and  $a_i \leq k - 1$  for every  $i = 1, \dots, t$ .*

*Proof* Let a vector  $s \in K$  be arbitrary. By Carathéodory's theorem (Caratheodory, 1913), the vector  $s$  belongs to some  $k$ -polytope  $K' = \mathcal{P}(v^1, \dots, v^{k+1})$  with  $v^1, \dots, v^{k+1} \in \{z^1, \dots, z^{a+1}\}$ . Assume that  $s = \sum_{i=1}^{k+1} \lambda^i v^i \in K$  for some  $\lambda^i \in [0, 1]$ . By the proof of Theorem 1, there exists a linear subspace  $L$  of  $\mathbb{R}^k$  so that  $\dim(L) \leq k - 1$  and  $K' - v^1 \subset L$ . Thus, there exist real numbers  $\mu^2, \dots, \mu^{k+1} \in \mathbb{R}$  so that  $\sum_{i=2}^{k+1} \mu^i (v^i - v^1) = 0$  and  $\mu^j \neq 0$  for some  $j \in \{2, \dots, k+1\}$ . Thus by choosing  $\mu^1 = -\sum_{i=2}^{k+1} \mu^i$ , it holds that  $\sum_{i=1}^{k+1} \mu^i v^i = 0$  and  $\sum_{i=1}^{k+1} \mu^i = 0$ . Therefore, for all  $\alpha \in \mathbb{R}$  it holds that

$$s = \sum_{i=1}^{k+1} \lambda^i v^i - \alpha \sum_{i=1}^{k+1} \mu^i v^i = \sum_{i=1}^{k+1} (\lambda^i - \alpha \mu^i) v^i$$

and  $\sum_{i=1}^{k+1} \lambda^i - \alpha \mu^i = 1$  Especially by choosing

$$\alpha = \min_{\substack{i=1, \dots, k+1 \\ \mu^i \neq 0}} \lambda^i / \mu^i$$

it holds in addition that  $\lambda^i - \alpha\mu^i \geq 0$  for all  $i = 1, \dots, k+1$  and  $\lambda^i - \alpha\mu^i = 0$  for some  $i \in \{1, \dots, k+1\}$ . Thus,  $s$  is presented as a convex combination of  $k$  vectors and, thus,  $s$  belongs to some  $b$ -polytope with  $b \leq k-1$ . It follows that  $K \subset \cup_{i=1}^t K^i$  for some  $a^i$ -polytopes  $K^i$  with  $a^i \leq k-1$  for all  $i = 1, \dots, t$ . From the construction it is seen that we have for each polytope  $K^i = \mathcal{P}(u^1, \dots, u^{b+1})$  with  $\{u^1, \dots, u^{b+1}\} \subset \{z^1, \dots, z^{a+1}\}$ . Thus, also  $K \supset \cup_{i=1}^t K^i$  by Proposition 1. This proves the claim.  $\square$

The inherent nondominance was defined in (Hartikainen *et al.*, 2011) for sets and not for collections of polytopes that we need in this paper. For this reason, we need to extend that definition for our needs.

**Definition 15** A collection of polytopes  $\mathcal{K}$  is said to be *inherently nondominated* if the set  $\cup_{K \in \mathcal{K}} K$  is inherently nondominated. Especially, a complex is inherently nondominated if its body is inherently nondominated.

Finally, we have extensions to Definitions 2 and 3. These are the main concepts in this paper.

**Definition 16** A collection of polytopes  $\mathcal{K}$  is said to be an *inherently nondominated Pareto front approximation* if the collection of polytopes is inherently nondominated and the singleton  $\{p\} \in \mathcal{K}$  for all  $p \in P$ .

Analogously to what is noticed in (Hartikainen *et al.*, 2011), the complex  $\{\emptyset\} \cup \{\{p\}, p \in P\}$  is by definition an inherently nondominated Pareto front approximation. However, it is not a good approximation because it does include any intermediate outcomes but only the initial Pareto optimal outcomes in  $P$ . This is why the following definition is important. The collection  $\mathcal{K}$  in the following definition is further on chosen as a Delaunay triangulation of the initial set of Pareto optimal outcomes  $P$ .

**Definition 17** Let  $\mathcal{K}$  be a collection of polytopes. A collection of polytopes  $\mathcal{A}$  is said to be a  $\mathcal{K}$ -*maximal inherently nondominated Pareto front approximation* if the collection  $\mathcal{A}$  is an inherently nondominated Pareto front approximation and for all polytopes  $K \in \mathcal{K} \setminus \mathcal{A}$  the collection  $\mathcal{A} \cup \{K\}$  is not inherently nondominated.

An important property is that a  $\mathcal{K}$ -maximal inherently nondominated Pareto front approximation is a complex whenever  $\mathcal{K}$  is. This is given by the following theorem.

**Theorem 3** *Let  $\mathcal{K}$  be a complex so that singleton  $\{p\} \in \mathcal{K}$  for all  $p \in P$ . If a collection of polytopes  $\mathcal{A}$  is a  $\mathcal{K}$ -maximal inherently nondominated Pareto front approximation, then the collection  $\mathcal{A} \subset \mathcal{K}$  is a complex.*

*Proof* The collection  $\mathcal{A}$  is not empty, because the collection of sets  $\{\emptyset\} \cup \{\{p\} : p \in P\}$  is an inherently nondominated Pareto front approximation. Assume a polytope  $K \in \mathcal{A}$  and let  $F$  be a face of the polytope  $K$ . Then two observations are immediate: (i) the face  $F \in \mathcal{K}$ , because the collection  $\mathcal{K}$  is a complex, and (ii) the collection  $\mathcal{A} \cup \{F\}$  is inherently nondominated, because its body is  $\text{body}(\mathcal{A} \cup \{F\}) = \text{body}(\mathcal{A})$  since by Definition 7 face  $F \subset K$ . Observations (i) and (ii) yield that also the face  $F \in \mathcal{A}$  which implies property (1) in Definition 11. Property (2) is trivially true, because  $\mathcal{A} \subset \mathcal{K}$ .  $\square$

Finally, we have the following theorem implying the existence of at least one  $\mathcal{H}$ -maximal inherently nondominated Pareto front approximation for an appropriate complex  $\mathcal{H}$ . The following theorem combined with Theorem 3 yields that there exists a complex  $\mathcal{A} \subset \mathcal{H}$  that is a  $\mathcal{H}$ -maximal inherently nondominated Pareto front approximation. In multiobjective optimization problems with more than two objectives, the maximal inherently nondominated Pareto front approximation may not be unique. For the proof of existence we need the Zorn's lemma.

**Lemma 1 (Zorn's lemma)** *Every partially ordered set, in which every chain (i.e. totally ordered subset) has an upper bound, contains at least one maximal element.*

**Theorem 4** *Let  $\mathcal{H}$  be a complex so that  $\{p\} \in \mathcal{H}$  for all  $p \in P$ . Then there exists a  $\mathcal{H}$ -maximal inherently nondominated Pareto front approximation.*

*Proof* Define a family of collections of polytopes

$$\mathbb{K} = \{\mathcal{B} : \mathcal{B} \subset \mathcal{H} \text{ is an inherently nondominated Pareto front approximation.}\}.$$

The collection  $\{\{p\} : p \in P\} \in \mathbb{K}$  and thus  $\mathbb{K}$  is not an empty family. Relation  $\subset$  is a partial ordering in the family of collections  $\mathbb{K}$ . Let collections  $\mathcal{B}^1, \mathcal{B}^2, \dots \in \mathbb{K}$  be a totally ordered chain so that  $\mathcal{B}^{i-1} \subset \mathcal{B}^i$  for all  $i = 2, 3, \dots$ . Each collection  $\mathcal{B}^i$  is then especially an inherently nondominated Pareto front approximation. Define a collection  $\overline{\mathcal{B}} = \bigcup_{i=1}^{\infty} \mathcal{B}^i$ . It is clear that the collection  $\overline{\mathcal{B}}$  is an upper bound for the chain  $\mathcal{B}^1, \mathcal{B}^2, \dots$ . In order to use Zorn's lemma we prove that the collection  $\overline{\mathcal{B}} \in \mathbb{K}$ . Trivially, the singleton  $\{p\} \in \overline{\mathcal{B}}$  for all  $p \in P$ , because  $\{p\} \in \mathcal{B}^1$  by definition. Assume now that the collection  $\overline{\mathcal{B}}$  is not inherently nondominated. Then there exist vectors  $a^1, a^2 \in \bigcup_{K \in \overline{\mathcal{B}}} K$  so that  $a^1 \leq a^2$ . But then  $a^1 \in \bigcup_{K \in \mathcal{B}^{i^1}} K$  and  $a^2 \in \bigcup_{K \in \mathcal{B}^{i^2}} K$  for some  $i^1, i^2 \in \mathbb{N}$  and, moreover, the vectors  $a^1, a^2 \in \bigcup_{K \in \mathcal{B}^{\max\{i^1, i^2\}}} K$ . This implies that the collection  $\mathcal{B}^{\max\{i^1, i^2\}}$  is not inherently nondominated, which is a contradiction with the fact that the collection  $\mathcal{B}^{\max\{i^1, i^2\}} \in \mathbb{K}$ . Since all the assumptions of the Zorn's lemma have been fulfilled, there exists a maximal element within  $\mathbb{K}$ . Trivially, this maximal element is a  $\mathcal{H}$ -maximal inherently nondominated Pareto front approximation.  $\square$

## 5 The Construction of a Maximal Inherently Nondominated Pareto Front Approximation

Naturally, the existence of a maximal inherently nondominated Pareto front approximation given by Theorem 4 is not sufficient for practical uses. In this section, we elaborate on how to construct a sub-complex of a Delaunay triangulation  $\mathcal{D}$  of the initial set of Pareto optimal outcomes  $P$  that is a  $\mathcal{D}$ -maximal inherently nondominated Pareto front approximation. Because of Corollary 1, we however exclude the  $k$ -simplices, with  $k$  being the number of objectives of the initial multiobjective optimization problem, from the complex  $\mathcal{D}$ . Without confusion, we use the same symbol  $\mathcal{D}$  for this sub-complex of the Delaunay triangulation. Even if Theorem 2 implies that every inherently nondominated polytope can be represented as a union of polytopes

with the number of vertices being lower than  $k$ , these polytopes may not be included in the Delaunay triangulation of  $P$ .

We first develop methods for checking whether or not a collection of polytopes is inherently nondominated. In Subsection 5.1, we show that checking this amounts to checking all pairs of polytopes in the collection for a certain property. In Subsection 5.2, we show how to use the concepts developed in Subsection 5.1 for constructing a  $\mathcal{D}$ -maximal inherently nondominated Pareto front approximation.

### 5.1 Checking if a Collection of Polytopes is Inherently Nondominated

In this subsection, we show an efficient way of checking whether a collection of polytopes, or a complex, is inherently nondominated. We first define what is meant by dominance between polytopes.

**Definition 18** Let  $K^1, K^2 \subset \mathbb{R}^k$  be polytopes. The polytope  $K^1$  is said to dominate (to be dominated by) the polytope  $K^2$ , if there exist vectors  $s^1 \in K^1$  and  $s^2 \in K^2$  so that  $s^1 \leq s^2$  ( $s^2 \leq s^1$ ). If the 0-polytope  $\mathcal{P}(z) = \{z\}$  dominates (is dominated by) a polytope  $K \subset \mathbb{R}^k$  then the vector  $z$  is said to dominate (be dominated by) the polytope  $K$ .

The connection between inherent nondominance and dominance between sets is given by the two following theorems. Theorem 5 deals with the inherent nondominance of a polytope and Theorem 6 deals with the inherent nondominance of a collection of polytopes.

**Theorem 5** *A polytope is inherently nondominated if and only if the polytope does not dominate itself.*

*Proof* The proof follows directly from Definitions 1 and 18. □

**Theorem 6** *Let  $\mathcal{K}$  be a collection of polytopes. The collection  $\mathcal{K}$  is inherently nondominated if and only if there does not exist polytopes  $K^1, K^2 \in \mathcal{K}$  so that  $K^1$  dominates  $K^2$ .*

*Proof* " $\Rightarrow$ ": Assume the contrary, i.e., there exist polytopes  $K^1, K^2 \in \mathcal{K}$  so that the polytope  $K^1$  dominates the polytope  $K^2$ . This means that there exists a vector  $s^1 \in K^1$  that dominates another vector  $s^2 \in K^2$ . This is a contradiction with the assumption that the collection of polytopes  $\mathcal{K}$  is inherently nondominated, since the vectors  $s^1, s^2 \in \text{body}(\mathcal{K})$ .

" $\Leftarrow$ ": Assume that the collection of polytopes  $\mathcal{K}$  is not inherently nondominated. Then there exist vectors  $s^1, s^2 \in \text{body}(\mathcal{K})$  so that the vector  $s^1$  dominates the vector  $s^2$ . This means that there exist polytopes  $K^1, K^2 \in \mathcal{K}$  so that the vector  $s^1 \in K^1$  and the vector  $s^2 \in K^2$ . But then, by Definition 18, the polytope  $K^1$  dominates the polytope  $K^2$  which is a contradiction with the assumption. □

As shown by Theorems 5 and 6, checking whether a polytope or a collection of polytopes is inherently nondominated can be done by checking for dominance between polytopes as defined in Definition 18. For this reason we need an efficient way

of checking for dominance between polytopes. According to the following theorem, the dominance between polytopes can be checked by solving optimization problems.

**Theorem 7** *Let  $K^1, K^2 \subset \mathbb{R}^k$  be non-empty polytopes. Consider optimization problems*

$$\begin{aligned} \min & \max_{i=1, \dots, k} (s_i^1 - s_i^2) \\ \text{s.t. } & s^1 \in K^1, s^2 \in K^2 \end{aligned} \quad (2)$$

and

$$\begin{aligned} \min & \sum_{i=1}^k (s_i^1 - s_i^2) \\ \text{s.t. } & s^1 \in K^1, s^2 \in K^2 \\ & s_i^1 \leq s_i^2 \text{ for all } i = 1, \dots, k. \end{aligned} \quad (3)$$

The polytope  $K^1$  dominates the polytope  $K^2$  if and only if one of the following holds:

- (i) the optimal value in problem (2) is less than zero
- OR
- (ii) the optimal value in problem (2) is exactly zero and the optimal value in problem (3) is less than zero.

*Proof* Notice that problem (2) has a solution, because the outer objective function  $g(s^1, s^2) := \max_{i=1, \dots, k} (s_i^1 - s_i^2)$  is continuous and the feasible set is compact and non-empty. Furthermore, problem (3) has a solution, if the optimal value in problem (2) is zero, because the objective function is continuous and the feasible set is compact and non-empty.

By Definition 18, if the polytope  $K^1$  dominates the polytope  $K^2$  then there exist vectors  $s^1 \in K^1$  and  $s^2 \in K^2$  so that  $s^1$  dominates  $s^2$ . The dominance can either be strong or weak i.e., either

$$(1) s_i^1 < s_i^2 \text{ for all } i = 1, \dots, k$$

OR

$$(2) s_i^1 \leq s_i^2 \text{ for all } i = 1, \dots, k \text{ and } s_j^1 < s_j^2 \text{ for some index } j \in \{1, \dots, k\}.$$

Clearly, alternative (i) holds if and only if alternative (1) holds. Assume now that alternative (2) holds but alternative (i) does not hold. Then the optimal value in problem (2) is equal to zero, because if it was more than zero alternative (2) could not hold, and the optimal value in problem (3) is less than zero, because  $\sum_{i=1}^k (s_i^1 - s_i^2) < s_j^1 - s_j^2$  for all  $j \in \{1, \dots, k\}$ . Finally, if alternative (ii) holds, then there exist vectors  $s^1 \in K^1$  and  $s^2 \in K^2$  so that  $s_i^1 \leq s_i^2$  for all  $i = 1, \dots, k$  and  $\sum_{i=1}^k (s_i^1 - s_i^2) < 0$ . But if  $\sum_{i=1}^k (s_i^1 - s_i^2) < 0$ , then there exists an index  $j \in \{1, \dots, k\}$  so that  $s_j^1 < s_j^2$  and thus alternative (2) must hold. This completes the proof.  $\square$

## 5.2 Removal of Polytopes from the Delaunay Triangulation

Assume that we have constructed the Delaunay triangulation  $\mathcal{D}$  of the initial set of Pareto optimal outcomes  $P \subset \mathbb{R}^k$  and that we have excluded the  $k$ -simplices from it. In this subsection, we propose an approach to finding a collection of polytopes  $\mathcal{R} \subset \mathcal{D}$  so that the collection of polytopes

$$\mathcal{D} \setminus \mathcal{R} \quad (4)$$

is a  $\mathcal{D}$ -maximal inherently nondominated Pareto front approximation based on the initial set of Pareto optimal outcomes  $P$ . In other words, we propose ways to remove polytopes from the Delaunay triangulation to get the approximation.

Without loss of generality we may assume that for all  $a = 0, \dots, m-1$ , the  $a$ -polytopes in the complex  $\mathcal{D}$  are  $K^{a,1}, \dots, K^{a,t^a}$ . For all  $a = m-1, m-2, \dots, 1$  and for all  $j = 1, \dots, t^a$ , it holds that the polytope  $K^{a,j} \in \mathcal{R}$  i.e., the polytope is removed, if either

- (R1) there exists an outcome  $p \in P$  that dominates or is dominated by the polytope  $K^{a,j}$  or the polytope  $K^{a,j}$  dominates itself
- OR
- (R2) there exists a  $b$ -polytope  $K^{b,j'} \in \mathcal{D} \setminus \mathcal{R}$  with either  $b > a$  or  $b = a$  and  $j' < j$  that dominates or is dominated by the polytope  $K^{a,j}$ .

Rule (R1) dictates that a polytope is removed if it either dominates or is dominated by some of the Pareto optimal outcomes in  $P$  or if the polytope is not inherently nondominated. There may be multiple  $\mathcal{D}$ -maximal inherently nondominated Pareto front approximations that can be constructed from polytopes that are not removed by rule (R1) and rule (R2) dictates which of them we choose. According to it, the polytopes with more vertices are less likely to be removed from the collection because whenever we choose not to remove a polytope, then the polytopes with less vertices are removed if they dominate or are dominated by this polytope.

**Theorem 8** *If the collection  $\mathcal{R}$  is given by rules (R1) and (R2) then the collection of polytopes given by (4) is a  $\mathcal{D}$ -maximal inherently nondominated Pareto front approximation.*

*Proof* The polytopes in the complex  $\mathcal{D}$  can be ordered as in rule (R2) with respect to the number of vertices  $a$  and the index  $j$ . According to rule (R2) a polytope with at least two vertices is removed from the complex if there exists a polytope prescribed by this rule that is either dominated or dominates this polytope. For this reason there cannot be distinct polytopes  $K^i, K^j \in \mathcal{D} \setminus \mathcal{R}$  with more vertices than 2 that dominate each other. The first part of rule (R1) dictates that there cannot be a polytope in the collection that dominates or is dominated by an outcome in  $P$  and, thus, there are no distinct polytopes in the collection that dominate each other. According to the second part of rule (R1), no polytope in  $\mathcal{D} \setminus \mathcal{R}$  is dominated by it self. According to Theorem 7, the collection of polytopes  $\mathcal{D} \setminus \mathcal{R}$  is inherently nondominated and since the singleton  $\{p\} \in \mathcal{D} \setminus \mathcal{R}$  for all  $p \in P$ , the collection of polytopes is an inherently nondominated Pareto front approximation. Trivially, the Pareto front approximation is  $\mathcal{D}$ -maximal.  $\square$

According to (McMullen, 1970), a Delaunay triangulation of the set  $P$  contains at most  $O(m^{\lceil k/2 \rceil})$  polytopes, where  $m$  is the number of outcomes in  $P$  and  $k$  is the number of objectives of the multiobjective optimization problem. For this reason, one may have to solve  $O(m^k)$  optimization problems in order to find out all dominations between the polytopes in the triangulation with problems (2) and (3). This is computationally the most expensive part of the approximation approach. Methods for reducing this computational expense go beyond the scope of this paper.

## 6 Examples of Inherently Nondominated Pareto Front Approximations

In this section, we present two examples of inherently nondominated Pareto front approximations constructed as proposed in the previous section. In the first example, we approximate the Pareto front of the three-objective DTLZ5 test problem introduced in (Deb *et al.*, 2002). The DTLZ5 test problem's special feature is that even if it has three objectives, the Pareto front is merely one-dimensional. Additionally, the outcome set  $f(S)$  of the DTLZ5 test problem is nonconvex. For these reasons, approximating the Pareto front of this problem is not trivial, which makes it a suitable task for demonstrating the power of our approximation approach. Even though the DTLZ5 problem is not computationally expensive, we can demonstrate our approximation approach with it, because the computational expense of the problem only affects the computation of the Pareto optimal outcomes  $P$  and not the Pareto front approximation. The second example concerns approximating the Pareto front of the three-objective DTLZ2 test problem (Deb *et al.*, 2002). In this example, we do not show the construction of the approximation but merely give the inherently nondominated Pareto front approximation to further illustrate the versatility of our approach.

The DTLZ5 test problem can be formulated as

$$\begin{aligned}
& \min && (f_1(x), f_2(x), f_3(x)) \\
& \text{s.t.} && 0 \leq x_i \leq 1 \text{ for all } i = 1, 2, 3 \\
& \text{where} && f_1(x) = (1 + x_3^{0.1}) \cos\left(\frac{x_1\pi}{2}\right) \cos(\theta(x)) \\
& && f_2(x) = (1 + x_3^{0.1}) \cos\left(\frac{x_1\pi}{2}\right) \sin(\theta(x)) \\
& && f_3(x) = (1 + x_3^{0.1}) \sin\left(\frac{x_1\pi}{2}\right) \\
& && \theta(x) = \frac{\pi}{4(1+x_3^{0.1})} (1 + 2x_3^{0.1}x_2).
\end{aligned} \tag{5}$$

The actual Pareto front of the DTLZ5 test problem is explicitly known and given as

$$\{(f_1(x), f_2(x), f_3(x))^T : x_3 = 0\}.$$

Assume that the set of initial Pareto optimal outcomes is

$$\begin{aligned}
P &= \{p^1, p^2, p^3, p^4, p^5\} \\
&= \left\{ \left( \frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}}, 0 \right), \left( \frac{\sqrt{2+\sqrt{2}}}{2\sqrt{2}}, \frac{\sqrt{2+\sqrt{2}}}{2\sqrt{2}}, \frac{\sqrt{2-\sqrt{2}}}{2} \right), \right. \\
&\quad \left. \left( \frac{1}{2}, \frac{1}{2}, \frac{1}{\sqrt{2}} \right), \left( \frac{\sqrt{2-\sqrt{2}}}{2\sqrt{2}}, \frac{\sqrt{2-\sqrt{2}}}{2\sqrt{2}}, \frac{\sqrt{2+\sqrt{2}}}{2} \right), (0, 0, 1) \right\}
\end{aligned}$$

given by  $x \in \{(x_1, x_2, x_3) : x_1 = 0, 1/4, 1/2, 3/4, 1, x_2 = 0 \text{ and } x_3 = 0\}$ .

A Delaunay triangulation of this set is given by

$$\begin{aligned}
\mathcal{D} &= \left\{ \emptyset, \mathcal{P}(p^1), \mathcal{P}(p^2), \mathcal{P}(p^3), \mathcal{P}(p^4), \mathcal{P}(p^5), \mathcal{P}(p^1, p^2), \mathcal{P}(p^2, p^3), \right. \\
&\quad \left. \mathcal{P}(p^3, p^4), \mathcal{P}(p^4, p^5), \mathcal{P}(p^5, p^1), \mathcal{P}(p^1, p^2, p^3, p^4, p^5) \right\}, \\
&= \{K^1, \dots, K^{12}\}.
\end{aligned}$$

The dimension of polytope  $K^{12}$  is two so it is not removed before applying the rules. The different dominations between polytopes in the Delaunay triangulation can be presented in a matrix with entries  $a_{ij}$  equal to 1 if the polytope  $K^i$  dominates the polytope  $K^j$  and equal to 0 otherwise. In this problem, the dominations yield a  $12 \times 12$  matrix

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 1 & 0 & 1 \end{bmatrix},$$

where all entries in the 10 top rows are zeros. As an example we verify that polytope  $K^7 = \mathcal{P}(p^1, p^2)$  is dominated by polytope  $K^{11} = \mathcal{P}(p^5, p^1)$  using problems (2) and (3). Problem (2) becomes

$$\begin{aligned} \min \max_{i=1, \dots, k} (s_i^1 - s_i^2) \\ \text{s.t. } s^1 &= (1 - \lambda^1) \left( \frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}}, 0 \right) + \lambda^1 (0, 0, 1) \\ s^2 &= (1 - \lambda^2) \left( \frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}}, 0 \right) + \lambda^2 \left( \frac{\sqrt{2+\sqrt{2}}}{2\sqrt{2}}, \frac{\sqrt{2+\sqrt{2}}}{2\sqrt{2}}, \frac{\sqrt{2-\sqrt{2}}}{2} \right) \\ 0 &\leq \lambda^j \leq 1 \text{ for } j = 1, 2 \end{aligned}$$

and with some reformulation it assumes the form of a linear optimization problem

$$\begin{aligned} \min t \\ \text{s.t. } t &\geq \frac{2 - \sqrt{2+\sqrt{2}}}{2\sqrt{2}} \lambda^2 - \frac{1}{\sqrt{2}} \lambda^1 \\ t &\geq \lambda^1 - \frac{\sqrt{2-\sqrt{2}}}{2} \lambda^2 \\ 0 &\leq \lambda^j \leq 1 \text{ for } j = 1, 2. \end{aligned}$$

This problem can be solved with e.g., the simplex method and the optimal value of  $t$  is

$$\frac{2 - \sqrt{2+\sqrt{2}} - \sqrt{2-\sqrt{2}}}{2(\sqrt{2}+1)} \approx -0.12698 (< 0).$$

By Theorem 7, the polytope  $\mathcal{P}(p^5, p^1)$  dominates polytope  $\mathcal{P}(p^1, p^2)$ .

Notice that in the complex  $\mathcal{D}$  the polytope  $\emptyset$  is a  $-1$ -polytope, the polytopes  $K^2, \dots, K^6$  are  $0$ -polytopes, the polytopes  $K^7, \dots, K^{11}$  are  $1$ -polytopes and the polytope  $K^{12}$  is a  $4$ -polytope. The polytope  $K^{12} \in \mathcal{R}$  by rule (R1), because it dominates itself. This shows up in the above matrix with the entry  $a_{12,12} = 1$ . Of the  $1$ -polytopes, the polytope  $K^{11}$  dominates e.g., the outcome  $p^3 \in P$  and, thus,  $K^{11} \in \mathcal{R}$  by rule (R1). This rule does not apply to the other polytopes except for  $K^{11}$  and  $K^{12}$ . As it can be seen from the matrix, there is no domination left when the polytopes  $K^{11}$  and  $K^{12}$  have been removed from the collection. For this reason, rule (R2) does not apply to any of the polytopes  $K^1, \dots, K^{10}$ . Thus, the  $\mathcal{D}$ -maximal inherently nondominated Pareto front approximation is

$$\begin{aligned} \mathcal{D} \setminus \mathcal{R} &= \mathcal{D} \setminus \left\{ \mathcal{P}(p^1, p^2, p^3, p^4, p^5), \mathcal{P}(p^1, p^5) \right\} \\ &= \{K^1, \dots, K^{10}\} \end{aligned}$$



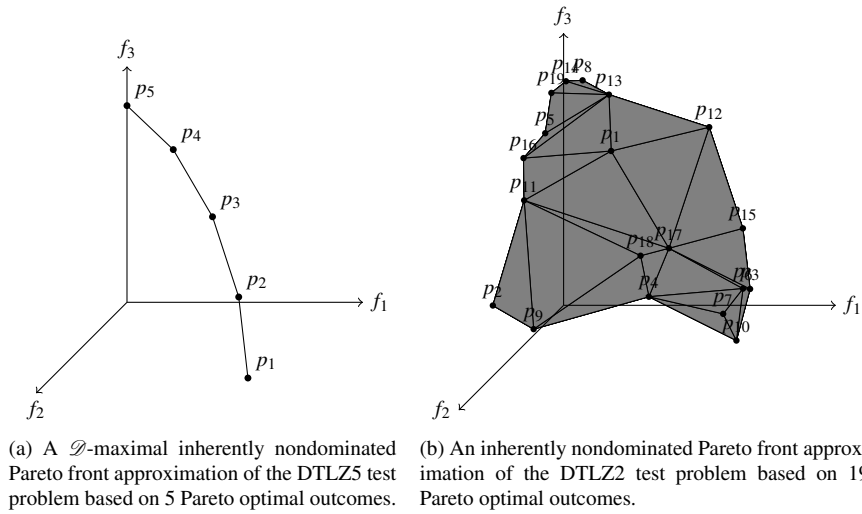


Fig. 1: Bodies of two inherently nondominated Pareto front approximations

The body of this complex is shown in Figure 1a.

The same approach applied to the DTLZ2 test problem introduced in (Deb *et al.*, 2002) with 19 random Pareto optimal outcomes  $p_1, \dots, p_{19}$  yields the approximation shown in Figure 1b. As in the DTLZ5 problem, the Pareto front of this problem is not convex, either.

The two examples presented in this section demonstrate that the proposed approximation approach can handle unusual or difficult problems. The Pareto front of the DTLZ5 test problem is merely one-dimensional while the problem has three objectives and the DTLZ2 test problem is nonconvex. In both cases, the approach produces an interpolation between the known Pareto optimal outcomes that can be used with an interactive method through the surrogate problem defined in Hartikainen *et al.* (2011).

## 7 Two Special Cases of Multiobjective Optimization Problems

As in the previous section, assume that  $\mathcal{D}$  is a Delaunay triangulation of the initial set of Pareto optimal outcomes  $P$  and that the  $k$ -simplices have been excluded from it. In this section, we demonstrate how the concept of a  $\mathcal{D}$ -maximal inherently nondominated Pareto front approximation applies to two special cases, which are the biobjective case and the  $\mathbb{R}_+^k$ -convex case. It is shown that in these special cases the approximation is similar to some approximations in literature and, thus, the approach proposed in this paper can be seen as a generalization of those methods.

## 7.1 Biobjective Case

If the number of objectives of the multiobjective optimization problem is two, then the  $\mathcal{D}$ -maximal inherently nondominated Pareto front approximation is unique and can be explicitly stated. With the help of the following theorem one may construct a  $\mathcal{D}$ -maximal inherently nondominated Pareto front approximation without the rules (R1) and (R2).

**Theorem 9** *If the number of objectives is  $k = 2$  and the initial set of Pareto optimal outcomes is  $P = \{p^1, \dots, p^m\}$  so that  $p_1^1 < p_1^2 < \dots < p_1^m$ , then the only  $\mathcal{D}$ -maximal inherently nondominated Pareto front approximation is*

$$\mathcal{H} = \{\emptyset\} \cup \{\{p\} : p \in P\} \cup \{\mathcal{P}(p^{i-1}, p^i) : i = 2, \dots, m\}.$$

*Proof* Notice that since the outcomes in  $P$  are Pareto optimal, it must be that  $p_2^1 > p_2^2 > \dots > p_2^m$ . Thus, the complex  $\mathcal{H}$  is inherently nondominated, because if  $s^1, s^2 \in \text{body}(\mathcal{H})$  and  $s_1^1 < s_1^2$  then  $s_2^1 > s_2^2$ .

Assume now a non-empty polytope  $K \in \mathcal{D}$  so that  $\mathcal{H} \cup \{K\}$  is inherently nondominated and let a vector  $s \in K$  be arbitrary. Since  $s \in \text{conv}(P)$ , it must be that  $p_1^1 \leq s_1 \leq p_1^m$ . Thus, there must be a vector  $s' \in \text{body}(\mathcal{H})$  so that  $s_1 = s'_1$ . Because the complex  $\mathcal{H} \cup \{K\}$  is assumed inherently nondominated, it must be that  $s'_2 = s_2$  and, thus,  $s' = s$ . Therefore,  $K \subset \text{body}(\mathcal{H})$ . Because the collection  $\mathcal{D}$  is a sub-complex of a Delaunay triangulation of the set  $P$ , it must be that  $K = K'$  for some  $K' \in \mathcal{H}$ . Thus, the complex  $\mathcal{H}$  is a  $\mathcal{D}$ -maximal inherently nondominated Pareto front approximation.

For uniqueness of the complex  $\mathcal{H}$ , assume another inherently nondominated Pareto front approximation  $\mathcal{H}' \subset \mathcal{D}$  so that  $\mathcal{H}' \setminus \mathcal{H} \neq \emptyset$ . Choose a non-empty polytope  $K' \in \mathcal{H}' \setminus \mathcal{H}$ . Since  $\text{vert}(K') \subset P$ ,  $\mathcal{H}'$  is a complex and  $K' \notin \mathcal{H}$ , there must be  $i, j \in \{1, \dots, m\}$  so that  $j - i > 1$  and  $\mathcal{P}(p^i, p^j) \subset K'$  and  $p^{i+1} \notin \mathcal{P}(p^i, p^j)$ . However, by Definition 16  $\{p^{i+1}\} \in \mathcal{H}'$  and since  $\mathcal{H}'$  is inherently nondominated,  $p^{i+1} \in \mathcal{P}(p^i, p^j)$ .  $\square$

**Corollary 2** *If the number of objectives is  $k = 2$ , then the  $\mathcal{D}$ -maximal inherently nondominated Pareto front approximation is connected.*

According to Theorem 9, the only  $\mathcal{D}$ -maximal inherently nondominated Pareto front approximation is the piece-wise linear curve that connects adjacent Pareto optimal outcomes. This approximation has been used in e.g., in (Schandl *et al.*, 2001).

## 7.2 Convex Case

In this subsection, we study the structure of the Pareto front approximation in convex multiobjective optimization problems. Convex multiobjective optimization problems have been studied extensively (see e.g., (Ruzika and Wiecek, 2005)). For our treatment of this case, we need a lemma from (Yu and Zeleny, 1975).

**Lemma 2** *The Pareto front of a polytope can be represented as a union of the Pareto faces (i.e., the faces of the polytope in which all the vectors are Pareto optimal) of the polytope.*

**Definition 19** Assume that  $B \subset \mathbb{R}^k$  is a set. Then a set  $A \subset \mathbb{R}^k$  is called  $B$ -convex, if the set  $A + B = \{a + b : a \in A, b \in B\}$  is convex.

**Theorem 10** *If the outcome set  $f(S)$  is  $\mathbb{R}_+^k$ -convex, then the complex*

$$\mathcal{K} = \{K \in \mathcal{D} : K \subset \text{PF}(\text{conv}(P))\}$$

*is a  $\mathcal{D}$ -maximal inherently nondominated Pareto front approximation.*

*Proof* First, the collection  $\mathcal{K}$  is inherently nondominated because, as noticed in (Hartikainen *et al.*, 2011), the Pareto front is inherently nondominated and  $\text{body}(\mathcal{K}) \subset \text{PF}(\text{conv}(P))$  by definition. Second, because a face of a polytope is a subset of the polytope, then if the polytope is in the collection  $\mathcal{K}$ , then the face is in the collection. This again implies that the collection  $\mathcal{K}$  is a complex. Thus the collection  $\mathcal{K}$  is an inherently nondominated complex.

To prove that the inherently nondominated complex  $\mathcal{K}$  is  $\mathcal{D}$ -maximal, let a polytope  $K \in \mathcal{D}$  be so that the collection  $\mathcal{K} \cup \{K\}$  is inherently nondominated. If  $K \notin \mathcal{K}$ , then there exists a vector  $s \in K \setminus \text{PF}(\text{conv}(P))$ . This immediately implies that there exists  $s' \in \text{PF}(\text{conv}(P))$  so that  $s' \leq s$ . From Lemma 2 and Definition 13 it follows that

$$\text{PF}(\text{conv}(P)) = \text{body}(\mathcal{K})$$

and, thus,  $s' \in \text{body}(\mathcal{K})$ , which contradicts the inherent nondominance of  $\mathcal{K} \cup \{K\}$ . Therefore  $K \in \mathcal{K}$  and the complex  $\mathcal{K}$  is a  $\mathcal{D}$ -maximal inherently nondominated Pareto front approximation.  $\square$

The set  $\text{PF}(\text{conv}(P))$  has been used to approximate the Pareto front in e.g., (Eskelinen *et al.*, 2010; Lotov *et al.*, 2004). According to the following corollary the body of a  $\mathcal{D}$ -maximal Pareto front approximation based on  $P$  is equal to this set in the  $\mathbb{R}_+^k$ -convex case.

**Corollary 3** *If the outcome set  $f(S)$  is  $\mathbb{R}_+^k$ -convex, then there exists a  $\mathcal{D}$ -maximal inherently nondominated Pareto front approximation  $\mathcal{K}$  so that*

$$\text{body}(\mathcal{K}) = \text{PF}(\text{conv}(P)).$$

## 8 Error Estimates

In this section, we examine how well a  $\mathcal{D}$ -maximal inherently nondominated Pareto front approximation  $\mathcal{K}$  approximates the Pareto front. More accurately, we develop the following estimate: given a vector  $s \in \text{body}(\mathcal{K})$  we estimate an *error vector*  $d(s) \in \mathbb{R}^k$  that fulfills two conditions:

1. there exists an outcome  $z \in f(S)$  so that  $z_i \leq s_i + d_i(s)$  for all  $i = 1, \dots, k$  and
2. there does not exist an outcome  $z \in f(S)$  so that  $z \leq s - d(s)$ .

These error estimates should have a clear meaning to the decision maker: (1) implies that there exists an actual Pareto optimal outcome that is at least as good as  $s + d(s)$  in all objectives and (2) implies that there is no actual outcome that dominates  $s - d(s)$ .

We estimate  $d(s)$  in two ways. In Subsection 8.1, we develop estimates for  $d(s)$  that require computing the approximation before the estimation and, in Subsection 8.2, we develop estimates that can be computed by means of just the set of initial Pareto optimal outcomes  $P$ . The former error estimates can be used to help the decision maker in choosing a vector on the Pareto front approximation to be projected on the actual Pareto front. The latter estimates can be used for deciding whether to add more outcomes into  $P$  before computing the approximation – if these error estimates are big it may be useful to compute more Pareto optimal outcomes if possible.

### 8.1 Estimates after Computing the Approximation

In this section, we assume that the complex  $\mathcal{K}$  has already been computed. We develop estimates for  $d(s)$  defined above. We start with a lemma.

**Lemma 3** *Let  $s, e \in \mathbb{R}^k$ . If there exists an outcome  $p \in P$  so that  $p_i \geq s_i - e_i$  for all  $i = 1, \dots, k$ , then there does not exist an outcome  $z \in f(S)$  so that  $z \leq s - e$ .*

*Proof* This is clear since  $p$  is assumed to be Pareto optimal. □

The following lemma gives the required estimates and the main result is given by Theorem 11.

**Lemma 4** *Let  $K = \mathcal{P}(z^1, \dots, z^{a+1}) \in \mathcal{K}$  be a polytope. Furthermore, let  $s = \sum_{j=1}^{a+1} \lambda^j z^j \in K$  be a vector in this polytope with  $\lambda^j \geq 0$  for all  $j = 1, \dots, a+1$  and  $\sum_{j=1}^{a+1} \lambda^j = 1$  and let  $z^{\max}, z^{\min}$  and  $z^{\text{dif}}$  be vectors in  $\mathbb{R}^k$  so that for all  $i \in \{1, \dots, k\}$*

$$z_i^{\max} = \max_{j \in \{1, \dots, a+1\}} z_i^j, \quad z_i^{\min} = \min_{j \in \{1, \dots, a+1\}} z_i^j \quad \text{and} \quad z_i^{\text{dif}} = z_i^{\max} - z_i^{\min}.$$

*Then,*

1.  $s_i - (1 - \lambda^j) z_i^{\text{dif}} \leq z_i^j$  and
2.  $s_i + (1 - \lambda^j) z_i^{\text{dif}} \geq z_i^j$

*for all  $j = 1, \dots, a+1$  and  $i = 1, \dots, k$ .*

*Proof* Let  $\hat{j} \in \{1, \dots, a+1\}$  and  $i \in \{1, \dots, k\}$  be arbitrary. With the above definitions

$$\begin{aligned}
s_i - (1 - \lambda^{\hat{j}})z_i^{\text{dif}} &= \sum_{j=1}^{a+1} \lambda^j z_i^j - (1 - \lambda^{\hat{j}})z_i^{\text{dif}} \\
&= z_i^{\max} + \sum_{j=1}^{a+1} \lambda^j (z_i^j - z_i^{\max}) - (1 - \lambda^{\hat{j}})z_i^{\text{dif}} \\
&\leq z_i^{\max} + \lambda^{\hat{j}}(z_i^{\hat{j}} - z_i^{\max}) - (1 - \lambda^{\hat{j}})z_i^{\text{dif}} \\
&= (1 - \lambda^{\hat{j}})z_i^{\max} + \lambda^{\hat{j}}z_i^{\hat{j}} - (1 - \lambda^{\hat{j}})z_i^{\max} + (1 - \lambda^{\hat{j}})z_i^{\min} \\
&= \lambda^{\hat{j}}z_i^{\hat{j}} + (1 - \lambda^{\hat{j}})z_i^{\min} \\
&\leq z_i^{\hat{j}},
\end{aligned}$$

where the first inequality follows from the property  $z_i^j - z_i^{\max} \leq 0$  for all  $j = 1, \dots, a+1$  and the second one follows from the inequality  $z_i^{\min} \leq z_i^{\hat{j}}$ . In a similar way we obtain

$$s_i + (1 - \lambda^{\hat{j}})z_i^{\text{dif}} \geq z_i^{\hat{j}}.$$

This proves the assertion.  $\square$

**Theorem 11** Let  $K = \mathcal{P}(z^1, \dots, z^{a+1}) \in \mathcal{K}$  be a polytope. Furthermore, let  $s = \sum_{j=1}^{a+1} \lambda^j z^j \in K$  be a vector in this polytope and let  $z^{\max}$ ,  $z^{\min}$  and  $z^{\text{dif}}$  be defined as above. Then  $d(s) \leq (1 - \lambda^{\hat{j}})z^{\text{dif}}$  for all  $j = 1, \dots, a+1$ .

*Proof* This follows from Lemmas 3 and 4.  $\square$

**Corollary 4** With the definitions of Theorem 11, it holds that  $d(s) \leq az^{\text{dif}}/(a+1)$ .

*Proof* Since  $\sum_{i=1}^{a+1} \lambda^i = 1$ , then there exists  $\hat{j} \in \{1, \dots, a+1\}$  so that  $\lambda^{\hat{j}} \geq 1/(a+1)$  and, thus,  $(1 - \lambda^{\hat{j}}) \leq 1 - 1/(a+1) = a/(a+1)$ . This combined with Theorem 11 proves the assertion.  $\square$

*Example 2* Continuing the DTLZ5 example from Section 6 assume that

$$s = 1/3p^3 + 2/3p^4 \approx (0.34707, 0.34707, 0.85162) \in K^9 = \mathcal{P}(p^3, p^4).$$

Then  $z^{\max} = (1/2, 1/2, \sqrt{2 + \sqrt{2}}/2)$ ,  $z^{\min} = (2 - \sqrt{2})/(2\sqrt{2}), (2 - \sqrt{2})/(2\sqrt{2}), 1/\sqrt{2})$  and

$$\begin{aligned}
z^{\text{dif}} = z^{\max} - z^{\min} &= \left( \frac{\sqrt{2} - \sqrt{2 - \sqrt{2}}}{2\sqrt{2}}, \frac{\sqrt{2} - \sqrt{2 - \sqrt{2}}}{2\sqrt{2}}, \frac{\sqrt{2}\sqrt{2 + \sqrt{2}} - 1}{2\sqrt{2}} \right) \\
&\approx (0.22940, 0.22940, 0.21677).
\end{aligned}$$

Theorem 11 yields

$$d(s) \leq (1 - 2/3)(0.22940, 0.22940, 0.21677) \approx (0.076467, 0.076467, 0.072257)$$

and, thus,

1. there exists an outcome  $z \in f(S)$  that is at least as good as

$$s + d \approx (0.42353, 0.42353, 0.92388)$$

in all objectives and

2. there does not exist an outcome  $z \in f(S)$  so that

$$z \leq s - d \approx (0.27060, 0.27060, 0.77936).$$

## 8.2 Estimates before Computing the Approximation

In this section, we assume that the complex  $\mathcal{K}$  has not yet been computed but we have the set of initial Pareto optimal outcomes  $P$ . Theorem 12 establishes an estimate for  $d(s)$  without computing the complex  $\mathcal{K}$ .

**Theorem 12** *Assume that  $K^l = \mathcal{P}(p^{j_1^l}, p^{j_2^l})$  with Pareto optimal outcomes  $p^{j_1^l}, p^{j_2^l} \in P$  and  $l = 1, \dots, r$  are all the 1-polytopes so that  $K^l \cup P$  is inherently nondominated. Define a vector  $e \in \mathbb{R}^k$  so that*

$$e_i = \max_{l=1, \dots, r} |z_i^{j_1^l} - z_i^{j_2^l}|$$

for all  $i = 1, \dots, k$ . Then it holds that

$$d(s) \leq \frac{k-1}{k} e$$

for all  $s \in \text{body}(\mathcal{K})$ .

*Proof* Assume that  $\mathcal{K}$  is a  $\mathcal{D}$ -maximal inherently nondominated Pareto front approximation. The structure of  $\mathcal{K}$  remains unknown but the existence of the approximation is proven in Theorem 4. Let a vector  $s \in \text{body}(\mathcal{K})$  be arbitrary. Then  $s \in \mathcal{P}(\hat{p}^1, \dots, \hat{p}^{a+1})$  for some  $\hat{p}^1, \dots, \hat{p}^{a+1} \in P$ . By Theorem 2, there exist  $z^1, \dots, z^k \in \{\hat{p}^1, \dots, \hat{p}^m\}$  so that  $s \in \mathcal{P}(z^1, \dots, z^k)$ . Now the union  $\mathcal{P}(z^{j^1}, z^{j^2}) \cup P$  is inherently nondominated for all indices  $j^1, j^2 \in \{1, \dots, k\}$ , because  $\mathcal{K}$  is an inherently nondominated Pareto front approximation and, thus,

$$|z_i^{j^1} - z_i^{j^2}| \leq e_i$$

for all indices  $i, j^1, j^2 \in \{1, \dots, k\}$ . This immediately implies  $z_i^{\text{dif}} \leq e_i$  for all  $i = 1, \dots, k$  with the definitions of Lemma 4 and this combined with Theorem 4 yields the claim.  $\square$

In order to be useful, the computation of the above a priori error estimate should be less expensive than the computation of the whole  $\mathcal{D}$ -maximal inherently nondominated Pareto front approximation. This indeed is the case, if the number of objectives  $k$  is sufficiently large. To find the polytopes  $K^l$  in Theorem 12, one has to solve problems (2) and (3)  $O(m^3)$  times, while the computation of the  $\mathcal{D}$ -maximal inherently nondominated Pareto front approximation demands solving problems (2) and (3)  $O(m^k)$  times in the worst case scenario. Thus, if  $k > 3$ , then the computation of the a priori estimate is worthwhile.

*Example 3* Continuing with the DTLZ5 example from Section 6, the 1-polytopes  $K^l$  of Theorem 12 are  $K^7, K^8, K^9, K^{10}$ . We calculate  $e_i = \max\{|p_i^1 - p_i^2|, |p_i^2 - p_i^3|, |p_i^3 - p_i^4|, |p_i^4 - p_i^5|\}$  for all  $i = 1, 2, 3$  and obtain

$$e = \left( \frac{\sqrt{2-\sqrt{2}}}{2\sqrt{2}}, \frac{\sqrt{2-\sqrt{2}}}{2\sqrt{2}}, \frac{\sqrt{2-\sqrt{2}}}{2} \right) \\ \approx (0.27060, 0.27060, 0.38268).$$

By Theorem 12,  $d(s) \leq 2/3e \approx (0.18040, 0.18040, 0.25512)$  and for every  $s \in \text{body}(\mathcal{K})$

1. there exists an outcome  $z \in f(S)$  that is at least as good as

$$s + (0.18040, 0.18040, 0.25512)$$

in all objectives and

2. there does not exist an outcome  $z \in f(S)$  so that  $z \leq s - (0.18040, 0.18040, 0.25512)$ .

## 9 Concluding Remarks on Decision Making with the Pareto Front Approximation

We have introduced a way to construct a Pareto front approximation that can be used in decision making involving computationally expensive multiobjective optimization problems. As shown in (Hartikainen *et al.*, 2011), the problem of choosing a preferred vector on the inherently nondominated Pareto front approximation  $\mathcal{K}$  can be formulated as a multiobjective optimization problem

$$\begin{aligned} \min \quad & (z_1, \dots, z_k) \\ \text{s.t.} \quad & z \in \text{body}(\mathcal{K}) \\ \text{where} \quad & z = (z_1, \dots, z_k)^T, \end{aligned} \tag{6}$$

which can be used as a surrogate to Problem (1). Additionally, it holds that

$$\text{PF}(\text{body}(\mathcal{K})) = \text{body}(\mathcal{K}),$$

because  $\mathcal{K}$  is inherently nondominated and, consequently, all vectors in the set  $\text{body}(\mathcal{K})$  are feasible solutions to problem (6). A preferred outcome of problem (6) is called a preferred vector on the approximation  $\mathcal{K}$ . Since problem (6) is a multiobjective optimization problem, any interactive multiobjective optimization methods can be employed to solve it.

The body of the inherently nondominated Pareto front approximation  $\mathcal{K}$  can be parameterized and so problem (6) has a representation that can be input to e.g., the WWW-NIMBUS<sup>®</sup> implementation (Miettinen and Mäkelä, 2000, 2006) (available at <http://nimbus.it.jyu.fi/>) of the NIMBUS method (Miettinen and Mäkelä, 1995, 2006). Obviously, the parameters that are used for parameterizing the body of  $\mathcal{K}$  are not meaningful to the decision maker but the vectors  $z \in \text{body}(\mathcal{K})$  are. The only drawback of this approach is that the parameters used for describing the different polytopes in the complex  $\mathcal{K}$  are discrete and thus problem (6) has discrete

variables in its decision space. In effect, the solvers used for solving the scalarizations employed by an interactive multiobjective optimization method must be able to handle discrete variables. Fortunately, e.g., WWW-NIMBUS<sup>®</sup> has global solvers that are able to do this.

The a priori error estimates are useful whenever a decision maker has chosen a preferred vector on the approximation. Whenever this happens, the decision maker can be given information about the actual Pareto optimal solutions in the form of error estimates. Having examined the error estimates, the decision maker can either choose to have the chosen point on the approximation projected on the actual Pareto front by means of the achievement scalarizing function (Wierzbicki, 1986) or not.

Future research on this topic is needed to implement the approximation approach given in this paper. For example, the computational expense of directly employing rules (R1) and (R2) grows exponentially when the number of objectives rises and smart computation schemes are needed to handle this. Second, we need to find an efficient way to solving problems (2) and (3). As further research, it is of interest to study which interactive multiobjective optimization methods and implementations are most suitable for solving problem (6) in real life problems and with real decision makers. The choice of the interactive method may obviously depend on the initial multiobjective optimization problem and also on the decision maker's desires. Finally, to make our approximation approach more versatile, we intend to examine how the continuity assumption in Section 2 can be relaxed.

Another direction of research is to study other uses of our approximation in multiobjective optimization. For example, in bilevel multiobjective optimization it is sometimes required to approximate the Pareto front of the lower level problem that is then brought to the upper level. Because of the inherent nondominance property, our approximation has potential to be efficient in this kind of use.

## References

- Bezerkin, V. E., Kamenev, G. K., and Lotov, A. V. (2006). Hybrid Adaptive Methods for Approximating a Nonconvex Multidimensional Pareto Frontier. *Computational Mathematics and Mathematical Physics*, **46**, 1918–1931.
- Boissonnant, J.-D. (1984). Geometric Structures for Three-Dimensional Shape Representation. *ACM Transactions on Graphics*, **3**, 266–286.
- Caratheodory, C. (1913). Bedingt Konvergente Reihen und Konvexe Systeme. *Journal für die reine und angewandte Mathematik*, **143**, 128–175.
- Coello Coello, C. A., Van Veldhuizen, D. A., and Lamont, G. B. (2007). *Evolutionary Algorithms for Solving Multi-Objective Problems*. Springer, New York.
- Deb, K. (2001). *Multi-Objective Optimization Using Evolutionary Algorithms*. John Wiley & Sons, Chichester.
- Deb, K., Thiele, L., Laumanns, M., and Zitzler, E. (2002). Scalable Multi-Objective Optimization Test Problems. *IEEE International Conference on E-Commerce Technology*, **1**, 825–830.
- Edelsbrunner, H. (1987). *Algorithms in Combinatorial Geometry*. Springer, New York.



- Edelsbrunner, H. (1998). Shape Reconstruction with Delaunay Complex. In C. L. Lucchesi and A. V. Moura, editors, *LATIN'98: Theoretical Informatics*, pages 119–132. Springer-Verlag Berlin, Heidelberg.
- Edelsbrunner, H. and Shah, N. R. (1994). Triangulating Topological Spaces. In *SCG '94: Proceedings of the Tenth Annual Symposium on Computational Geometry*, pages 285–292. ACM, New York.
- Efremov, R. V. and Kamenev, G. K. (2009). Properties of a Method for Polyhedral Approximation of the Feasible Criterion Set in Convex Multiobjective Problems. *Annals of Operations Research*, **166**, 271–279.
- Eskelinen, P., Miettinen, K., Klamroth, K., and Hakanen, J. (2010). Pareto Navigator for Interactive Nonlinear Multiobjective Optimization. *OR Spectrum*, **32**, 211–227.
- Fortune, S. (1997). Voronoi Diagrams and Delaunay Triangulations. In J. E. Goodman and J. O'Rourke, editors, *Handbook of Discrete and Computational Geometry*. CRC Press, Boca Raton.
- Fukuda, K. (2004). Polyhedral Computation FAQ. Swiss Federal Institute of Technology, url = <http://www.ifor.math.ethz.ch/~fukuda/polyfaq/polyfaq.html>.
- George, P.-L. and Borouchaki, H. (1998). *Delaunay Triangulation and Meshing: Application to Finite Elements*. Hermes, Paris.
- Goel, T., Vaidyanathan, R., Haftka, R. T., Shyy, W., Queipo, N. V., and Tucker, K. (2007). Response Surface Approximation of Pareto Optimal Front in Multi-Objective Optimization. *Computer Methods in Applied Mechanics and Engineering*, **196**, 879–893.
- Goodman, J. E. and O'Rourke, J., editors (1997). *Discrete and Computational Geometry*. CRC Press, Boca Raton.
- Grünbaum, B. (1967). *Convex Polytopes*. Interscience Publishers, London.
- Hartikainen, M., Miettinen, K., and Wiecek, M. M. (2011). Pareto Front Approximations for Decision Making with Inherent Non-dominance. In Y. Shi, S. Wang, G. Kou, and J. Wallenius, editors, *New State of MCDM in the 21st Century, Selected Papers of the 20th International Conference on Multiple Criteria Decision Making 2009*, pages 35–46. Springer-Verlag Berlin, Heidelberg.
- Hasenjäger, M. and Sendhoff, B. (2005). Crawling Along the Pareto Front: Tales From the Practice. In *The 2005 IEEE Congress on Evolutionary Computation (IEEE CEC 2005)*, pages 174–181, Piscataway, NJ. IEEE Press.
- Keeney, R. L. and Raiffa, H. (1993). *Decisions with Multiple Objectives*. Cambridge University Press, Cambridge.
- Laukkanen, T., Tveit, T.-M., Ojalehto, V., Miettinen, K., and Fogelholm, C.-J. (2010). An Interactive Multi-Objective Approach to Heat Exchanger Network Synthesis. *Computers & Chemical Engineering*, **34**, 943–952.
- Lotov, A. V., Bushenkov, V. A., and Kamenev, G. A. (2004). *Interactive Decision Maps*. Kluwer Academic Publishers, Boston.
- Luque, M., Ruiz, F., and Miettinen, K. (2011). Global Formulation for Interactive Multiobjective Optimization. *OR Spectrum*, **33**, 27–48.
- Martin, J., Bielza, C., and Insua, D. R. (2005). Approximating Nondominated Sets in Continuous Multiobjective Optimization Problems. *Naval Research Logistics*, **52**, 469–480.

- McMullen, P. (1970). The Maximum Number of Faces of a Convex Polytope. *Mathematika*, **17**, 179–184.
- Miettinen, K. (1999). *Nonlinear Multiobjective Optimization*. Kluwer Academic Publishers, Boston.
- Miettinen, K. and Mäkelä, M. (1995). Interactive Bundle-based Method for Nondifferentiable Multiobjective Optimization: NIMBUS. *Optimization*, **34**, 231–246.
- Miettinen, K. and Mäkelä, M. (2000). Interactive Multiobjective Optimization System WWW-NIMBUS on the Internet. *Computers & Operations Research*, **27**, 709–723.
- Miettinen, K. and Mäkelä, M. M. (2006). Synchronous Approach in Interactive Multiobjective Optimization. *European Journal of Operational Research*, **170**, 909–922.
- Miettinen, K., Ruiz, F., and Wierzbicki, A. P. (2008). Introduction to Multiobjective Optimization: Interactive Approaches. In J. Branke, K. Deb, K. Miettinen, and R. Slowinski, editors, *Multiobjective Optimization: Interactive and Evolutionary Approaches*, pages 27–57. Springer-Verlag Berlin, Heidelberg.
- Monz, M. (2006). *Pareto Navigation - Algorithmic Foundation of Interactive Multi-criteria IMRT Planning*. Ph.D. thesis, University of Kaiserslautern.
- Rajan, V. T. (1994). Optimality of the Delaunay Triangulation in  $\mathbb{R}^d$ . *Discrete and Computational Geometry*, **12**, 189–202.
- Ruzika, S. and Wiecek, M. M. (2005). Approximation Methods in Multiobjective Programming. *Journal of Optimization Theory and Applications*, **126**, 473–501.
- Sawaragi, Y., Nakayama, H., and Tanino, T. (1985). *Theory of Multiobjective Optimization*. Academic Press, Orlando.
- Schandl, B., Klamroth, K., and Wiecek, M. M. (2001). Norm-Based Approximation in Bicriteria Programming. *Computational Optimization and Applications*, **20**, 23–42.
- Schandl, B., Klamroth, K., and Wiecek, M. M. (2002). Norm-Based Approximation in Multicriteria Programming. *Computers & Mathematics with Applications*, **44**, 925–942.
- Shenton, D. and Cendes, Z. (1985). Three-Dimensional Finite Element Mesh Generation Using Delaunay Tessellation. *IEEE Transactions on Magnetics*, **21**, 2535–2538.
- Steuer, R. E. (1986). *Multiple Criteria Optimization: Theory, Computation and Application*. John Wiley & Sons, New York.
- Steuer, R. E. (1989). The Tchebycheff Procedure of Interactive Multiple Objective Programming. In B. Karpak and S. Zionts, editors, *Multiple Criteria Decision Making and Risk Analysis Using Micro Computers*. Springer-Verlag Berlin, Heidelberg.
- Steuer, R. E. and Choo, E.-U. (1983). An Interactive Weighted Tchebycheff Procedure for Multiple Objective Programming. *Mathematical Programming*, **26**, 326–344.
- Vegter, G. (1997). Computational Topology. In J. E. Goodman and J. O’Rourke, editors, *Handbook of Discrete and Computational Geometry*, pages 719–742. CRC Press, Boca Raton.

- 
- Wierzbicki, A. P. (1986). On the Completeness and Constructiveness of Parametric Characterizations to Vector Optimization Problems. *OR Spectrum*, **8**, 73–87.
- Yu, P. L. and Zeleny, M. (1975). The Set of All Nondominated Solutions in Linear Cases and a Multicriteria Simplex Method. *Journal of Mathematical Analysis and Applications*, **49**, 430–468.

**PIII**

**A COMPUTATIONALLY INEXPENSIVE APPROACH IN  
MULTIOBJECTIVE HEAT EXCHANGER NETWORK  
SYNTHESIS**

by

Markus Hartikainen · Kaisa Miettinen 2010

in M. Collan (editor) *Proceedings of the 2nd International Conference on Applied  
Operational Research - ICAOR'10*, pages 99-109

Reproduced with kind permission of the International Conference on Applied  
Operational Research.

# A Computationally Inexpensive Approach in Multiobjective Heat Exchanger Network Synthesis

Markus Hartikainen  
Kaisa Miettinen

Department of Mathematical Information Technology  
University of Jyväskylä  
Finland

markus.e.hartikainen@jyu.fi  
kaisa.miettinen@jyu.fi

June 26, 2010

## Abstract

We consider a heat exchanger network synthesis problem formulated as a multiobjective optimization problem. The Pareto front of this problem is approximated with a new approximation approach and the preferred point on the approximation is found with the interactive multiobjective optimization method NIMBUS. Using the approximation makes the solution process computationally inexpensive. Finally, the preferred outcome on the Pareto front approximation is projected on the actual Pareto front.

**Keywords:** heat exchanger network synthesis, multiobjective optimization, Pareto front approximation, interactive decision making, NIMBUS

## 1 Introduction

Multiobjective optimization means optimizing multiple objectives at the same time (11). In multiobjective optimization problems there is no well defined single optimal solution, but we can identify a set of so-called Pareto optimal solutions where none of the objectives can be improved without impairing some other. A vector containing the values of all objectives as its components is called an outcome and an outcome given by a Pareto optimal solution is called a Pareto optimal outcome. The set of Pareto optimal solutions can be called the Pareto front.

The many mathematically equal Pareto optimal solutions cannot be ordered without preference information related to outcomes on the Pareto front and the person giving this

information is called a decision maker. We can say that the aim of multiobjective optimization approaches is to help the decision maker in finding the outcome that is the most preferable. Different approaches to multiobjective optimization problems are summarized e.g., in (1; 9; 11).

The heat exchanger network synthesis problem is an important problem concerning heat exchange and the efficient use of energy in it. In (10), this problem is formulated in a new way as a multiobjective optimization problem and solved with an approach where GAMS (the General Algebraic Modeling System, see <http://www.gams.com/>) and the interactive multiobjective optimization method NIMBUS (12; 14) are combined. The use of the NIMBUS method enables the examination of the trade-offs between different objectives of the problem.

The drawback in the interactive method is, however, the fact that the problem is computationally expensive and the decision maker has to wait a relatively long time to get feedback for his/her preferences. This is a common issue, when the problem formulation includes a simulator and function evaluations take time. To keep waiting times feasible a local optimizer is used in (10) to generate Pareto optimal solutions within NIMBUS. Local optimizers, however, have the risk of getting caught in local optima instead of global ones.

In (8), an idea inspired by (4) for solving computationally expensive multiobjective optimization problems is introduced. In this paper, it is proposed that the Pareto front should be approximated in a way that enables examining this Pareto front approximation with existing interactive multiobjective optimization methods. It is argued that the computational expense of the approximation phase which takes place before introducing the decision maker in the solution process does not matter as much as the time-consuming computation done while the decision maker is waiting. In that paper it is argued that an *inherently nondominated Pareto front approximation*, as defined therein, is a good approximation to be used because it does not mislead the decision maker in what is attainable in the problem and what is not. In (7), a method for constructing such an inherently nondominated Pareto front approximation is presented based on the Delaunay triangulation.

In this paper, we present an application of the Pareto front approximation approach in heat exchanger network synthesis. We start with a small set of Pareto optimal outcomes. An inherently nondominated Pareto front approximation is constructed based on these outcomes with the approach given in (7). After this, the Pareto front approximation is parametrized to get a multiobjective optimization problem related to the approximation as defined in (8) and the preferred outcome on the Pareto front approximation is found with the WWW-NIMBUS<sup>®</sup> implementation of the NIMBUS method. The parametrized multiobjective optimization problem based on the approximation is computationally much less costly than the original problem. This is a benefit compared to (Laukkanen et al, 2010). Finally, the preferred solution on the Pareto front approximation is projected on the actual Pareto front by the means of an achievement scalarizing function (18).

The rest of this paper is organized as follows. In Section 2, we briefly describe the heat exchanger network synthesis problem considered. We introduce the tools used, that is our approximation algorithm and NIMBUS, in Section 3. Section 4 is devoted to generating

a Pareto front approximation for the problem and to finding the preferred point on the approximation with the help of NIMBUS. Finally, conclusions are drawn in Section 5.

## 2 Heat Exchanger Network Synthesis

The heat exchanger network synthesis is a problem for the efficient use of energy. According to (10), the objective of research in heat exchanger network synthesis is to

... design a heat exchanger network that minimises the total annualised cost, given sets of hot streams, cold streams, hot utilities and cold utilities. Each hot and cold stream has a specific heat capacity flowrate, a start- and target temperature.

For more detailed treatment of the topic see (10) and references therein. Different optimization based solution approaches are given e.g., in (2; 10; 16). Even though the problem has conflicting objectives, only in (10) it is clearly formulated as a multiobjective optimization problem.

A way to model the heat exchanger network synthesis, called the SynHeat model, is introduced in (19). Based on this model, a multiobjective optimization problem with four objectives

$$\begin{array}{ll}
 \min & \text{Cold utility consumption} \\
 \min & \text{Hot utility consumption} \\
 \min & \text{Number of heat exchanger units} \\
 \min & \text{Total heat exchanger surface area} \\
 \text{s.t.} & \text{Energy balance for each stream} \\
 & \text{Energy balance for each stage} \\
 & \text{Calculation of hot and cold utility requirements} \\
 & \text{Assignment of inlet temperatures} \\
 & \text{Feasibility of temperatures} \\
 & \text{Logical constraints for process stream matches and utility matches} \\
 & \text{Calculation of approach temperatures}
 \end{array}$$

was formulated in (10). For details, see the appendix of (10). In this paper, we to consider a slight modification of the above with three objectives

$$\begin{array}{ll}
 \min & \text{Number of heat exchanger units} \\
 \min & \text{Total heat exchanger surface area} \\
 \min & \text{Hot utility consumption} \\
 \text{s.t.} & x \in S,
 \end{array} \tag{1}$$

where  $S$  is the set given by the constraints in the above multiobjective optimization problem. The most important reason not to consider the cold utility consumption objective is that it has been noticed to correlate with the hot utility consumption and it suffices to examine only

one of them. Also an advantage of having three objectives is that then we can conveniently illustrate the approximated Pareto front. Let us however stress that the usability of our approximation approach does not depend on the number of objectives.

### 3 Background: Approximation and NIMBUS

Before solving the synthesis problem, we briefly describe the methods used. In other words, we introduce the approximation method used as well as the NIMBUS method.

A Pareto front approximation is defined by (17) to be a set  $A$  in the objective space considered a surrogate of the Pareto front. Approximating the Pareto front of a multiobjective optimization problem is an interesting problem from both the application and the theoretical viewpoint. Surveys of approximation methods can be found in (Hartikainen et al., 2010 b) and in references therein. However, as noted in (8), most of the Pareto front approximations have not been designed with decision making at mind but mainly for representing Pareto optimal solutions.

In (8), inherent nondominance is introduced as a desirable property for a Pareto front approximation and it is shown that an inherently nondominated approximation of the Pareto front is a good basis for decision making. A set  $A$  in the objective space is defined to be inherently nondominated, if there does not exist vectors  $a, b \in A$  so that  $a$  dominates  $b$ . A vector  $a$  is said to dominate another vector  $b$ , if  $a$  is at least as good as  $b$  in all objectives and strictly better in at least one. If  $P$  is a set of Pareto optimal outcomes for the multiobjective optimization problem, an inherently nondominated set  $A$  is called an inherently nondominated Pareto front approximation based on  $P$ , if also  $P \subset A$ . Furthermore, an inherently nondominated set  $A$  is called a  $B$ -maximal inherently nondominated Pareto front approximation for some other set  $B$  if there does not exist an inherently nondominated set  $\tilde{A} \subset B$  so that  $A \subsetneq \tilde{A}$ . In (8), also an approach to solve multiobjective optimization problems using a inherently nondominated Pareto front approximation was proposed, where a parametrized problem is formed with the approximation so that e.g., interactive multiobjective optimization methods can be applied for decision making on the approximation.

In (7), an approach to construct a  $\mathcal{D}$ -maximal inherently nondominated approximation is given, where  $\mathcal{D}$  is a Delaunay triangulation of a given subset of Pareto optimal outcomes  $P$ . The Delaunay triangulation is a complex (i.e., collection of polytopes with certain additional properties (6)) so that the body (i.e., the union of all polytopes in the complex) is the convex hull of the outcomes in  $P$  and also some other properties are satisfied. The Delaunay triangulation is a useful concept in computational geometry (5). It is shown in (7) that an inherently nondominated complex is such that there are no two polytopes in the complex so that one dominates the other. It is said that a polytope  $K^1$  in the objective space dominates another polytope  $K^2$  also in the objective space, if there exists vectors  $s^1 \in K^1$  and  $s^2 \in K^2$  so that  $s^1$  dominates  $s^2$ . In (7), it is shown that, if all the objectives of the multiobjective optimization problem are to be minimized and  $k$  is the number of objectives, the domination



between polytopes  $K^1$  and  $K^2$  can be found by solving optimization problems

$$\begin{aligned} \min & \max_{i=1,\dots,k} (s_i^1 - s_i^2) \\ \text{s.t. } & s^1 \in K^1, s^2 \in K^2 \end{aligned} \quad (2)$$

and

$$\begin{aligned} \min & \sum_{i=1}^k (s_i^1 - s_i^2) \\ \text{s.t. } & s^1 \in K^1, s^2 \in K^2 \\ & s_i^1 \leq s_i^2 \text{ for all } i = 1, \dots, k. \end{aligned} \quad (3)$$

The polytope  $K^1$  dominates the polytope  $K^2$  if and only if one of the following holds: (i) the optimal value in problem (2) is less than zero or (ii) the optimal value in problem (2) is exactly zero and the optimal value in problem (3) is less than zero.

The dimension of a polytope  $K \subset \mathbb{R}^k$  is defined to be the minimal dimension of a linear subspace  $L$  for which there exists a points  $z \in \mathbb{R}^k$  so that  $K \subset z + L$ . According to (6), the dimension of a polytope  $K$  with vertices  $z^1, \dots, z^a \in \mathbb{R}^k$  is the rank of the matrix

$$\begin{bmatrix} 1 & z_1^1 & \dots & z_k^1 \\ \vdots & \vdots & \vdots & \vdots \\ 1 & z_1^a & \dots & z_k^a \end{bmatrix}$$

minus one. A Delaunay triangulation contains polytopes of different dimensions. In (7), it is shown that polytopes with dimension equal to the number of objectives are not inherently nondominated. For this reason, we can remove these polytopes straight away and, thus, we let  $\mathcal{D}$  from now on be the Delaunay triangulation containing only the polytopes with dimension less than the number of objectives.

In (7), an approach to construct a  $\mathcal{D}$ -maximal inherently nondominated Pareto front approximation is proposed based on removing polytopes from the Delaunay triangulation. In this paper, we construct the approximation for problem (1) by removing polytopes from the triangulation  $\mathcal{D}$  according to the rules (R1) and (R2) given in (7). Assume that the  $a$ -polytopes (i.e., polytopes with  $a$  vertices) in the Delaunay triangulation  $\mathcal{D}$  are  $K^{a,1}, \dots, K^{a,t^a}$ . A polytope  $K^{a,j}$  is removed from the triangulation  $\mathcal{D}$ , if (R1) there exists a point  $p \in P$  that dominates or is dominated by the polytope or the polytope dominates itself or (R2) there exists a  $b$ -polytope  $K^{b,j'} \in \mathcal{D}$  with either  $b > a$  or ( $b = a$  and  $j' < j$ ) that is not removed from the triangulation and that dominates or is dominated by the polytope  $K^{a,j}$ .

Before we describe how the Pareto front is approximated, we briefly introduce NIMBUS. We use NIMBUS in this paper, because it was used also by (10) as it has been successfully applied in various design and planning problems (15). NIMBUS is an interactive multiobjective optimization method (12; 14; 11). In interactive methods (9; 11; 15), the decision maker expresses his/her preferences iteratively and before every iteration the decision maker is given additional information about the problem related to the preferences that he/she has given previously. This iterative procedure enables the decision maker to learn about the problem (15) and also the decision maker is able guide the search to the areas of the Pareto front that are most interesting to him/her.

Interactive methods differ by the types of preference information asked from the decision maker and by the ways this preference information is used. The NIMBUS method uses classification of objectives to indicate how the current solution should be changed; given a Pareto optimal solution to the multiobjective optimization problem the decision maker can classify the objective functions into classes  $I^<$ ,  $I^{\leq}(z)$ ,  $I^=$ ,  $I^{\geq}(z)$  and  $I^>$  defined, respectively, as classes of objective functions that the decision maker wants to improve as much as possible, wants to improve to the limit  $z$ , is allowed to remain unchanged, is allowed to deteriorate until limit  $z$  and is allowed to change freely for a while. This preference information is then converted into several different single-objective subproblems with the help of different scalarization functions as proposed in (14). These subproblems are then solved to generate different Pareto optimal outcomes which are then shown to the decision maker who can see how well the designed changes could be attained. The decision maker can choose any of these outcomes as the starting point of the next round of iteration. This iteration procedure can either start with an outcome given by the decision maker or from a so-called neutral outcome and it is repeated as long as the decision maker is satisfied with the outcome at hand. Further information of this so-called synchronous NIMBUS with other means to direct the search process is given in (14).

The WWW-NIMBUS<sup>®</sup> is a web based implementation of the NIMBUS method introduced in (13; 14). WWW-NIMBUS<sup>®</sup> is free for academic use at <http://nimbus.mit.jyu.fi/>.

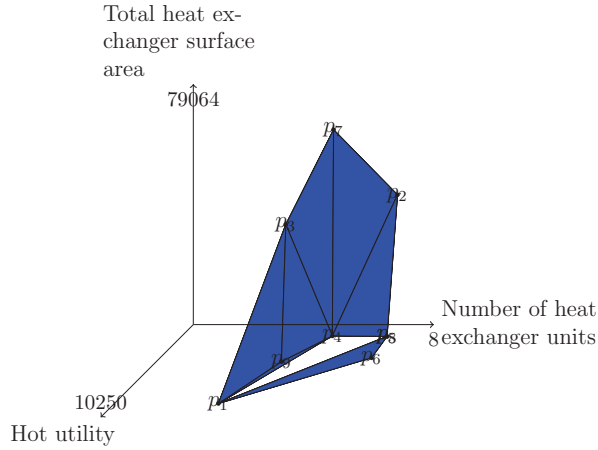
## 4 Approximation and Solution Process in Heat Exchanger Network Synthesis

In this section, we solve the heat exchanger network synthesis problem and for that we first generate an inherently nondominated Pareto front approximation for it. We assume we are given nine Pareto optimal outcomes (forming the set P) summarized in Table 1.

Table 1: The Pareto optimal outcomes for the multiobjective heat exchanger network synthesis problem

Point	Number of heat exchanger units	Total heat exchanger surface area	Hot utility consumption
$p^1$	4	1897	10250
$p^2$	8	52175	1755
$p^3$	5	52175	5371
$p^4$	6	4913	3028
$p^5$	8	4734	3028
$p^6$	8	2608	4920
$p^7$	6	79064	2882
$p^8$	8	4734	3028
$p^9$	5	4153	5902

Figure 1: The body of the  $\mathcal{D}$ -maximal inherently nondominated Pareto front approximation



The Deulaunay triangulation of this set  $P$  contains 75 polytopes. Twelve of these polytopes are three-dimensional and they are removed as explained before, because the number of objectives is three in this case. By solving optimization problems (2) and (3) for each pair of polytopes in the complex  $\mathcal{D}$  we determine which polytopes dominate each other and by following rules (R1) and (R2) one can see which polytopes are removed from the triangulation  $\mathcal{D}$ . Solving problems (2) and (3) for each pair is easily implemented on any programming language and here we have used GNU Octave (3). The polytopes in the resulting  $\mathcal{D}$ -maximal inherently nondominated Pareto front approximation are summarized in Table 2 and the body of the resulting approximation is drawn in Figure 1.

Table 2: The polytopes in the  $\mathcal{D}$ -maximal inherently nondominated Pareto front approximation

Polytope	Vertices	Polytope	Vertices	Polytope	Vertices
$\mathcal{P}_1$	$p^6$	$\mathcal{P}_2$	$p^7$	$\mathcal{P}_3$	$p^2$
$\mathcal{P}_4$	$p^1$	$\mathcal{P}_5$	$p^3$	$\mathcal{P}_6$	$p^4$
$\mathcal{P}_7$	$p^8$	$\mathcal{P}_8$	$p^9$	$\mathcal{P}_9$	$p^6 p^1$
$\mathcal{P}_{10}$	$p^7 p^2$	$\mathcal{P}_{11}$	$p^3 p^7$	$\mathcal{P}_{12}$	$p^3 p^1$
$\mathcal{P}_{13}$	$p^3 p^4$	$\mathcal{P}_{14}$	$p^4 p^7$	$\mathcal{P}_{15}$	$p^4 p^2$
$\mathcal{P}_{16}$	$p^8 p^6$	$\mathcal{P}_{17}$	$p^8 p^2$	$\mathcal{P}_{18}$	$p^8 p^4$
$\mathcal{P}_{19}$	$p^8 p^1$	$\mathcal{P}_{20}$	$p^9 p^4$	$\mathcal{P}_{21}$	$p^9 p^1$
$\mathcal{P}_{22}$	$p^4 p^1$	$\mathcal{P}_{23}$	$p^9 p^3$	$\mathcal{P}_{24}$	$p^3 p^4 p^7$
$\mathcal{P}_{25}$	$p^4 p^7 p^2$	$\mathcal{P}_{26}$	$p^8 p^4 p^2$	$\mathcal{P}_{27}$	$p^8 p^6 p^1$
$\mathcal{P}_{28}$	$p^9 p^4 p^1$	$\mathcal{P}_{29}$	$p^9 p^3 p^1$	$\mathcal{P}_{30}$	$p^9 p^3 p^4$

Now that we have formulated an inherently nondominated approximation we can move to the next phase and show how to use it for decision making. Through a parametrization of the approximation, the multiobjective problem of choosing the preferred point on the approximation  $\mathcal{A} = \{\mathcal{P}_1, \dots, \mathcal{P}_{30}\}$  can be formulated as a mixed-integer problem with three objectives

$$\begin{aligned} \min \quad & f(\lambda, t) = \begin{aligned} & t_1(\lambda_1 p^3 + \lambda_2 p^4 + \lambda_3 p^7) + t_2(\lambda_1 p^4 + \lambda_2 p^7 + \lambda_3 p^2) \\ & + t_3(\lambda_1 p^8 + \lambda_2 p^4 + \lambda_3 p^2) + t_4(\lambda_1 p^8 + \lambda_2 p^6 + \lambda_3 p^1) \\ & + t_5(\lambda_1 p^9 + \lambda_2 p^4 + \lambda_3 p^1) + t_6(\lambda_1 p^9 + \lambda_2 p^3 + \lambda_3 p^1) \\ & + t_7(\lambda_1 p^9 + \lambda_2 p^3 + \lambda_3 p^4) \end{aligned} \\ \text{s.t.} \quad & \sum_{i=1}^3 \lambda_i = 1 \\ & \sum_{i=1}^7 t_i = 1 \\ & \lambda \in [0, 1]^3 \\ & t \in \{0, 1\}^7. \end{aligned}$$

The above formulation is imputed into the WWW-NIMBUS<sup>®</sup> implementation of the NIMBUS method and the most preferred point on the approximation can be found with NIMBUS.

At this point the decision maker is involved. The iterations of the solution process with WWW-NIMBUS<sup>®</sup> can be seen Table 4. The chosen alternative on each iteration is underlined.

Table 3: The course of the NIMBUS optimization procedure

Iter.	Issue	Number of heat exchanger units	Total heat exchanger surface area	Hot utility consumption
0	Starting point	6.161743	3431.044	6347.206
	Classification	$I \leq (5)$	$I \geq (8000)$	$I <$
1	Alternative 1	5.829885	4783.843	3517.424
	Alternative 2	8.0	52175.59	1755.5
	<u>Alternative 3</u>	6.0	4913.04	3028.43
	Alternative 4	5.392376	4458.865	4774.958
2	Classification	$I =$	$I \geq (6000)$	$I \leq (2500)$
	Alternative 1	7.162517	8383.323	2932.537
	Alternative 2	6.153829	8548.212	2930.523
	Alternative 3	6.117182	7682.189	2953.848

Because the decision maker sees that the desired changes in iteration 2 were not possible, alternative 3 of iteration 1 is selected as the preferred outcome. During the solution process the decision maker is interested in reducing the hot utility consumption near 3000 with as few as possible heat exchanger units and as small as possible surface area. The true Pareto optimal outcome corresponding to the selected approximate outcome is obtained by using the achievement scalarizing function (18). The achievement scalarizing function is

maximized with the reference point as the preferred point on the approximation and this yields a Pareto optimal outcome to the initial problem with the number of heat exchanger units 6.00, total heat exchanger surface area 4791.71 and hot utility consumption 3043.59. The solution related to this outcome is taken as the final outcome since it is satisfactory to the decision maker. However, if this was not the case we could have added this point to the set  $P$  and repeated the approximation procedure to get a more accurate approximation.

## 5 Conclusions

This paper shows an application of a new Pareto front approximation approach to solving a heat exchanger network synthesis problem. In the solution process the Pareto front is first approximated and then the decision maker can find the most preferred approximated outcome with an interactive method, here interactive multiobjective optimization method NIMBUS. Finally, the preferred approximated outcome is projected on the true Pareto front.

The method has many advantages. First of all, because the starting point of the approximation, that is, the finite set of Pareto optimal outcomes  $P$  is computed before involving the decision maker, one can even use time consuming global optimization methods as in this case. Since the approximation has analytical error estimates given in (7), one can find out how good the approximation is. After having computed the Pareto front approximation it is possible to use any multiobjective optimization method to study the approximation and search for a preferred outcome. Because the parametrized problem formed based on the approximation is computationally inexpensive, the decision maker does not have to wait for new solutions being generated. In this paper, it was shown that NIMBUS can be used for that. Finally, the preferred outcome on the approximation can be projected on the real Pareto front with achievement scalarizing function (18).

## 6 Acknowledgments

This research was partly supported by the Academy of Finland, grant number 128495. The authors would like to thank MSc Vesa Ojalehto for his significant help with providing the Pareto optimal points and for his expertise with the GAMS model. The authors would also like to thank MEng Timo Laukkanen for the GAMS model that was used in this paper.

## References

- [1] J. Branke, K. Deb, K. Miettinen, and K. Slowinski, editors. *Multiobjective Optimization: Interactive and Evolutionary Approaches*. Springer, Berlin, 2008.
- [2] J. Cerda and A. W. Westerberg. Synthesizing Heat Exchanger Networks Having Restricted Stream/Stream Matches Using Transportation. *Chemical Engineering Science*, 38:1723–1740, 1983.

- [3] John W. Eaton. *GNU Octave Manual*. Network Theory Limited, 2002.
- [4] P. Eskelinen, K. Miettinen, K. Klamroth, and J. Hakanen. Pareto Navigator for Interactive Nonlinear Multiobjective Optimization. *OR Spectrum*, 23:211–227, 2010.
- [5] J. E. Goodman and J. O’Rourke, editors. *Discrete and Computational Geometry*. CRC Press, Boca Raton, 1997.
- [6] B. Grünbaum. *Convex Polytopes*. Interscience Publishers, London, 1967.
- [7] M. Hartikainen, K. Miettinen, and M. M. Wiecek. A Pareto Front Approximation with Inherent Non-Dominance and the Delaunay Triangulation, 2010. University of Jyväskylä, Reports of the Department of Mathematical Information Technology, Series B. Scientific Computing, Number B5/2010.
- [8] M. Hartikainen, K. Miettinen, and M. M. Wiecek. Inherent Non-Dominance — Pareto Front Approximations for Decision Making, 2010. University of Jyväskylä, Reports of the Department of Mathematical Information Technology, Series B. Scientific Computing, Number B2/2010.
- [9] C. Hwang and A. S. M. Masud. *Multiple Objective Decision Making - Methods and Applications: a State-of-the-Art Survey*. Springer, Berlin, 1979.
- [10] T. Laukkanen, T.-M. Tveit, V. Ojalehto, K. Miettinen, and C.-J. Fogelholm. An Interactive Multi-Objective Approach to Heat Exchanger Network Synthesis. *Computers & Chemical Engineering*, 34:943–952, 2010.
- [11] K. Miettinen. *Nonlinear Multiobjective Optimization*. Kluwer Academic Publishers, Boston, 1999.
- [12] K. Miettinen and M. Mäkelä. Interactive Bundle-based Method for Nondifferentiable Multiobjective Optimization: NIMBUS. *Optimization*, 34:231–246, 1995.
- [13] K. Miettinen and M. Mäkelä. Interactive Multiobjective Optimization System WWW-NIMBUS on the Internet. *Computers & Operations Research*, 27:709–723, 2000.
- [14] K. Miettinen and M. M. Mäkelä. Synchronous Approach in Interactive Multiobjective Optimization. *European Journal of Operational Research*, 170:909–922, 2006.
- [15] K. Miettinen, F. Ruiz, and A. Wierzbicki. Introduction to Multiobjective Optimization: Interactive Approaches. In J. Branke, K. Deb, K. Miettinen, and R. Slowinski, editors, *Multiobjective Optimization: Interactive and Evolutionary Approaches*, pages 27–57. Springer, Berlin, 2008.
- [16] S. A. Papoulias and I. E. Grossmann. A Structural Optimization Approach to Process Synthesis - II: Heat Recovery Networks. *Computers & Chemical Engineering*, 7:707–721, 1983.

- [17] S. Ruzika and M. M. Wiecek. Approximation Methods in Multiobjective Programming. *Journal of Optimization Theory and Applications*, 126:473–501, 2005.
- [18] A. Wierzbicki. On the Completeness and Constructiveness of Parametric Characterizations to Vector Optimization Problems. *OR Spectrum*, 8:73–87, 1986.
- [19] T. F. Yee and I. E. Grossmann. Simultaneous Optimization of Models for Heat Integration - II. Heat Exchanger Network Synthesis. *Computers and Chemical Engineering*, 14:1165–1184, 1990.

**PIV**

**PAINT: PARETO FRONT INTERPOLATION FOR NONLINEAR  
MULTIOBJECTIVE OPTIMIZATION**

by

Markus Hartikainen · Kaisa Miettinen · Margaret M. Wiecek

*Computational Optimization and Applications*, To Appear.

DOI: 10.1007/s10589-011-9441-z

Reproduced with kind permission of Springer Science+Business Media.



# PAINT: Pareto Front Interpolation for Nonlinear Multiobjective Optimization

Markus Hartikainen · Kaisa Miettinen ·  
Margaret M. Wiecek

Received: date / Accepted: date

**Abstract** A method called PAINT is introduced for computationally expensive multiobjective optimization problems. The method interpolates between a given set of Pareto optimal outcomes. The interpolation provided by the PAINT method implies a mixed integer linear surrogate problem for the original problem which can be optimized with any interactive method to make decisions concerning the original problem. When the scalarizations of the interactive method used do not introduce nonlinearity to the problem (which is true e.g., for the synchronous NIMBUS method), the scalarizations of the surrogate problem can be optimized with available mixed integer linear solvers. Thus, the use of the interactive method is fast with the surrogate problem even though the problem is computationally expensive. Numerical examples of applying the PAINT method for interpolation are included.

**Keywords** Multiobjective optimization · Interactive decision making · Computationally expensive problems · Approximation

---

This research was partly supported by the Academy of Finland (grant number 128495) and Jenny and Antti Wihuri foundation

---

M. Hartikainen

Department of Mathematical Information Technology, P.O. Box 35 (Agora), 40014 University of Jyväskylä, Finland

E-mail: markus.e.hartikainen@jyu.fi

K. Miettinen

Department of Mathematical Information Technology, P.O. Box 35 (Agora), 40014 University of Jyväskylä, Finland

E-mail: kaisa.miettinen@jyu.fi

M. M. Wiecek

Department of Mathematical Sciences, Clemson University, U.S.A.

E-mail: wmalgor@clemson.edu

## 1 Introduction

Multiobjective optimization means optimizing multiple conflicting objectives at the same time [see e.g., 23]. Multiobjective optimization problems may have many Pareto optimal solutions, whose objectives cannot be improved without impairing some other objectives. A vector containing the objective values for a single solution is called an outcome and the set of outcomes given by Pareto optimal solutions is often called the Pareto front. The ultimate aim of multiobjective optimization is to help a decision maker find a solution that is preferable to him/her. A decision maker is a person who has the right or is expected to make decisions concerning the real life problem that is mathematically modeled by the multiobjective optimization problem. Here we do not assume that he/she has any knowledge about different multiobjective optimization methods.

One way to classify different multiobjective optimization methods, introduced in [15] and followed in [23], is based on the relative order of the decision maker's preference articulation and optimization. In a priori methods, the preferences of the decision maker are first specified and then a single Pareto optimal solution is found with respect to these and in a posteriori methods many Pareto optimal solutions are generated and then the decision maker is expected to choose a preferred one among them. If the decision maker's preferences do not play a role or are unavailable then the method is called a no-preference method. Finally, interactive methods employ an iterative procedure and allow the decision maker to correct his/her preferences and also enable the decision maker to learn about the problem [see e.g., 28]. Interactive methods have in many instances been seen as the most prominent [8, 20, 23, 28], because they allow the decision maker to gain more insight about the problem while solving it without introducing too much cognitive load at a time. In other words, the decision maker can consider only those Pareto optimal solutions that have been generated based on his/her preferences. We have, thus, focused on interactive methods in our research.

In interactive methods, new solutions to the multiobjective optimization problem are often produced through scalarization, i.e., by converting the multiobjective optimization problem into single objective optimization problems [20]. Different interactive methods use different scalarizations and some interactive methods even use multiple scalarizations (e.g., the synchronous NIMBUS method introduced in [27]). In this paper, a scalarization of the multiobjective optimization problem refers to a single objective optimization problem whose optimal solutions are solutions to the multiobjective optimization problem. Usually, scalarizations include a way to take into account the decision maker's preferences (as e.g., in achievement scalarizing problem (8) from [33], where the reference point contains aspiration levels for the objectives). For more information about scalarizations, see e.g., [23, 26].

A drawback with iterative procedures of interactive methods is that the decision maker has to wait while new Pareto optimal solutions are generated with his/her updated preferences. This is usually done in each iteration.

Since many real-life problems are computationally expensive [see e.g., 14, 18], scalarizations of the problems may also be computationally expensive. By a computationally expensive multiobjective optimization problem we here mean a problem that requires a long time to compute the objective values for a given solution. For these problems, the time that is needed to optimize the scalarizations may be too long for the decision maker to spare. Besides, the decision maker may become unwilling to explore different solutions to the problem in the so-called learning phase (in which the most interesting region on the Pareto front is to be identified [28]).

In this paper, we develop and implement a Pareto front approximation method PAINT (PAreto front INTerpolation) that interpolates between a set of given Pareto optimal outcomes. The interpolation satisfies the property of inherent nondominance introduced in [13] and, thus, implies a surrogate problem that can replace the computationally expensive original problem and can be solved with any interactive method to yield a preferred solution for the original problem. The inherent nondominance guarantees that none of the interpolants dominate or are dominated by any of the given Pareto optimal outcomes, neither do they dominate each other. The interpolation is based on the results developed in [12], where it was suggested that a certain subcomplex of the Delaunay triangulation of the given Pareto optimal outcomes should be used. Our surrogate problem allows the decision maker to consider approximate outcomes (i.e., interpolants of the given Pareto optimal outcomes produced by the PAINT method). In this paper, we also develop a mixed integer linear formulation for the surrogate problem that is implied by the interpolation. Replacing the original computationally expensive problem with the surrogate problem will naturally offer computational time savings when an interactive method is used.

Approaches similar to ours, where a Pareto front approximation is used to search for interesting outcomes not limited to a given set, have been proposed in [8, 19, 29]. The main difference between these approaches and ours is that the others use their own tailor-made methods for finding a vector containing preferred approximate outcomes on the approximations, while our approximation aims to work (through the mixed integer linear surrogate problem) in concert with any interactive method. Furthermore, [8, 29] are only applicable to convex multiobjective optimization problems, while ours can handle nonconvex problems. As mentioned before, PAINT has been developed to produce interpolate outcomes to be used with an interactive method through the surrogate problem. However, when looking at PAINT as a plain approximation method, it can be compared to some existing methods in the literature. This kind of approximation methods can be found in a survey [30] and in papers [2, 7, 9, 16, 21]. These methods, however, do not concentrate on the question of how to choose a Pareto optimal solution on the produced approximation as we do. Also, none of the above methods guarantee the property of inherent nondominance for nonconvex optimization problems with more than two objectives.

The rest of this paper is structured as follows: In Section 2, notation and definitions are given. The PAINT method and related computational issues are described in Section 3. In Section 4, the surrogate multiobjective optimization problem implied by the approximation is discussed. Section 5 shows examples of different Pareto front approximations to demonstrate the versatility of the PAINT method. Finally, conclusions are drawn in Section 6.

## 2 Notation and definitions

In this paper, we study multiobjective optimization problems

$$\begin{aligned} \min & (f_1(x), \dots, f_k(x)) \\ \text{s.t. } & x \in S, \end{aligned} \tag{1}$$

where the integer  $k$  is the number of real-valued objectives. The set  $S$  is called the feasible set. A vector  $z = f(x) = (f_1(x), \dots, f_k(x))^T$  with  $x \in S$  is called an outcome. The set  $f(S)$  is called the outcome set. We assume that the set  $S$  is connected and that the objective functions  $f_i$  are continuous. This assumption is made in order to justify interpolating between the objective function values of Pareto optimal solutions.

The set  $\mathbb{R}^k$  is called the outcome space. For a vector  $z$  in the outcome space, a component  $z_i$  is called an objective value. Because problem (1) is a minimization problem, less is preferred to more in each objective. A vector  $z^1$  in the outcome space is said to dominate another vector  $z^2$  in the outcome space if  $z^1$  is at least as good as  $z^2$  in all objectives and strictly better in at least one. If  $z^1$  dominates  $z^2$  then it is written  $z^1 \leq z^2$ .

An outcome  $z^1$  is said to be Pareto optimal, if there does not exist another outcome  $z^2$  so that  $z^2$  dominates  $z^1$ . The set of all Pareto optimal outcomes is called the Pareto front. In this paper, the set  $P$  always refers to the given set of Pareto optimal outcomes on which the Pareto front approximation is based. The number of these Pareto optimal outcomes is denoted by  $m$  and the elements of this set are denoted by  $p^j$ , where  $j = 1, \dots, m$ . This set is called the initial set of Pareto optimal outcomes and it is taken as given, i.e., we assume that we cannot influence the locations of these outcomes in the outcome space.

Important elements in our PAINT method are polytopes and collections of polytopes for which we mostly follow the definitions given in [10]: For a nonnegative integer  $a$ , the convex hull of  $z^1, \dots, z^{a+1} \in \mathbb{R}^k$  is denoted by  $\mathcal{P}(z^1, \dots, z^{a+1})$  and is called an  $a$ -polytope. It is said that the polytope  $\mathcal{P}(z^1, \dots, z^{a+1})$  is defined by vectors  $z^1, \dots, z^{a+1}$ . A vertex of a polytope  $K$  is a point  $x \in K$  for which it holds that  $\lambda z + (1-\lambda)y = x$ ,  $\lambda \in (0, 1)$  and  $z, y \in K$  imply  $x = y = z$ . The set of vertices of a polytope  $K$  is denoted by  $\text{vert}(K)$ . A face of a polytope  $K$  is another polytope  $K'$  so that  $K' = \emptyset$ ,  $K' = K$  or there exist vectors  $z^1, z^2 \in \mathbb{R}^k$  so that  $K' = \{z^1 + h : h \in \mathbb{R}^k, h^T z^2 = 0\} \cap K$ . A polytope  $\mathcal{P}(z^1, \dots, z^{a+1})$  is a simplex if the vectors  $z^1, \dots, z^{a+1}$  are affinely independent.

Collections of polytopes are sets whose elements are polytopes and they are denoted by calligraphic letters  $\mathcal{A}, \mathcal{B}, \mathcal{K}, \dots$ . The body of a collection of polytopes  $\mathcal{A}$  is  $\text{body}(\mathcal{A}) = \cup_{K \in \mathcal{A}} K$ . A (polyhedral) complex  $\mathcal{K}$  is a special type of collection of polytopes so that (a) if it holds that a polytope  $K \in \mathcal{K}$  and another polytope  $K'$  is a face of  $K$  then it must hold that  $K' \in \mathcal{K}$ , and (b) if it holds that polytopes  $K^1, K^2 \in \mathcal{K}$  then the set  $K^1 \cap K^2$  must be a possibly empty face of both polytopes  $K^1$  and  $K^2$ . A triangulation of a finite set  $P \subset \mathbb{R}^k$  is a complex  $\mathcal{T}$  so that the body of  $\mathcal{T}$  is the convex hull of the set  $P$  and the set of vertices in  $\mathcal{T}$  is  $P$ . A polytope  $\mathcal{P}(z^1, \dots, z^k)$  with  $z^1, \dots, z^k \in P$  is called Delaunay (in  $P$ ) if there exists an open ball  $B \in \mathbb{R}^k$  with  $\text{cl}(B) \cap P = \{z^1, \dots, z^{a+1}\}$  (here  $\text{cl}(B)$  denotes the closure of  $B$ ) and  $B \cap P = \emptyset$ . A complex  $\mathcal{D}$  is a Delaunay triangulation of  $P$  if  $\mathcal{D}$  is a triangulation of  $P$  so that every polytope in  $\mathcal{D}$  is Delaunay. A polyhedral complex is called a simplicial complex, if all the polytopes are simplices.

In [13], the most important concepts for the theory behind our Pareto front approximation were given. First, a set  $A \subset \mathbb{R}^k$  is called *inherently nondominated*, if there does not exist  $a, b \in A$  so that  $a \leq b$ . Given a set of outcomes, one can check whether any of them dominates the others. Inherent nondominance is a more advanced concept used to guarantee that when a new infinite set of interpolating outcomes has been generated, none of them is dominated by or dominates others. In order to connect the inherently nondominated set to a known set of Pareto optimal outcomes  $P$  it was defined that an inherently nondominated set  $A$  is called an *inherently nondominated Pareto front approximation* (based on a set of outcomes  $P$ ), if it holds that  $P \subset A$ . Since also the set  $P$  is by definition an inherently nondominated Pareto front approximation, it was further defined that for a set  $B \subset \mathbb{R}^k$ , an inherently nondominated Pareto front approximation  $A \subset B$  is called a  $B$ -maximal inherently nondominated Pareto front approximation, if there does not exist an inherently nondominated set  $A'$  so that  $A \subsetneq A' \subset B$ . A  $B$ -maximal inherently nondominated Pareto front approximation is a maximal (by inclusion) collection of points from  $B$  that could be a part of a same Pareto front as the initial Pareto optimal outcomes in  $P$  are.

Since an inherently nondominated Pareto front approximation is to be used as an interpolation of the Pareto optimal outcomes in  $P$ , it usually contains an infinite number of points. On the other hand, such an infinite set can be represented by a finite number of polytopes. Thus, it is easier to construct such interpolations from polytopes than from points. Hence, the above concepts were in [12] slightly modified to fit collections of polytopes and polyhedral complexes. A collection of polytopes  $\mathcal{K}$  is called inherently nondominated and an inherently nondominated Pareto front approximation, whenever its body,  $\text{body}(\mathcal{K})$ , is that. Finally, if  $\mathcal{B}$  is a collection of polytopes, an inherently nondominated collection of polytopes  $\mathcal{K}$  is called a  $\mathcal{B}$ -maximal inherently nondominated Pareto front approximation if there does not exist a polytope  $K \in \mathcal{B} \setminus \mathcal{K}$  so that the collection of polytopes  $\mathcal{K} \cup \{K\}$  is inherently nondominated. Thus, the difference between this definition and the corresponding definition for a set is that a  $\mathcal{B}$ -maximal inherently nondominated Pareto front approximation

is a maximal collection of polytopes (not points) in  $\mathcal{B}$  that could be a part of a same Pareto front as the initial Pareto optimal outcomes in  $P$  are. A member of the body of an inherently nondominated Pareto front approximation is called an *approximate Pareto optimal outcome*.

To construct inherently nondominated collections of polytopes, a dominance between polytopes was defined in [12]. Let  $K^1, K^2 \subset \mathbb{R}^k$  be polytopes. It is defined that the polytope  $K^1$  dominates the polytope  $K^2$  if there exist vectors  $s^1 \in K^1$  and  $s^2 \in K^2$  so that  $s^1$  dominates  $s^2$ . As shown in that paper, a complex  $\mathcal{K} \subset \mathcal{D}$  is inherently nondominated if and only if there does not exist polytopes  $K^1, K^2 \in \mathcal{K}$  so that  $K^1$  dominates  $K^2$ .

### 3 The PAINT method

In this section, we introduce the algorithm of the PAINT method, to be called Algorithm 1. We describe data structures that can be used in implementing this algorithm. Based on these data structures, we illustrate how the steps of Algorithm 1 can be carried out. We also discuss the implementation aspects of the algorithm and the complexity of the PAINT method. Finally, we develop a way of resolving dominance between two polytopes, which is needed in Algorithm 1.

The PAINT method takes as input a set of  $m$  Pareto optimal outcomes  $P$  and its output is a  $\mathcal{D}$ -maximal inherently nondominated Pareto front approximation based on  $P$ , where  $\mathcal{D}$  is a Delaunay triangulation of the set  $P$ . The method is based on the ideas presented in [12] and it guarantees that the approximation satisfies the following rules (R1) and (R2) given in [12]: assuming that  $l$ -polytopes in the Delaunay triangulation  $\mathcal{D}$  are  $K^{l,j}$  with  $j = 1, \dots, t^l$ , a polytope  $K^{l,j} \in \mathcal{D}$  is removed, if either

- (R1) there exists an outcome  $p \in P$  that dominates or is dominated by the polytope  $K^{l,j}$  or the polytope  $K^{l,j}$  dominates itself
- OR
- (R2) there exists an  $m$ -polytope  $K^{m,j'} \in \mathcal{D}$  with either  $m > l$  or  $m = l$  and  $j' < j$  that is not removed, and that dominates or is dominated by the polytope  $K^{l,j}$ .

In [12], it is shown that a resulting polyhedral complex after having applied rules (R1) and (R2) is a  $\mathcal{D}$ -maximal inherently nondominated Pareto front approximation.

In the implementation of the PAINT method presented in this paper, the Delaunay triangulation and the inherently nondominated Pareto front approximation are represented as matrices  $D$  and  $A$ , respectively. In those matrices, each row represents the vertices of a polytope with each entry referring to an outcome in  $P$ . For example a row with entries 1, 2 and 3 represents a polytope defined by outcomes  $p^1, p^2$  and  $p^3$  in  $P$ . Polytopes that are defined with fewer outcomes than there are columns in  $D$  or  $A$  are handled by repeating the same outcome multiple times. This way of representing the complexes enables us to

write e.g., the outcomes defining the polytope represented by row  $j$  in the matrix  $D$  as  $p^{D_{j,1}}, p^{D_{j,2}}, \dots, p^{D_{j,b}}$  with  $b$  being the number of columns in  $D$ .

### 3.1 Algorithm

The PAINT method is described in Algorithm 1. Throughout the algorithm, indices  $a$  and  $b$  stand for the number of polytopes in the Delaunay triangulation and the maximal number of outcomes defining a polytopes in the Delaunay triangulation, respectively. The algorithm begins by reading in the initial set of Pareto optimal outcomes  $P$ . Line 2 of the algorithm is concerned with constructing a Delaunay triangulation of the set  $P$ . We propose to do this with the Quickhull algorithm [1], which constructs the Delaunay triangulation by building the convex hull of a higher dimensional related set (see [3]). The Quickhull algorithm assumes that the outcomes in  $P$  are in general position (see [1]), which is not always the case for e.g., linear multiobjective optimization problems for which the outcomes may be affinely dependent. This dependency can be removed by e.g., slightly perturbing the points as proposed in [6]. Whenever necessary, we have used perturbation in the examples in Section 5.

In line 3 of Algorithm 1, the rows of matrix  $D$  representing the Delaunay triangulation are ordered in a descending order with respect to the number of different entries in the row that is defined as the number of outcomes defining the polytope that is represented by the row. The ordering ensures that the polytopes are checked in the loop starting from line 5 in a descending order with respect to the number of outcomes defining the polytope.

In lines 6, 11 and 19 of the algorithm, there are three different if-conditions. The purpose of the conditions in lines 6 and 11 is to make sure that the final approximation follows rule (R1), and the purpose of the condition in line 19 is to make sure that the final approximation follows rule (R2). Conveniently, all these conditions reduce to checking dominance between two polytopes, which is the condition in line 19: (a) the inherent nondominance of a polytope in line 6 can be reduced to checking whether a polytope dominates itself, because in [12] it was proven that a polytope is inherently nondominated if and only if it does not dominate itself, and (b) the dominance between an outcome and a polytope in line 11 is actually dominance between two polytopes because a singleton containing a vector in  $\mathbb{R}^k$  is by definition a polytope. Thus, it is adequate to build mathematical tools for determining dominance between two polytopes. This is done in Subsection 3.2 by using two linear optimization problems.

Assume now that the algorithm is in the loop starting from row 5 and ending with row 26 and that the index  $i$  value is  $\hat{i}$ . If the polytope given by row  $\hat{i}$  in  $D$  is not inherently nondominated, it dominates or is dominated by an outcome in  $P$  or it dominates or is dominated by a polytope given by any of rows  $d+1, \dots, \hat{i}-1$  in the current  $D$ , then the polytope is removed from  $D$  in lines 7, 12 and 20, respectively. This is done by increasing  $d$  (the number of polytopes that have been already removed) and interchanging rows  $d$  and  $\hat{i}$  in

---

**Algorithm 1** PAINT method: Construction of the inherently nondominated Pareto front approximation
 

---

```

1: Read the Pareto optimal outcomes  $P = \{p^1, p^2, \dots, p^m\} \subset \mathbb{R}^k$ .
2:  $D \leftarrow$  the Delaunay triangulation of  $P$ 
3: Sort the rows of  $D$  in descending order w.r.t. number of different entries in the row
4:  $a, b \leftarrow$  the number of rows and columns of  $D$ , respectively,  $d \leftarrow 0$ 
5: for  $i = 1$  to  $a$  do
6:   if the polytope given by vertices  $p^{D_{i,1}}, p^{D_{i,2}}, \dots, p^{D_{i,b}}$  is not inherently nondominated then
7:      $d \leftarrow d + 1$  and interchange rows  $i$  and  $d$  of the matrix  $D$ 
8:   else
9:      $deleted \leftarrow \text{false}$ 
10:    for  $j = 1$  to  $m$  do
11:      if the polytope given by vertices  $p^{D_{i,1}}, p^{D_{i,2}}, \dots, p^{D_{i,b}}$  dominates or is dominated by outcome  $p^j$  then
12:         $d \leftarrow d + 1$  and interchange rows  $i$  and  $d$  of the matrix  $D$ 
13:         $deleted \leftarrow \text{true}$ 
14:        Break
15:      end if
16:    end for
17:    if not  $deleted$  then
18:      for  $l = d + 1$  to  $i - 1$  do
19:        if the polytope given by vertices  $p^{D_{i,1}}, p^{D_{i,2}}, \dots, p^{D_{i,b}}$  dominates or is dominated by the polytope given by vertices  $p^{D_{l,1}}, p^{D_{l,2}}, \dots, p^{D_{l,b}}$  then
20:           $d \leftarrow d + 1$  and interchange rows  $i$  and  $d$  of the matrix  $D$ 
21:          Break
22:        end if
23:      end for
24:    end if
25:  end if
26: end for
27:  $A \leftarrow \begin{bmatrix} D_{d+1,1} & \dots & D_{d+1,b} \\ \vdots & & \vdots \\ D_{a,1} & \dots & D_{a,b} \end{bmatrix}$ 
28: return  $A$ 

```

---

the matrix  $D$ . Because in line 27 only rows in  $D$  from row  $d + 1$  onwards are set as matrix  $A$ , such a polytope will not be a part of the approximation. If a polytope is not removed as described above, the row representing this polytope will be inserted into matrix  $A$  (representing the inherently nondominated Pareto front approximation) in line 27.

Finally, let us consider the complexity of the proposed implementation of Algorithm 1. According to [1], the worst case complexity of the construction of the Delaunay triangulation with the Quickhull algorithm is  $O((k + 1) \log(m))$  for  $k \leq 2$  and  $O((k + 1)c_m/m)$  for  $k \geq 3$ , where

$$c_m = O(m^{\lfloor (k+1)/2 \rfloor} / \lfloor (k+1)/2 \rfloor!).$$

According to [22], a Delaunay triangulation of the set  $P$  contains at most  $O(m^{\lceil k/2 \rceil})$  polytopes, where  $m$  is the number of outcomes in  $P$  and  $k$  is the number of objectives of the multiobjective optimization problem. Thus, one



may have to resolve  $O(m^k)$  dominance relations between polytopes to determine which of the polytopes are to be removed. This means that resolving the dominance relations between polytopes is computationally the most expensive part of the PAINT method. Since in Subsection 3.2 it is shown that the dominance between polytopes can be resolved by solving two linear optimization problems, the worst case complexity of the PAINT method is  $O(m^k)$  in linear optimization problems.

The complexity of solving the linear optimization problems of Subsection 3.2 depends on the number of objectives and the number of variables. In these problems, the numbers of objectives and variables depend linearly on the maximal number of outcomes defining a polytope in the approximation and on the number of objectives. In practical problems, the number of objectives is usually rather low, and if one perturbs the Pareto optimal outcomes before constructing the Delaunay triangulation (as suggested in [6]), the maximal number of outcomes defining a polytope is the number of objectives plus one. Since these numbers are so low, also the linear optimization problems defined in Subsection 3.2 are fairly small and quick to solve. Consequently, we may here assume that in practice the complexity of solving a single linear optimization problem within the algorithm is low and independent of the numbers of objectives and variables. With current personal computers, a feasible number of objectives is about 10 and the maximal number of Pareto optimal outcomes is about 200.

### 3.2 Resolving dominance between two polytopes

According to [12], a polytope  $K^1 \subset \mathbb{R}^k$  dominates another polytope  $K^2 \subset \mathbb{R}^k$  (which needs to be resolved in lines 6, 11 and 19 of Algorithm 1) if and only if one of the following holds:

(i) the optimal value in problem

$$\begin{aligned} \min & \max_{i=1, \dots, k} (s_i^1 - s_i^2) \\ \text{s.t. } & s^1 \in K^1, s^2 \in K^2 \end{aligned} \quad (2)$$

is less than zero

OR

(ii) the optimal value in problem (2) is exactly zero and the optimal value in problem

$$\begin{aligned} \min & \sum_{i=1}^k (s_i^1 - s_i^2) \\ \text{s.t. } & s^1 \in K^1, s^2 \in K^2 \\ & s_i^1 \leq s_i^2 \text{ for all } i = 1, \dots, k. \end{aligned} \quad (3)$$

is less than zero.

In order to efficiently solve the above problems, it is beneficial to build linear programs that are equivalent to them. Assuming that the polytope  $K^1$  is given by row  $r_1$  in the matrix  $D$  (representing the Delaunay triangulation) and the polytope  $K^2$  is given by row  $r_2$  in  $D$ , we introduce the following

matrix  $B \in \mathbb{R}^{k \times 2b}$  (with  $b$  being the number of columns in  $D$  and  $k$  being the number of objectives of the problem)

$$B = \begin{bmatrix} p_1^{D_{r_1,1}} & p_1^{D_{r_1,2}} & \dots & p_1^{D_{r_1,b}} & -p_1^{D_{r_2,1}} & \dots & -p_1^{D_{r_2,b}} \\ p_2^{D_{r_1,1}} & p_2^{D_{r_1,2}} & \dots & p_2^{D_{r_1,b}} & -p_2^{D_{r_2,1}} & \dots & -p_2^{D_{r_2,b}} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ p_k^{D_{r_1,1}} & p_k^{D_{r_1,2}} & \dots & p_k^{D_{r_1,b}} & -p_k^{D_{r_2,1}} & \dots & -p_k^{D_{r_2,b}} \end{bmatrix}.$$

In matrix  $B$ , the first  $b$  columns include the components of the outcomes defining the polytope  $K^1$  and the last  $b$  columns include the opposites of the components of the outcomes defining the polytope  $K^2$ . Using the matrix  $B$  we can give the following representation for problem (2):

$$\begin{aligned} \min \quad & t \\ \text{s.t.} \quad & [B, -\mathbf{1}] \begin{bmatrix} \lambda \\ \mu \\ t \end{bmatrix} \leq \begin{bmatrix} 0 \\ \vdots \\ 0 \end{bmatrix} \\ & \sum_{j=1}^b \lambda_j = 1, \sum_{j=1}^b \mu_j = 1 \\ & \lambda \in [0, 1]^b, \mu \in [0, 1]^b, t \in \mathbb{R}, \\ \text{where} \quad & -\mathbf{1} = \begin{bmatrix} -1 \\ \vdots \\ -1 \end{bmatrix}. \end{aligned} \tag{4}$$

In a similar way, problem (3) can be formulated as

$$\begin{aligned} \min \quad & g \begin{bmatrix} \lambda \\ \mu \end{bmatrix} \\ \text{s.t.} \quad & B \begin{bmatrix} \lambda \\ \mu \end{bmatrix} \leq \begin{bmatrix} 0 \\ \vdots \\ 0 \end{bmatrix} \\ & \sum_{j=1}^b \lambda_j = 1, \sum_{j=1}^b \mu_j = 1 \\ & \lambda \in [0, 1]^b, \mu \in [0, 1]^b, \\ \text{where} \quad & g = [p_1^{D_{r_1,1}} + \dots + p_k^{D_{r_1,1}}, \dots, p_1^{D_{r_1,b}} + \dots + p_k^{D_{r_1,b}}, \\ & -p_1^{D_{r_2,1}} - \dots - p_k^{D_{r_2,1}}, \dots, -p_1^{D_{r_2,b}} - \dots - p_k^{D_{r_2,b}}]. \end{aligned} \tag{5}$$

In both problems (4) and (5), the left hand side of the inequality constraint

$$B \begin{bmatrix} \lambda \\ \mu \end{bmatrix} = \sum_{j=1}^b \lambda_j p^{D_{r_1,j}} - \sum_{j=1}^b \mu_j p^{D_{r_2,j}}$$

gives a vector that is in the set  $K^1 - K^2$ , because the components of both vectors  $\lambda, \mu \in [0, 1]^b$  sum up to one. Furthermore, by definition, each element

in the set  $K^1 - K^2$  can be expressed in this way. Consequently, each row in the inequality constraint of problem (4) is of the form

$$\sum_{j=1}^b \lambda_j p_i^{D_{r_1,j}} - \sum_{j=1}^b \mu_j p_i^{D_{r_2,j}} \leq t$$

for some  $i = 1, \dots, k$ . Thus, minimizing  $t$  as the objective function is equivalent to minimizing the maximum of  $\sum_{j=1}^b \lambda_j p_i^{D_{r_1,j}} - \sum_{j=1}^b \mu_j p_i^{D_{r_2,j}}$  over all  $i$ . This implies that problem (4) is equivalent to problem (2).

In problem (5), each row in the inequality constraint is of the form

$$\sum_{j=1}^b \lambda_j p_i^{D_{r_1,j}} \leq \sum_{j=1}^b \mu_j p_i^{D_{r_2,j}}$$

for some  $i = 1, \dots, k$ . The objective function becomes

$$g \begin{bmatrix} \lambda \\ \mu \end{bmatrix} = \sum_{l=1}^k \left( \sum_{j=1}^b \lambda_j p_l^{D_{r_1,j}} - \sum_{j=1}^b \mu_j p_l^{D_{r_2,j}} \right).$$

The inequality constraints and the objective function are thus the same as in problem (3).

Problems (4) and (5) can be solved with any linear programming solvers e.g., ILOG CPLEX (see <http://www.cplex.com/>) or GLPK (see <http://www.gnu.org/software/glpk/>) or with the MATLAB linprog function (see <http://www.mathworks.com/access/helpdesk/help/toolbox/optim/ug/linprog.html>).

#### 4 Decision making with the approximation – a surrogate problem

In this section, we discuss how one can use interactive methods with the approximation created in the previous section. The main tool is the surrogate multiobjective optimization problem implied by the inherently nondominated Pareto front approximation. The surrogate problem was introduced in [12, 13]. In order to accomplish this, there are two issues that need to be addressed:

1. The scalarizations of the surrogate problem given by the interactive method at use need to be efficiently solvable in order to present the new computed solutions to the decision maker in real time.
2. Once a preferred approximate outcome on the Pareto front approximation has been found with the help of the surrogate problem and an interactive method, one must find the closest Pareto optimal outcome on the Pareto front of the original problem.

Our solution to the first issue is a multiobjective mixed integer linear formulation for the surrogate problem. Powerful mixed integer linear solvers, e.g., CPLEX and GLPK, allow then to efficiently solve scalarizations of the surrogate problem provided that the scalarization of the interactive method maintains linearity of the problem. This enables us to efficiently use many interactive methods e.g., the synchronous NIMBUS [23, 25, 27], because all its scalarizations can be formulated in a way that they maintain linearity.

In [12], the surrogate problem given by an inherently nondominated Pareto front approximation  $\mathcal{A}$  was formulated as

$$\begin{aligned} \min \quad & (z_1, \dots, z_k) \\ \text{s.t.} \quad & z \in \cup_{K \in \mathcal{A}} K. \end{aligned} \quad (6)$$

The idea of the surrogate problem is that its outcomes are in the same space as the outcomes in  $P$  and, thus, the decision maker should be able to articulate his/her preferences about them when using an interactive method to solve problem (6), because the objectives have corresponding meanings in (6) and in the original problem..

Problem (6) cannot be inputted into standard multiobjective optimization solvers because it contains a non-algebraic constraint  $z \in \cup_{K \in \mathcal{A}} K$ . Problem (6), however, has an equivalent mixed integer linear formulation

$$\begin{aligned} \min \quad & (z_1, \dots, z_k) \\ \text{s.t.} \quad & \sum_{j=1}^a \sum_{l=1}^b \lambda_{j,l} = 1 \\ & \sum_{l=1}^b \lambda_{j,l} \leq y_j, \text{ for all } j = 1, \dots, a \\ & \sum_{j=1}^a y_j = 1 \\ & \lambda \in [0, 1]^{a \times b} \\ & y \in \{0, 1\}^a, \\ \text{where} \quad & z_i = \sum_{j=1}^a \sum_{l=1}^b \lambda_{j,l} P_i^{A_{l,j}} \text{ for all } i = 1, \dots, k. \end{aligned} \quad (7)$$

In multiobjective optimization problem (7), there are two variables  $\lambda \in [0, 1]^{a \times b}$  and  $y \in \{0, 1\}^a$ . The component  $\lambda_{j,l}$  of variable  $\lambda$  is for all  $j = 1, \dots, a$  and  $l = 1, \dots, b$  the coefficient of the vertex  $l$  of the polytope given by row  $j$  in the matrix  $A$ . The variable  $y$  determines which of the rows of matrix  $\lambda$  are nonzero. Note, that the third equality states that all but one component of the variable  $y$  must be zero. Matrix  $A$  refers to the representation of the inherently nondominated Pareto front approximation given by the PAINT method. According to the first equality constraint, all components  $\lambda_{j,l}$  must sum up to one. According to the second inequality constraint, the entries in all but one row in the matrix representing the variable  $\lambda$  must sum up to zero, because only one component of  $y$  is nonzero. Thus, problem (7) is equivalent to problem (6).

As an example of scalarizations that maintain linearity we consider the achievement scalarizing problem [33], which can be formulated for multiobjective optimization problem (1) as

$$\min_{x \in S} \max_{i=1, \dots, k} w_i (f_i(x) - \bar{z}_i) \quad (8)$$

with the normalizing weight  $w$  in the positive orthant of  $\mathbb{R}^k$  and the reference point  $\bar{z} \in \mathbb{R}^k$  representing desirable objective values. For example, with the CPLEX solver and using an Acer laptop with Core 2 Duo P8700 processor and 4 GB of RAM, we solved the achievement scalarizing problem formulated for a surrogate problem implied by an inherently nondominated Pareto front approximation with the number of objectives  $k = 5$ , number of initial points  $m = 330$ , number of polytopes  $a = 1824$  and the maximal number of outcomes defining a polytope  $b = 5$  in approximately 3 seconds. We consider this Pareto front approximation to be as large as we need for decision making. We also consider the computational time of 3 seconds to be small enough that the formulations and tools of this section can be used with e.g., the NIMBUS method to compute new approximate Pareto optimal outcomes for the decision maker. These results show promise for further applicability of the PAINT method.

The surrogate problem, while useful in solving computationally expensive problems, also has some limitations. First, the problem does not provide information about the preimage of the Pareto front in the decision space. Second, the outcomes of the surrogate problem include all the inherently nondominated interpolations between the Pareto optimal outcomes and, thus, the method does not detect any possible areas where the Pareto front is disconnected. The first limitation is an issue if the decision variables are very meaningful to the decision maker in making decisions. This may sometimes be the case in engineering problems, even though the opposite is argued in [17]. The second limitation is an issue if the Pareto front has big areas of discontinuity and the decision maker is especially interested in those areas in the outcome space. These are topics of further research on this subject.

Finally, when a preferred approximate Pareto optimal outcome has been found, we must find the decision vector in the decision space  $S$  that gives the outcome on the actual Pareto front of the original problem that is the best match to it. As described above, we propose to solve the achievement scalarizing problem formulated for the original problem with the preferred approximate Pareto optimal outcome as the reference point. If the resulting outcome on the actual Pareto front is satisfactory to the decision maker, we stop. If we find out that the approximate Pareto optimal outcome was satisfactory to the decision maker but the actual Pareto optimal outcome was not, then we must update the inherently nondominated Pareto front approximation. This is done by adding the new Pareto optimal outcome to the set  $P$  and running the PAINT method again. This yields a Pareto front approximation that is more accurate close to the new Pareto optimal outcome. Note, however, that both of these operations (finding the closest element on the actual Pareto front and rerunning the PAINT method) take time. Hence, repeating them multiple times should be avoided, if possible.

## 5 Examples

In this section, we present examples of inherently nondominated Pareto front approximations constructed with the PAINT method described in Algorithm 1. The implementation of Algorithm 1 was written under GNU Octave [5]. Delaunay triangulations of the finite sets of Pareto optimal outcomes were constructed with a Qhull implementation (see <http://www.qhull.org/>) of the Quickhull algorithm [1]. Optimization problems (4) and (5) were solved with GLPK (GNU Linear Programming Kit, see <http://www.gnu.org/software/glpk/>). This all was done on an Acer laptop with Core 2 Duo P8700 processor and 4 GB of RAM running Fedora 12.

The different example problems with multiple objectives have been selected to demonstrate different types of Pareto fronts and to show that the PAINT method is suitable for approximating all of them. The problems in Subsections 5.1, 5.2 and 5.3 are test problems that are used for testing evolutionary multiobjective optimization algorithms. These problems are not computationally expensive but have been selected because their Pareto front has an interesting geometry to depict in three dimensions. Let us point out that even though the PAINT method has been developed for enabling faster solution processes for computationally expensive multiobjective optimization problems, it can naturally be applied to computationally inexpensive problems. This is because the computational expense of the problem does not play any role within the PAINT method after the initial set of Pareto optimal outcomes has been computed. The problem in Subsection 5.4 is a real application of multiobjective optimization to wastewater treatment planning. With it, we demonstrate how the PAINT method can be used with real-life applications.

We emphasize that the purpose of the examples is not to illustrate the decision making aspect (including the surrogate problem), which has to be studied with real-life problems and with real decision makers. As mentioned, here we want to demonstrate how PAINT works as an approximation method. Naturally, the benefits of the surrogate problem will better show up with problems with more than three objectives, for which one cannot graphically present the Pareto front nor its approximation.

### 5.1 The three-objective Viennet’s test problem

The Pareto front of the three-objective Viennet’s test problem [32] consists of a one-dimensional curve in  $\mathbb{R}^3$ . Figure 1a illustrates 240 Pareto optimal outcomes for this problem generated with a local search assisted evolutionary multiobjective optimization algorithm [31]. This figure is drawn to give an understanding of what the Pareto front looks like. If we dealt with a computationally expensive multiobjective optimization problem, the generation of this large set would be too time-consuming.

Table 1 lists a subset of 20 Pareto optimal outcomes  $\{p^1, p^2, \dots, p^{20}\}$  which are randomly selected from the Pareto optimal outcomes in Figure 1a. This

Table 1: The initial set of Pareto optimal outcomes  $P$  for the Viennet's test problem

Outcome	$z_1$	$z_2$	$z_3$
$p^1$	0.030032	0.880533	0.991796
$p^2$	0.000000	1.000000	-0.628862
$p^3$	1.000000	0.894206	0.058335
$p^4$	0.929845	0.893301	0.062197
$p^5$	0.029038	0.880566	1.000000
$p^6$	0.001497	0.955310	-0.588038
$p^7$	0.111630	0.881325	0.471040
$p^8$	0.451435	0.888326	0.124438
$p^9$	0.000529	0.972149	-0.616055
$p^{10}$	0.602037	0.887156	0.097756
$p^{11}$	0.013991	0.895472	0.014610
$p^{12}$	0.128648	0.881001	0.450349
$p^{13}$	0.000014	0.995955	-0.628542
$p^{14}$	0.002388	0.945419	-0.557775
$p^{15}$	0.262487	0.882276	0.212985
$p^{16}$	0.845441	0.889814	0.070032
$p^{17}$	0.386630	0.883861	0.143005
$p^{18}$	0.004519	0.928747	-0.471886
$p^{19}$	0.000457	0.974112	-0.617907
$p^{20}$	0.005243	0.924974	-0.439333

subset is taken as the initial set of Pareto optimal outcomes  $P$  for the problem. These Pareto optimal outcomes are also illustrated in Figure 1b. Notice that the vectors are distributed rather non-uniformly. If we dealt with a computationally expensive multiobjective optimization problem, generating this set would obviously be a lot less time-consuming than generating the outcomes in Figure 1a.

Figure 1c shows the body of a Delaunay triangulation of the set  $P$ , which contains 395 polytopes. As it can be seen, the body of the Delaunay triangulation covers the whole convex hull of  $P$ . By comparing Figures 1a and 1c one can see that the Delaunay triangulation as such does not give a good approximation of the Pareto optimal outcomes because its body contains many vectors that are very far from the Pareto optimal outcomes.

After the Delaunay triangulation has been constructed the inappropriate polytopes are removed from it. In this case, the inherently nondominated Pareto front approximation is represented (as described in Section 3) by the

matrix

$$\begin{array}{cccccccccccccccc}
 \left[ \begin{array}{cccccccccccccccc}
 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 & 12 & 13 & 14 & 15 & 16 \\
 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 & 12 & 13 & 14 & 15 & 16 \\
 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 & 12 & 13 & 14 & 15 & 16 \\
 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 & 12 & 13 & 14 & 15 & 16
 \end{array} \right. \\
 17 & 18 & 19 & 20 & 1 & 1 & 1 & 1 & 2 & 3 & 3 & 4 & 5 & 6 & 6 & 7 \\
 17 & 18 & 19 & 20 & 5 & 7 & 11 & 12 & 13 & 4 & 16 & 16 & 11 & 9 & 14 & 12 \\
 17 & 18 & 19 & 20 & 1 & 1 & 1 & 1 & 2 & 3 & 3 & 4 & 5 & 6 & 6 & 7 \\
 17 & 18 & 19 & 20 & 1 & 1 & 1 & 1 & 2 & 3 & 3 & 4 & 5 & 6 & 6 & 7 \\
 8 & 8 & 9 & 10 & 10 & 11 & 12 & 13 & 14 & 15 & 18 & 1 & 1 & 3 & 8 \\
 10 & 17 & 19 & 16 & 17 & 20 & 15 & 19 & 18 & 17 & 20 & 5 & 7 & 4 & 10 \\
 8 & 8 & 9 & 10 & 10 & 11 & 12 & 13 & 14 & 15 & 18 & 11 & 12 & 16 & 17 \\
 8 & 8 & 9 & 10 & 10 & 11 & 12 & 13 & 14 & 15 & 18 & 1 & 1 & 3 & 8
 \end{array} \right]^T$$

Notice that the matrix has been transposed and cut into three parts to save space. The body of this approximation is shown in Figure 1d. Comparing this figure with Figure 1c one may notice that this approximation is a subset of the Delaunay triangulation. By comparing Figures 1a and 1d, one can see that the approximation is able to describe also many of the Pareto optimal outcomes in Figure 1a but not in the set  $P$ .

All in all, 88.1% of the polytopes in the Delaunay triangulation were removed in the PAINT method. All of these were removed due to rule (R1), because 58.7% of the polytopes were removed as they were not inherently nondominated and, after this, 29.4% of the polytopes were removed as they dominated or were dominated by an outcome in  $P$ . The PAINT method took 2.3 seconds on the configuration described above. The  $\mathcal{D}$ -maximal Pareto front approximation contained 47 polytopes.

## 5.2 The three-objective DTLZ2 test problem

The Pareto front of the three-objective DTLZ2 test problem [4] consists of the subset of the unit sphere that is in the positive orthant of  $\mathbb{R}^k$ . A set of 200 Pareto optimal outcomes can be seen in Figure 2a. Again, this set would be time-consuming to produce for a computationally expensive multiobjective optimization problem. As in the case of Viennet's test problem, the Figure 2b illustrates a random subset of this consisting of 20 outcomes.

Figure 2c depicts an inherently nondominated Pareto front approximation which is the output of the PAINT method. Since the Pareto front of this problem is a two-dimensional part of the surface of the ball, the body of the inherently nondominated Pareto front approximation is two-dimensional even though no information about the dimensionality was given to the algorithm. As it should be, the approximation can again describe also the Pareto optimal outcomes that are not in the initial set of Pareto optimal outcomes  $P$ .



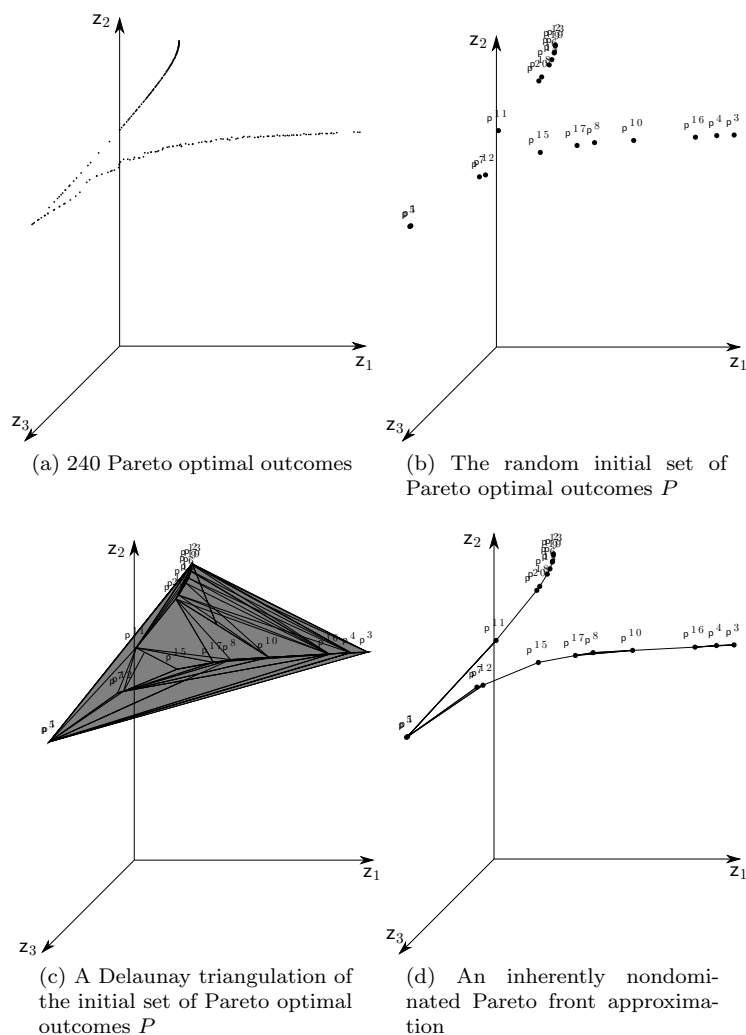


Fig. 1: Approximating the Pareto front of the three-objective Viennet's test problem

In this case, 56.8% of the polytopes in the Delaunay triangulation were removed due to (R1) as 33.6% of the polytopes were not inherently nondominated and 23.2% were inherently nondominated, but dominated or were dominated by an outcome in  $P$ . Of the polytopes that were not removed due to rule (R1), 31.8% were removed due to rule (R2). In total, the PAINT method took 8.3 seconds. The final  $\mathcal{D}$ -maximal inherently nondominated Pareto front approximation given by PAINT contained 85 polytopes.

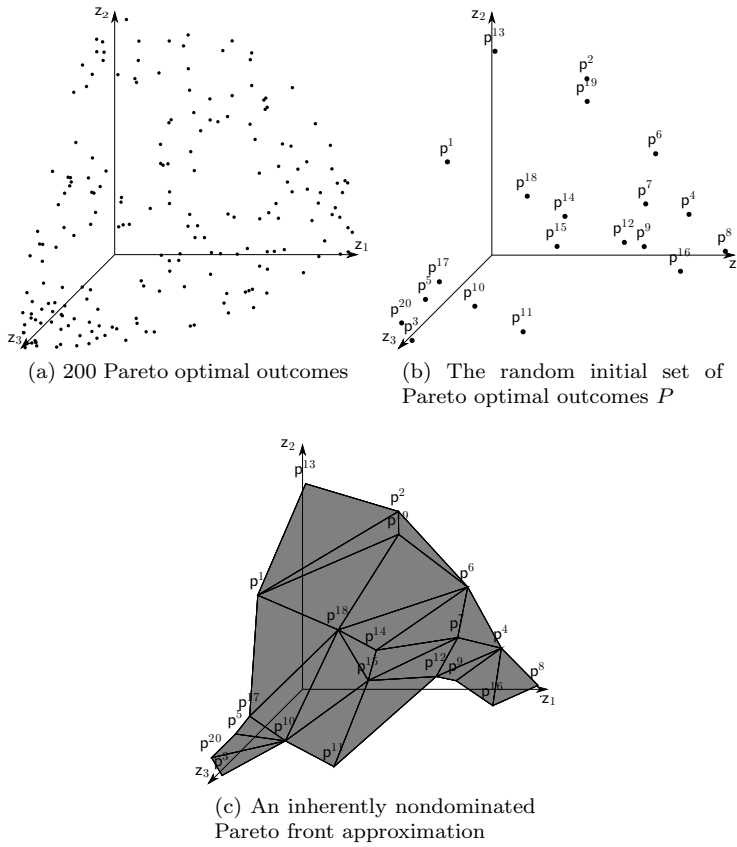


Fig. 2: Approximating the Pareto front of the three-objective DTLZ2 test problem

### 5.3 The four-objective DTLZ2 test problem

The four-objective DTLZ2 test problem is similar to the three-objective one but with one more objective. That is, the Pareto front of this problem consists of a subset of the surface of the unit ball that is in the positive orthant of  $\mathbb{R}^4$ .

Table 2 lists the initial Pareto optimal outcomes in  $P$ . These outcomes are randomly generated vectors on the Pareto front of this problem.

Of the polytopes in the Delaunay triangulation 38.8% were removed due to rule (R1) as 19.8% were not inherently nondominated and 19.0% were inherently nondominated, but dominated or were dominated by an outcome in  $P$ . Furthermore, 34.2% of the polytopes that were not removed due to rule (R1) were removed due to rule (R2). Here the PAINT method took 111.2 seconds. The  $\mathcal{D}$ -maximal inherently nondominated Pareto front approximation given by PAINT contained 148 polytopes. The body of the inherently

Table 2: The given set of Pareto optimal outcomes  $P$  for the four-objective DTLZ2 test problem

Outcome	$z_1$	$z_2$	$z_3$	$z_4$
$p^1$	0.1712	0.7377	0.6467	0.0909
$p^2$	0.4399	0.2015	0.4218	0.7668
$p^3$	0.2457	0.1934	0.6819	0.6613
$p^4$	0.7375	0.3556	0.2103	0.5343
$p^5$	0.2430	0.6618	0.6582	0.2641
$p^6$	0.4781	0.6984	0.0493	0.5304
$p^7$	0.0952	0.4074	0.5114	0.7506
$p^8$	0.5689	0.3861	0.7032	0.1810
$p^9$	0.1823	0.0621	0.7022	0.6854
$p^{10}$	0.3304	0.5462	0.5204	0.5672
$p^{11}$	0.3074	0.7199	0.2407	0.5738
$p^{12}$	0.5578	0.3268	0.3376	0.6842
$p^{13}$	0.3685	0.5472	0.4703	0.5861
$p^{14}$	0.3598	0.4594	0.6085	0.5378
$p^{15}$	0.5401	0.1739	0.6530	0.5016
$p^{16}$	0.0395	0.2779	0.7468	0.6030
$p^{17}$	0.7778	0.4705	0.4110	0.0682
$p^{18}$	0.5801	0.6938	0.1218	0.4089
$p^{19}$	0.6200	0.2287	0.4903	0.5682
$p^{20}$	0.5681	0.3916	0.6646	0.2868

nondominated Pareto front approximation can be seen in Figure 3. In the figure the approximation has been projected to  $\mathbb{R}^3$  by the projection function  $p : \mathbb{R}^4 \rightarrow \mathbb{R}^3, (x_1, x_2, x_3, x_4) \mapsto (x_1, x_2, x_3)$  and the fourth objective is marked with colour.

#### 5.4 Wastewater treatment planning

Finally, we have a real life multiobjective optimization problem of wastewater treatment planning from [11]. The problem considers the so-called activated sludge process, which is globally the most common method of wastewater treatment. In this process, biomass (which is called activated sludge) suspended in the wastewater to be treated is cultivated and maintained in an aerated bioreactor. The wastewater is purified, i.e., organic carbon, nitrogen and phosphorus are removed during its retention in the bioreactor. The bioreactor is followed by a clarifier basin, in which the biomass is separated by gravitational settling and returned to the bioreactor, and the treated wastewater is directed as overflow to further treatment or to discharge. Excess activated sludge is removed from the process and treated separately. The process performs nitrification, i.e., oxidation of ammonium nitrogen to nitrate nitrogen by autotrophic, slow-growing micro-organisms. The biochemical reactions involved consume a lot of oxygen and alkalinity. Oxygen is supplied by aeration compressors and alkalinity partly by influent wastewater, partly by adding chemicals, e.g.,  $\text{Na}_2\text{CO}_3$ .

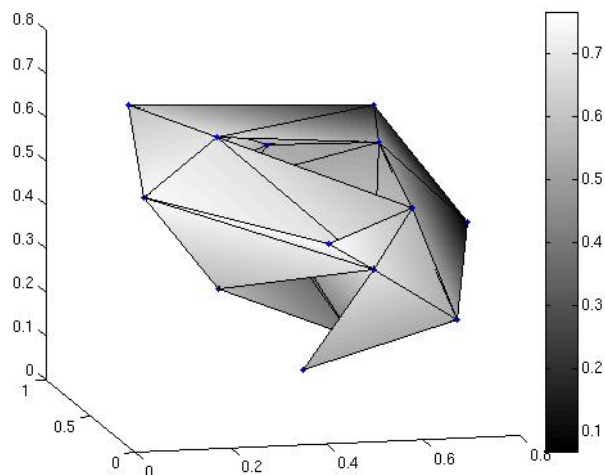


Fig. 3: A projection of an inherently nondominated Pareto front approximation for the 4-objective DTLZ2 test problem. The value of the fourth objective is marked with colour.

Aeration consumes energy and chemicals cost money, so minimizing the need for aeration and alkalinity addition is important for the operational economy of the plant.

The considered multiobjective optimization problem is computationally expensive and no closed form equations are known for the objectives. In [11], the process was simulated and a global solver was used to find Pareto optimal solutions with the interactive method NIMBUS. Because the problem is computationally expensive, it took a long time to find new Pareto optimal solutions. Hence, this problem has room for improvement with our methods.

We considered the 11 solutions generated with NIMBUS in [11]. One of the 11 solutions was dominated by another solution in the set and, thus, it was dropped. We assumed that the ten remaining solutions were Pareto optimal, which was in fact uncertain because a global solver was used to optimize the scalarizations of the multiobjective optimization problem that were given by the interactive method NIMBUS. The outcomes implied by these solutions (shown in Table 3) were given to the PAINT method as input  $P$ . The PAINT method took 2.5 seconds and the final inherently nondominated Pareto front approximation contained 37 polytopes. The body of the Pareto front approximation is shown in Figure 4.

In computational tests, the achievement scalarizing problem formulated for problem (7) takes 0.01 seconds to be solved with CPLEX for various values of reference points and weights. After a preferred solution to the surrogate problem has been found, one can project the outcome on the Pareto front

Table 3: The given set of Pareto optimal outcomes  $P$  for the multiobjective wastewater problem

Outcome	residual ammonium nitrogen concentration [ $gN/m^3$ ]	alkalinity chemical dosing rate [ $m^3/d$ ]	aeration energy consumption [ $kW$ ]
$p^1$	8.05	218	460
$p^2$	3.52	286	490
$p^3$	1.69	326	506
$p^4$	4.9	298	477
$p^5$	1.11	336	515
$p^6$	0.55	347	528
$p^7$	9.36	246	448
$p^8$	30.2	7.23	308
$p^9$	0.9	333	519
$p^{10}$	0.72	332	524

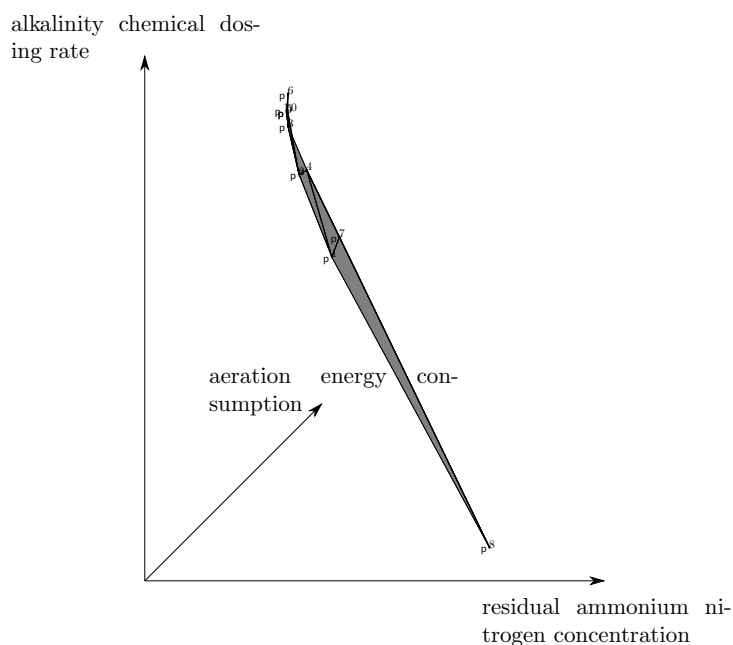


Fig. 4: An inherently nondominated Pareto front approximation for the multiobjective wastewater treatment planning problem

by solving globally the achievement scalarizing problem of the reference point method with the preferred solution to the surrogate problem as the preferred solution.

In this case, the initial set of Pareto optimal outcomes was not the best possible for our approximation because, as can be seen from Figure 4, the

outcome  $p^8$  is very far from the other outcomes. This may lead to inaccuracy of the approximation in interpolating between outcome  $p^8$  and the others.

If one had used the PAINT method in [11], the decision making process would have been much faster with the interactive method. Because with the help of the surrogate problem one could have provided the decision maker quickly with new approximate Pareto optimal outcomes that correspond to his/her preferences, the decision maker should have been more inclined to further explore the Pareto front. This could have led the decision maker to get more insight about the problem.

## 6 Conclusions

We have demonstrated how one can in practice generate a Pareto front approximation by interpolating between a small finite set of given Pareto optimal outcomes. The interpolation is produced by the PAINT method proposed in this paper. The theory behind this method is based on a property called inherent nondominance previously introduced by the authors. In this paper, we have described a way of implementing the method and presented computational results on interpolating on Pareto fronts with the implementation. We have also illustrated how the Pareto front approximation implies a mixed integer linear surrogate problem for the original one that can be used in decision making concerning the original problem. This approach is especially useful with computationally expensive multiobjective optimization problems.

The combination of the following two benefits of our approach make it novel: (1) the proposed Pareto front approximation method can approximate non-convex Pareto fronts, and (2) any interactive method can be used in decision making with the proposed Pareto front approximation because of the mixed integer linear surrogate problem that it implies. These benefits make our approach applicable to different computationally expensive multiobjective optimization problems and enable it to support various decision makers.

As an example of interactive methods that can be applied with the Pareto front approximation produced, we have considered achievement scalarizing function based approaches. The achievement scalarizing problem formulated for the presented surrogate problem can be solved e.g., with the CPLEX linear solver. This enables us to use an interactive method based on decision maker's preferences formulated as a reference point, like the NIMBUS method. While the surrogate problem is a multiobjective optimization problem that can in theory be used with any interactive method, the implementations of the methods may restrict its use because the surrogate problem contains integer variables.

Further research on this subject includes applying the PAINT method with real-life problems and real decision makers. We can use the interface of IND-NIMBUS [24] (see <http://ind-nimbus.it.jyu.fi/>) and solve the scalarized problems of the mixed integer surrogate problem with the CPLEX solver. In

this way, we can study the performance of the PAINT method in decision making.

## Acknowledgements

We thank Mr. Karthik Sindhya for providing the Pareto optimal outcomes for the Viennet's test problem. We also thank Dr. Jussi Hakanen for his help with the wastewater treatment planning problem.

## References

1. C. B. Barber, D. P. Dobkin, and H. Huhdanpää. The Quickhull Algorithm for Convex Hulls. *ACM Transactions on Mathematical Software*, 22:469–483, 1996.
2. V. E. Bezerkin, G. K. Kamenev, and A. V. Lotov. Hybrid Adaptive Methods for Approximating a Nonconvex Multidimensional Pareto Frontier. *Computational Mathematics and Mathematical Physics*, 46:1918–1931, 2006.
3. K. Q. Brown. Voronoi Diagrams from Convex Hulls. *Information Processing Letters*, 9:223–228, 1979.
4. K. Deb, L. Thiele, M. Laumanns, and E. Zitzler. Scalable Multi-Objective Optimization Test Problems. *IEEE International Conference on E-Commerce Technology*, 1:825–830, 2002.
5. J. W. Eaton. *GNU Octave Manual*. Network Theory Limited, 2002.
6. H. Edelsbrunner and E. P. Mücke. Simulation of Simplicity: a Technique to Cope with Degenerate Cases in Geometric Algorithms. *ACM Transactions on Graphics*, 9:66–104, 1990.
7. R. V. Efremov and G. K. Kamenev. Properties of a Method for Polyhedral Approximation of the Feasible Criterion Set in Convex Multiobjective Problems. *Annals of Operations Research*, 166:271–279, 2009.
8. P. Eskelinen, K. Miettinen, K. Klamroth, and J. Hakanen. Pareto Navigator for Interactive Nonlinear Multiobjective Optimization. *OR Spectrum*, 32:211–227, 2010.
9. T. Goel, R. Vaidyanathan, R. T. Haftka, W. Shyy, N. V. Queipo, and K. Tucker. Response Surface Approximation of Pareto Optimal Front in Multi-Objective Optimization. *Computer Methods in Applied Mechanics and Engineering*, 196:879–893, 2007.
10. B. Grünbaum. *Convex Polytopes*. Interscience Publishers, London, 1967.
11. J. Hakanen, K. Miettinen, and K. Sahlstedt. Wastewater Treatment: New Insight Provided by Interactive Multiobjective Optimization. *Decision Support Systems*, 51:328–337, 2011.
12. M. Hartikainen, K. Miettinen, and M. M. Wiecek. Constructing a Pareto Front Approximation for Decision Making. *Mathematical Methods of Operations Research*, 73:209–234, 2011.

13. M. Hartikainen, K. Miettinen, and M. M. Wiecek. Pareto Front Approximations for Decision Making with Inherent Non-dominance. In Y. Shi, S. Wang, G. Kou, and J. Wallenius, editors, *New State of MCDM in the 21st Century, Selected Papers of the 20th International Conference on Multiple Criteria Decision Making 2009*, pages 35–46. Springer-Verlag Berlin, Heidelberg, 2011.
14. M. Hasenjäger and B. Sendhoff. Crawling Along the Pareto Front: Tales From the Practice. In *The 2005 IEEE Congress on Evolutionary Computation (IEEE CEC 2005)*, pages 174–181, Piscataway, NJ, 2005. IEEE Press.
15. C. Hwang and A. S. M. Masud. *Multiple Objective Decision Making - Methods and Applications: a State-of-the-Art Survey*. Springer-Verlag Berlin, Heidelberg, 1979.
16. G. K. Kamenev. Study of an Adaptive Single-Phase Method for Approximating the Multidimensional Pareto Frontier in Nonlinear Systems. *Computational Mathematics and Mathematical Physics*, 49:2103–2113, 2009.
17. R. L. Keeney. *Value-Focused Thinking: a Path to Creative Decisionmaking*. Harvard University Press, Harvard, 1996.
18. T. Laukkanen, T.-M. Tveit, V. Ojalehto, K. Miettinen, and C.-J. Fogelholm. An Interactive Multi-Objective Approach to Heat Exchanger Network Synthesis. *Computers & Chemical Engineering*, 34:943–952, 2010.
19. A. V. Lotov, V. A. Bushenkov, and G. A. Kamenev. *Interactive Decision Maps*. Kluwer Academic Publishers, Boston, 2004.
20. M. Luque, F. Ruiz, and K. Miettinen. Global Formulation for Interactive Multiobjective Optimization. *OR Spectrum*, 33:27–48, 2011.
21. J. Martin, C. Bielza, and D. R. Insua. Approximating Nondominated Sets in Continuous Multiobjective Optimization Problems. *Naval Research Logistics*, 52:469–480, 2005.
22. P. McMullen. The Maximum Number of Faces of a Convex Polytope. *Mathematika*, 17:179–184, 1970.
23. K. Miettinen. *Nonlinear Multiobjective Optimization*. Kluwer Academic Publishers, Boston, 1999.
24. K. Miettinen. IND-NIMBUS for Demanding Interactive Multiobjective Optimization. In T. Trzaskalik, editor, *Multiple Criteria Decision Making'05*, pages 137–150. The Karol Adamecki University of Economics in Katowice, Katowice, 2006.
25. K. Miettinen and M. Mäkelä. Interactive Bundle-based Method for Nondifferentiable Multiobjective Optimization: NIMBUS. *Optimization*, 34:231–246, 1995.
26. K. Miettinen and M. M. Mäkelä. On Scalarizing Functions in Multiobjective Optimization. *OR Spectrum*, 24:193–213, 2002.
27. K. Miettinen and M. M. Mäkelä. Synchronous Approach in Interactive Multiobjective Optimization. *European Journal of Operational Research*, 170:909–922, 2006.
28. K. Miettinen, F. Ruiz, and A. P. Wierzbicki. Introduction to Multiobjective Optimization: Interactive Approaches. In J. Branke, K. Deb, K. Miettinen, and R. Slowinski, editors, *Evolutionary Multi-Criterion Optimization*, pages 1–20. Springer, Berlin, Heidelberg, 2006.



- tinien, and R. Slowinski, editors, *Multiobjective Optimization: Interactive and Evolutionary Approaches*, pages 27–57. Springer-Verlag Berlin, Heidelberg, 2008.
29. M. Monz. *Pareto Navigation - Algorithmic Foundation of Interactive Multi-criteria IMRT Planning*. PhD thesis, University of Kaiserslautern, 2006.
  30. S. Ruzika and M. M. Wiecek. Approximation Methods in Multiobjective Programming. *Journal of Optimization Theory and Applications*, 126:473–501, 2005.
  31. K. Sindhya, K. Deb, and K. Miettinen. Improving Convergence of Evolutionary Multi-Objective Optimization with Local Search: a Concurrent-Hybrid Algorithm. *Natural Computing*, To Appear. DOI:10.1007/s11047-011-9250-4.
  32. R. Viennet, C. Fonteix, and I. Marc. Multicriteria Optimization Using a Genetic Algorithm for Determining a Pareto Set. *International Journal of Systems Science*, 27:255–260, 1996.
  33. A. P. Wierzbicki. On the Completeness and Constructiveness of Parametric Characterizations to Vector Optimization Problems. *OR Spectrum*, 8:73–87, 1986.

**PV**

**DEMONSTRATING THE APPLICABILITY OF PAINT TO  
COMPUTATIONALLY EXPENSIVE REAL-LIFE  
MULTIOBJECTIVE OPTIMIZATION**

by

Markus Hartikainen · Vesa Ojalehto 2011

arXiv:1109.3411v1 [cs.NA]

# Demonstrating the Applicability of PAIN'T to Computationally Expensive Real-life Multiobjective Optimization

Markus Hartikainen and Vesa Ojalehto

September 16, 2011

## Abstract

We demonstrate the applicability of a new PAIN'T method to speed up iterations of interactive methods in multiobjective optimization. As our test case, we solve a computationally expensive non-linear, five-objective problem of designing and operating a wastewater treatment plant. The PAIN'T method interpolates between a given set of Pareto optimal outcomes and constructs a computationally inexpensive mixed integer linear surrogate problem for the original problem. We develop an IND-NIMBUS<sup>®</sup> PAIN'T module to combine the interactive NIMBUS method and the PAIN'T method and to find a preferred solution to the original problem. With the PAIN'T method, the solution process with the NIMBUS method take a comparatively short time even though the original problem is computationally expensive.

## 1 Introduction

In this section, we give background for our study and a brief overview of this paper. First, in Section 1.1, we describe the aim of this study and the structure of this paper. In Section 1.2, we introduce the basic concepts of multiobjective optimization that are used in this paper. In Section 1.3, we consider the main issues related to computationally expensive multiobjective optimization. Finally, in Section 1.4, we describe our test case, i.e., the multiobjective problem of designing and operating a wastewater treatment plant.

### 1.1 About this Paper

In this paper, we demonstrate how the interpolation method PAIN'T (introduced in [12–14]) can be used to speed up the iterations of an interactive method when solving computationally expensive multiobjective optimization problems. For this, we revisit a computationally expensive five-objective optimization problem from [37] that models designing and operating a wastewater treatment plant.

In our study of the wastewater problem, we used the interactive NIMBUS method, as was used in [37]. Compared to the previous study of [37], the iterations of the interactive method NIMBUS were much faster because of a surrogate problem constructed with the PAINT method. In our study, the decision maker was Mr. Kristian Sahlstedt as in [37] and, thus, we were able to ask the decision maker to compare his experiences of using the NIMBUS method to solve the problem with and without the PAINT method. This comparison of the two approaches gave a unique perspective to our study.

The structure of this paper is as follows: After describing the background of our study in this section, we outline the PAINT approach to solving computationally expensive problems and the PAINT method in Section 2. We describe the new IND-NIMBUS<sup>®</sup> PAINT module (also called the PAINT module in this paper for short) in Section 3. In Section 4, we illustrate how we used the PAINT method to construct a Pareto front approximation and a mixed integer linear surrogate problem for the wastewater treatment problem. In Section 4, we also describe our decision maker's involvement in solving the problem with the PAINT module. In Section 5, we further analyze the decision making process of Section 4. Finally, in Section 6, we give our conclusions and ideas for further research.

## 1.2 Multiobjective Optimization

Multiobjective optimization concerns simultaneously optimizing multiple conflicting objectives. A general formulation for a multiobjective optimization problem with  $k$  objectives is

$$\begin{aligned} \min \quad & (f_1(x), \dots, f_k(x)) \\ \text{s.t.} \quad & x \in S, \end{aligned} \tag{1}$$

where  $f_i$  are the objective functions and  $S$  is the feasible set. A vector  $x \in S$  is called a (feasible) solution. For these problems, instead of a single optimal solution there typically exist many Pareto optimal solutions. A solution  $x \in S$  is said to (Pareto) dominate another solution  $y \in S$  if  $f_i(x) \leq f_i(y)$  for all  $i = 1, \dots, k$  and  $f_j(x) < f_j(y)$  for at least one  $j \in \{1, \dots, k\}$ . A solution  $x^* \in S$  is Pareto optimal, if there does not exist a solution  $x \in S$  that dominates it. A vector  $z = (f_1(x), \dots, f_k(x))$  with  $x \in S$  is called an outcome, and an outcome is called Pareto optimal if it is given by a Pareto optimal solution. The set of Pareto optimal outcomes is called the Pareto front.

Although many Pareto optimal solutions typically exist, only one has to be chosen for implementation. Distinguishing between Pareto optimal solutions requires preference information about the objectives of the problem. In multiobjective optimization, it is often assumed that there exists a decision maker who is an expert in the application area and who is prepared to answer questions concerning those preferences. In this paper, this whole process of choosing a single solution for implementation is called solving the problem and, when we want to emphasize the decision maker's involvement, it is also referred to as the decision making process.

The type of information that is asked from the decision maker depends on the multiobjective optimization method that is used to solve the problem. Dif-

ferent types of multiobjective optimization methods (as categorized in [22, 38]) are no-preference methods, a priori methods, a posteriori methods and interactive methods. In no-preference methods the decision maker is not asked any questions. No-preference methods are applicable to problems, where the decision maker is not available or does not want to get involved. In a priori methods, the decision maker is first asked for preference information and then the best solution according to those preferences is found. The difficulty with a priori methods is that the decision maker may find it hard to define preferences without ever seeing feasible or Pareto optimal solutions. In a posteriori methods, a representative set of the Pareto optimal solutions is found from which the decision maker is allowed to choose a preferred one. The difficulty with a posteriori methods is that generating a representative solution set may be time-consuming and choosing a preferred solution from a large set of solutions may be hard (see e.g., [19]).

In this paper, we follow the ideology of interactive methods in solving multiobjective optimization problems. In interactive methods, it is understood that any preference information given by the decision maker is only partial and perhaps flawed. Thus, the decision maker is allowed to explore the Pareto optimal solutions by guiding the interactive method. This allows the decision maker to learn about the problem (as argued e.g., in [27]) and find a preferred solution without examining too many solutions. For more information about interactive methods, see e.g., [22, 27]. More specifically, in this paper, we use the interactive synchronous NIMBUS method, introduced in [24–26].

### 1.3 Solving Computationally Expensive Multiobjective Optimization Problems

Some multiobjective optimization problems are computationally expensive (see e.g., [2, 10, 15, 37, 43]). This may be caused e.g., by the need to use computationally expensive simulations for evaluating the objective functions. Interactive methods have an advantage to a posteriori methods in solving computationally expensive problems, because the decision maker may guide the search in interactive methods and, thus, fewer solutions need to be computed. There is, however, a drawback. When using interactive methods, the decision maker has to wait while new solutions are computed with respect to his/her updated preferences. For computationally expensive problems, this may take a long time, which may be frustrating for the decision maker (as argued e.g., in [18]).

In order to compute new solutions faster within the interactive method, one can use approximation. Two different approximation schemes can be identified: approximating the objective functions and approximating the Pareto front. The objective functions may be approximated e.g., with meta-models like the response surface methodologies, Support Vector Machines or Radial Basis Functions (see e.g., [30]). These have been used in multiobjective optimization e.g., in [30, 42]. This is not, however, a straightforward task, because as the number of decision variables and objectives increases, the approximation itself becomes a very computationally expensive task. Another approach is approximating the Pareto front. Pareto front approximations can be found

e.g., in [1, 4, 6, 21, 29, 36, 44], where [1, 6, 21, 29] include decision making aspects connected to these. Note that in this paper, we distinguish between a Pareto front representation (a discrete set of Pareto optimal outcomes) and a Pareto front approximation (something more approximate that possibly contains vectors that are not outcomes of the problem, but merely approximate them).

In this paper, we use the Pareto front approximation approach introduced in [12–14]. In those papers, a new Pareto front approximation method PAINT is introduced and details on decision making with the produced approximation are covered. The PAINT method uses a novel way to integrate the knowledge about Pareto dominance into the approximation. The PAINT method interpolates between a given set of Pareto optimal outcomes to construct a Pareto front approximation. The approach differs from the other approaches for decision making with Pareto front approximations (mentioned above) because it is able to approximate also nonconvex Pareto fronts. Furthermore, the Pareto front approximation constructed with PAINT implies a multiobjective mixed integer linear surrogate problem (for the original problem) that can be solved with any interactive method. The other approaches are either applicable only to convex multiobjective optimization problems or use only a custom-made procedure for choosing a preferred point on the approximation. Further details on the PAINT method are covered in Section 2.

## 1.4 Designing and Operating a Wastewater Treatment Plant

Designing and operating a wastewater treatment plant is a complex problem with many conflicting criteria that have to be considered at the same time. In this paper, we consider a plant using so-called activated sludge process, which is globally the most common method of wastewater treatment. We model the problem as a five-objective optimization problem, which was previously studied also in [37]. The five-objective problem is an extension of the three-objective problem treated in [10]. The approach of this paper differs from the approach of [37] because we use the PAINT method to approximate the Pareto front and to construct a surrogate problem for the original problem. In this way, the time that the decision maker has to wait while using an interactive method becomes shorter.

Figure 1 shows the schematic layout of the wastewater treatment plant that was designed in [37]. The wastewater treatment begins with grit removal. After the grit removal, solids are separated by a gravitational settling. Raw and mixed sludge removed from the primary settlers is fermented in a separate reactor and partly recycled back to the water line to provide readily biodegradable carbon source for denitrification. The bioreactor consists of four anoxic zones, three aerobic zones and one deoxygenation zone. Nitrate-rich activated sludge is recycled from zone 8 of the bioreactor to zone 1. Return sludge and primary effluent are directed to zone 1. Methanol is injected to zone 2 to support denitrification. Excess sludge is pumped from zone 8 of the bioreactor to the beginning of the water process, from which it is removed in the primary settlers together

with raw sludge. Raw and mixed sludge is thickened gravitationally into approximately 4.5% total solids prior to anaerobic digestion. Anaerobic digestion produces biogas and the produced biogas can be converted into electrical or thermal energy. The digested sludge is dewatered by centrifuges into approximately 28% total solids. The reject water from sludge treatment is pumped to the beginning of the plant. The wastewater treatment process is simulated with the commercial GPS-X simulator (see [8]) and the model is based on the findings of Pöyry Engineering Ltd. For more information about the wastewater treatment plants using activated sludge process, see e.g., [10, 32, 37].

The objectives of the optimization problem are the amount of nitrogen in the effluent ( $g/m^3$ , grams per a cubic meter of effluent), aeration power consumption in the activated sludge process ( $kW$ ), chemical consumption ( $g/m^3$ , grams per a cubic meter of effluent), excess sludge production ( $kg/d$ , kilograms per day) and biogas production ( $m^3/d$ , cubic meters per day). The first one is the main goal of activated sludge process and the four others are connected to the operational costs. This multiobjective optimization problem allows the simultaneous consideration of the performance of the plant (through the nitrogen removal rate) and different aspects of the operational costs. Naturally, the last objective is maximized and the others are minimized. Decision variables of the problem are the percentage of inflow pumped to fermentation, the amount of excess sludge removed, the dissolved oxygen setpoint in the last aerobic zone and the methanol dose. Thus, the methanol dose is both a decision variable and an objective.

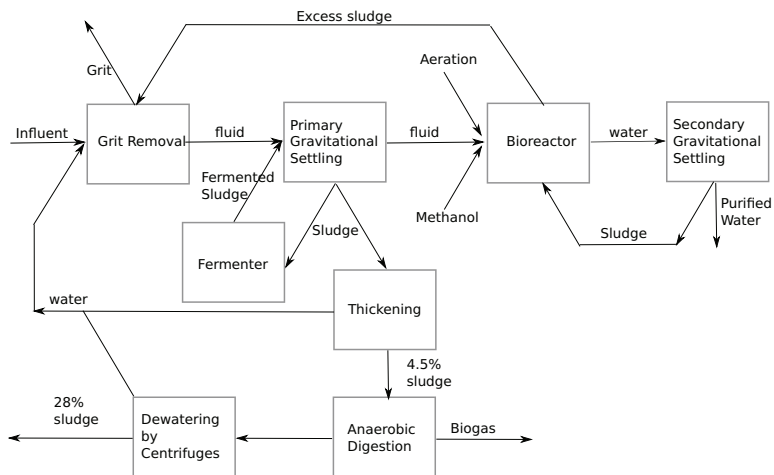


Figure 1: A schematic layout of the wastewater treatment plant

Each simulation of the wastewater treatment plant of [37] took about 11 seconds on the GPS-X simulator. This made the problem computationally expensive. In addition, one could notice from the Pareto optimal outcomes computed for the problem that the problem is nonconvex. During the analysis in [37], 200 simulations were run to optimize the scalarizations (i.e., single objective optimization problems, whose optimal solutions are Pareto optimal

solutions to the multiobjective optimization problem) given by the interactive NIMBUS method that was used to solve the problem. This means that each iteration of the interactive method took more than half an hour. Even though interesting solutions to the problem were found in [37], the computational time of iterations was an inconvenience to the decision maker (according to personal communications with the authors of [37]). This means that there was room for improvement using the PAINT method.

## 2 The PAINT Approach to Solving Computationally Expensive Problems

In this section, we describe the PAINT approach to solving computationally expensive problems. The applicability of the PAINT approach is then demonstrated in Section 4 by solving a computationally expensive multiobjective optimization problem of wastewater treatment plant design and operation.

The PAINT approach is based on the Pareto front approximation constructed by the PAINT method. The PAINT method was proposed in [13], and it is based on the concept of an inherently nondominated Pareto front approximation introduced in [14] and the mathematical concepts of [12]. The PAINT method interpolates between a given set of Pareto optimal outcomes in a way that the interpolants neither dominate nor are dominated by the set of given Pareto optimal outcomes and, in addition, they are not dominated by each other (i.e., the interpolation is an inherently nondominated Pareto front approximation, as defined in [14]). In this paper, a vector on the Pareto front approximation is called an approximate (Pareto optimal) outcome.

The general functionality of the PAINT method is as follows: The PAINT method first constructs the Delaunay triangulation of the given set of Pareto optimal outcomes and then chooses the appropriate polytopes from it to the Pareto front approximation. In this paper, this is realized with the Octave-based (see [5, 31]) implementation that was developed during the research of [13].

The Pareto front approximation constructed with the PAINT method implies a computationally inexpensive mixed integer linear surrogate problem for the original problem, as described in [13]. The Pareto front of the surrogate problem is exactly the Pareto front approximation and, thus, a preferred solution to the surrogate problem implies a preferred vector on the Pareto front approximation, which is also called a preferred approximate outcome in this paper. The algorithm of the PAINT method and more exact details can be found in [13].

Decision making in the PAINT approach is described in Figure 2. In the PAINT approach to solving computationally expensive problems, we assume that there exists a set of Pareto optimal solutions to the computationally expensive problem. This set may have been generated with any a posteriori method. The set of the related outcomes is inputted into the PAINT method. PAINT then interpolates between the set of given Pareto optimal outcomes and outputs the interpolation that implies a mixed integer linear surrogate problem



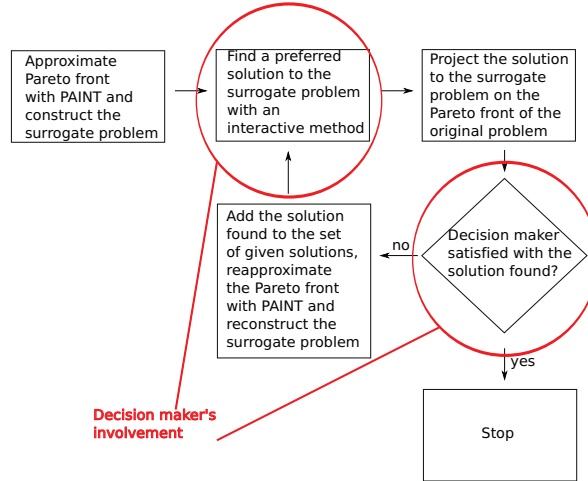


Figure 2: A flowchart of the decision making process

for the original problem.

After the mixed integer linear surrogate problem has been formulated, the decision maker gets involved and uses an interactive method of his/her choice to find a preferred solution to the surrogate problem. The outcomes given by Pareto optimal solutions to the surrogate problem are vectors on the Pareto front approximation, which is in the same space as the original Pareto front. Thus, the decision maker is able to give his/her preferences on them. The preferred approximate outcome is projected on the actual Pareto front of the original problem by solving achievement scalarizing problem (see [40, 41]) with the approximate outcome as a reference point. More details on the projection can be found in [13]. Projecting the solution may take time, depending on the computational costs of the problem. If the problem is very computationally expensive, the projection can be done without the involvement of the decision maker.

The projection of the preferred approximate outcome (i.e., a Pareto optimal solution to the original problem) is shown to the decision maker and, if he/she is satisfied, the decision making process stops, because a preferred solution has been found. If the decision maker is not satisfied, it is possible to update the Pareto front approximation by adding the new Pareto optimal outcome to the given set of Pareto optimal outcomes and by recomputing the approximation with the PAINT method. This yields a more accurate approximation and we can again use an interactive method to find a preferred solution to the new (more accurate) surrogate problem. This process can be repeated as many times as necessary.

The PAINT method is a powerful tool as it can interpolate between any given set of Pareto optimal outcomes, i.e., the way that the outcomes have been generated does not affect the functionality of the method. In addition, since it is based on the concept of inherent nondominance (see [14]), it will not

provide interpolants that would mislead the decision maker. Finally, the mixed integer linear surrogate problem implied by the approximation allows one to use any interactive method for finding a preferred approximate outcome on the Pareto front approximation.

The PAINT method has a couple of shortcomings, already noted in [13]. First, the PAINT method does not provide any information about the preimage of the Pareto front approximation in the decision space. This means that the decision maker has to project the approximate outcome (i.e., the solution to the surrogate problem) on the Pareto front of the original problem in order to find out the values of the decision variables. Second, the PAINT method cannot detect any disconnectedness in the Pareto front, but always interpolates between the outcomes whenever the interpolation is inherently nondominated. Thus, the approximation might be inaccurate if e.g., the decision space is disconnected or the objective functions are highly nonconvex.

### 3 A New IND-NIMBUS<sup>®</sup> PAINT Module

IND-NIMBUS<sup>®</sup> (see [23]) is a multi-platform desktop software framework, currently available for Windows and Linux operating systems, intended to provide a flexible tool-set for implementation of multiobjective optimization methods. So far, the IND-NIMBUS framework has been used to implement the synchronous NIMBUS [24–26] and the Pareto Navigator [6] methods. The IND-NIMBUS<sup>®</sup> software can be connected to an external sources that model the problem, such as the GPS-X simulator used for modeling the wastewater treatment plant. For this paper, the IND-NIMBUS<sup>®</sup> software framework has been used to develop a so-called IND-NIMBUS<sup>®</sup> PAINT module that combines the PAINT and NIMBUS methods for computationally expensive multiobjective optimization. The PAINT module implements most of the functionalities described in Figure 2.

The synchronous NIMBUS method [24–26] is an interactive multiobjective optimization method. The NIMBUS method uses classification of objectives as the preference information. Given a Pareto optimal solution to the multiobjective optimization problem, the decision maker can classify the objectives into classes  $I^<$ ,  $I^{\leq}$ ,  $I^=$ ,  $I^{\geq}$  and  $I^>$  defined, respectively, as classes of objective functions that the decision maker wants to improve as much as possible, wants to improve to a given aspiration level  $z_i$ , allows to remain unchanged, allows to deteriorate until a given bound  $\epsilon_i$  and allows to change freely for a while. This preference information is converted into several different single objective subproblems with the help of different scalarization functions as proposed in [26]. These subproblems are solved to generate different Pareto optimal solutions, which are shown to the decision maker who can then see how well the desired preferences could be attained. The decision maker can choose any of these solutions as the starting point of the next iteration, i.e., classification. This iterative procedure can either start with a solution given by the decision maker or from a so-called neutral compromise solution and it is repeated until the decision maker is satisfied with the solution at hand. Further information about the synchronous NIMBUS with other means to direct the search process

is given in [26].

The NIMBUS method has been successfully applied to shape design of ultrasonic transducers [16], designing a paper machine headbox [11], optimal control in continuous casting of steel [28], separation of glucose and fructose [9], intensity modulated radiotherapy treatment planning [34], brachytherapy [35] and optimizing heat exchanger network synthesis [20], among others. In addition, it uses classification of objectives that has been found cognitively just [19]. These facts make the NIMBUS method an ideal choice as the interactive method for solving the PAINT surrogate problem of the wastewater treatment plant model.

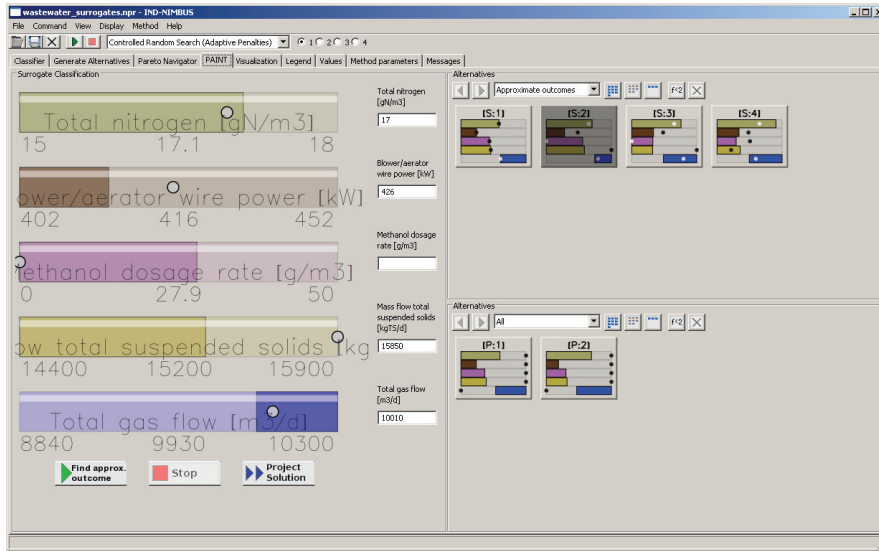


Figure 3: A screen shot of the IND-NIMBUS<sup>®</sup> PAINT module

Figure 3 shows the screen shot of the IND-NIMBUS<sup>®</sup> PAINT module. In the PAINT module, the decision maker can give his/her preferences concerning the surrogate problem by classification of the objective functions. The given classification information is used to formulate a single objective subproblem of the surrogate problem. The subproblem is modeled using the Optimizing Programming Language (OPL, see [39]), and this (mixed integer linear model) is solved using CPLEX (see [17]). An optimal solution to the subproblem gives a new approximate Pareto optimal outcome, corresponding to the preferences given by the decision maker. This approximate outcome is shown to the decision maker. If the decision maker so wishes, he reclassify the objectives of the new approximate outcome which yields another approximate outcome.

As described in Section 2, approximate Pareto optimal outcomes can be projected on the Pareto front of the original problem using the PAINT module (using the Project Solution button near the bottom of the screen). The projection of the approximate outcome, that is, a Pareto optimal solution to

the original problem is shown to the decision maker. As mentioned, for a computationally expensive problem this may take time, but fortunately projecting an approximate outcome can be done without the involvement of the decision maker.

The approximate Pareto optimal outcomes and the actual Pareto optimal solutions that have been found during the decision making process are visualized on the right side of the PAIN module. The decision maker can choose any of the approximate Pareto optimal outcomes as the starting point of the next NIMBUS iteration (i.e., as the basis for a new classification of objectives). The process stops when the decision maker has found a preferred solution to the original problem through projecting an approximate Pareto optimal outcome.

In the current version of the PAIN module, only one of the scalarizations of the synchronous NIMBUS method (i.e., the achievement scalarizing subproblem) has been implemented. That is, unlike in the synchronous NIMBUS method, the decision maker can see only one approximate Pareto optimal outcome for given preferences. It should also be noted that any solver capable of solving the surrogate subproblem (e.g., GLPK, see [7]) could be used instead of CPLEX.

The current version of the PAIN module does not implement the construction or updating of the surrogate problem. If one wishes to update the surrogate problem using Pareto optimal outcomes obtained by e.g., projection, the decision making process must be stopped, and the surrogate problem must be manually updated using Octave. In future versions, updating the Pareto front approximation should be implemented under a third button in the PAIN module that would then automatically update the approximation and the surrogate problem.

## 4 Solving the Wastewater Treatment Case

In this section, we demonstrate how the PAIN method and the IND-NIMBUS<sup>®</sup> PAIN module were used to solve the wastewater treatment problem, described in Section 1.4. First, in Section 4.1, we describe the construction of the Pareto front approximation with PAIN before the involvement of the decision maker. Then, in Section 4.2, we describe how the decision maker used the PAIN module to solve the wastewater treatment problem.

### 4.1 Pre-Decision Making phase

First, a set of 200 mutually nondominated solutions to the wastewater problem was found with the evolutionary UPS-EMO algorithm (introduced in [3]) and the GPS-X simulator. To study the optimality of these solutions, each one was locally improved using an achievement scalarizing problem [40, 41], which was optimized with Matlab `fmincon`-function with finite differences approximated gradients. This resulted in 195 mutually nondominated solutions. The maximal improvement in the values of the achievement scalarizing problem was at most 3% so the local improvement did not cause much change. This built our confidence that the final solutions were close to Pareto optimal. We took the

outcomes given by these solutions as the set of given Pareto optimal outcomes for the PAINT method. The whole process of producing this set took about three days on a standard laptop.

After this, we computed a Pareto front approximation based on the given set of Pareto optimal outcomes with the PAINT method (see Section 2). The PAINT method chose 4272 polytopes for interpolation in the Pareto front. In order to reduce the computational complexity of the implied mixed integer linear surrogate problem, we removed polytopes that were subsets of larger polytopes from the approximation. This resulted in a collection of 608 polytopes whose union covered the same space in  $\mathbb{R}^5$  as that of the larger collection. In addition, all sets of vertices of the polytopes in the collection were affinely independent and, thus, the number of vertices of all the polytopes was five or less, as shown in [12]. Using the PAINT method to construct the Pareto front approximation took approximately 19 hours on Intel<sup>®</sup> Xeon<sup>®</sup> E5410 CPU.

The mixed integer linear surrogate problem implied by the smaller collection was equivalent to that implied by the larger collection, but it was computationally less expensive. As described in [13], the surrogate problem could be written as

$$\begin{aligned}
& \min && (z_1, \dots, z_5) \\
& \text{s.t.} && \sum_{j=1}^{608} \sum_{l=1}^5 \lambda_{j,l} = 1 \\
& && \sum_{l=1}^5 \lambda_{j,l} \leq y_j, \text{ for all } j = 1, \dots, 608 \\
& && \sum_{j=1}^{608} y_j = 1 \\
& \text{where} && \lambda \in [0, 1]^{608 \times 5} \\
& && y \in \{0, 1\}^{608} \\
& && z_i = \sum_{j=1}^{608} \sum_{l=1}^5 \lambda_{j,l} p_i^{A_{l,j}} \text{ for all } i = 1, \dots, 5,
\end{aligned} \tag{2}$$

where each row of the matrix  $A \in \mathbb{R}^{608 \times 5}$  contained the indices of the vertices of one polytope in the smaller collection of polytopes. The component  $\lambda_{j,l}$  of the matrix variable  $\lambda \in \mathbb{R}^{608 \times 5}$  was for all  $j = 1, \dots, 608$  and  $l = 1, \dots, 5$  the coefficient of the vertex  $l$  of the polytope given by row  $j$  in the matrix  $A$ . The variable  $y$  determined which of the rows of the matrix  $\lambda$  was nonzero. By the third constraint, only one row in the matrix  $\lambda$  had nonzero elements.

Problem (2) had  $608 \times 5 = 3040$  continuous variables and 608 binary variables. CPLEX was able to solve e.g., an achievement scalarizing problem for the surrogate problem in less than a second. This was a tremendous improvement to solving a scalarization of the original problem, which took about half an hour with the Controlled Random Search algorithm (see [33]), which is implemented in the IND-NIMBUS software.

## 4.2 Decision Making Phase

Using the PAINT module, our decision maker (Mr. Kristian Sahlstedt from Pöyry Environment Ltd) was able to examine the approximate outcomes and to project any of them on the Pareto front of the original problem. This entire decision making process was done within a couple of hours and the decision maker's involvement was only about an hour, which could not have been possible by merely using the original computationally expensive problem.

Out- come	Amount of Nitrogen [ $gN/m^3$ ] (min)	Aeration power [ $kW$ ] (min)	Chemical consump- tion [ $g/m^3$ ] (min)	Excess sludge [ $kg/d$ ] (min)	Biogas production [ $m^3/d$ ] (max)
$s^1$	16.67	412.2	21.89	15060	9731
$s^2$	17.13	416.3	27.86	15250	9935
$s^3$	17.30	419.0	16.27	14870	9560
$s^4$	17.74	414.6	14.41	14910	9571
$p^1$	16.80	414.1	18.24	14960	9626
$p^2$	17.10	411.6	15.10	14860	9529

Table 1: The approximate and actual Pareto optimal outcomes inspected by the decision maker with the PAINT module

Before the decision maker started using the PAINT module, we gave him a brief overview of the methods from the user’s perspective. We told him that a set of Pareto optimal outcomes has been computed and that a new PAINT method has been used to interpolate between those outcomes. We also informed him that the outcomes given by PAINT are only approximate Pareto optimal outcomes and, thus, more computation has to be done to find the closest real Pareto optimal outcome. In addition, we told him that the PAINT method does not unfortunately provide any information about the decision variables and those values can only be known after the real Pareto optimal solution is found. Since our decision maker had previous experiences with the NIMBUS method, all of this was very clear to him. In addition, he did not find any of this too inconvenient.

Table 1 shows the approximate Pareto optimal outcomes generated (approximate outcomes  $s^1, \dots, s^4$ ) and the outcomes given by actual Pareto optimal solutions to the wastewater treatment problem (outcomes  $p^1, p^2$ ) that were inspected by the decision maker. The decision making process started from the approximate outcome  $s^1$  in Table 1. The outcome  $s^1$  was given by the neutral compromise solution to the surrogate problem.

The decision maker wanted to see further (approximate) Pareto optimal outcomes. After a classification of objectives in the PAINT module, the optimal solution to the new subproblem for the surrogate problem gave the approximate Pareto optimal outcome  $s^2$ . The approximate outcome  $s^2$  has more biogas production than the approximate outcome  $s^1$ , but is worse in all the other objectives. Especially, the chemical consumption is very large. Let us emphasize that finding the approximate outcome  $s^2$  was especially smooth, since the mixed integer linear problem was computationally inexpensive.

Because the decision maker was not completely satisfied with the approximate outcome  $s^2$ , he decided to continue and find another approximate outcome. This yielded the approximate outcome  $s^3$ , which has much lower chemical consumption and a slightly lower excess sludge production than both approximate outcomes  $s^1$  and  $s^2$ . Unfortunately, the approximate outcome  $s^3$  is worse

than both approximate outcomes  $s^1$  and  $s^2$  in all the other objectives.

A classification of objectives of the approximate outcome  $s^3$  and solving the new subproblem for the surrogate problem led to the approximate outcome  $s^4$ . This new approximate outcome is roughly the same as the approximate outcome  $s^3$  in both excess sludge production and biogas production. However, the amount of nitrogen for the approximate outcome  $s^4$  is slightly higher than for the approximate outcome  $s^3$ , but this is compensated by chemical consumption and aeration power that are considerably smaller.

After having inspected the four approximate Pareto optimal outcomes, the decision maker felt that he had learned enough about the surrogate problem. First, the decision maker decided to project the approximate outcome  $s^1$  on the Pareto front of the original wastewater treatment problem. The projection of the approximate outcome  $s^1$  (i.e., solution to the achievement scalarizing problem with the approximate outcome  $s^1$  as the reference point) took a little over half an hour. The projection was done using the GPS-X simulator and the Controlled Random Search algorithm. The projection was outcome  $p^1$  in Table 1. According to our decision maker's assessment, the outcome  $p^1$  was fairly close to the approximate outcome  $s^1$  in all objectives. However, he felt that there might still be more preferred solutions to the problem.

Because the approximate outcome  $s^1$  and the actual Pareto optimal outcome  $p^1$  were close to each other, it was decided not to update the Pareto front approximation. Instead, the decision maker wanted to project the approximate outcome  $s^4$  and obtained the outcome  $p^2$  in Table 1. The Pareto optimal outcome  $p^2$  has slightly higher amount of nitrogen in the effluent than the Pareto optimal outcome  $p^1$ , but it has considerably lower aeration power and the amount of chemical consumption. The Pareto optimal outcome  $p^2$  was very preferred by the decision maker and he chose it as the final solution to the problem.

The final solution (including a way to design and operate a wastewater treatment plant) will be further inspected by more accurate simulators before implementing. However, it will act as a guideline for the design of the wastewater treatment plant.

## 5 Analyzing the Decision Making Process with PAINT

We filmed our decision maker Mr. Sahlstedt during the decision making process and asked him some additional questions regarding the usability of the methods. The purpose of the video was to reveal any issues that he might have had while using the PAINT module and to find out whether any aspects of the PAINT method that were hard to understand.

When analyzing the video, it seems that the key point in the usability of the PAINT method and the PAINT module is informing the decision maker about the approximate nature of the method. In order to make the PAINT method more usable, it would be a good idea to produce an introductory video introducing the key points of the method. Since our decision maker was already

familiar with the IND-NIMBUS<sup>®</sup> software, there was no need to introduce it. This may not always be the case and, thus, the video should also include a short introduction to the NIMBUS method and the IND-NIMBUS<sup>®</sup> software.

The decision maker thought that the PAINT method and the PAINT module were easy and intuitive to use. In addition, he said that the PAINT method provided a definite improvement to merely using the NIMBUS method because of the faster computational times (a couple of seconds against half an hour) between the iterations of the interactive method. He thought that neither the approximate nature nor the fact that the preferences had to be based only on the objective function values were big drawbacks. In addition, no approximate outcomes that the decision maker would have assessed implausible were found during the decision making process.

For both projected approximate outcomes  $s^1$  and  $s^4$ , our decision maker assessed that the actual Pareto optimal solutions were close enough, taking into account the uncertainties in the model, and there was no need to recompute the Pareto front approximation. Thus, the inability of the PAINT module in reconstructing the Pareto front approximation (as mentioned in Section 3) was not an issue.

Except for correcting a couple of minor bugs from the software, the decision maker did not offer any improvements. He did, however, agree with us that the PAINT method should be able to detect disconnectedness in the Pareto front and that the decision variables should be somehow approximated, too. However, detecting disconnectedness was not an issue on this occasion, because all the approximate Pareto optimal outcomes found seemed plausible to the decision maker and the approximate Pareto optimal outcomes that were projected were rather close to their projections. Finally, since one of the decision variables was also an objective (i.e., the methanol dose), that was approximated in our problem, although the PAINT method does not in general do this.

## 6 Conclusions

In this paper, we described how the PAINT method can be used to speed up iterations of interactive methods when solving computationally expensive multiobjective optimization problems. The PAINT method was used to construct a Pareto front approximation that then implied a mixed integer linear surrogate problem for the original problem. As our case problem, we studied a five-objective optimization problem of designing and operating a wastewater treatment plant. In addition, we introduced a new IND-NIMBUS<sup>®</sup> PAINT module that combines the PAINT method and the interactive NIMBUS method.

The PAINT method and the PAINT module worked well in this problem. The decision maker found it easy and intuitive to use the interactive NIMBUS method to find a preferred approximate outcome on the Pareto front approximation. The low computational cost of using the interactive method with the surrogate problem was a definite improvement to using interactive method directly to solve the computationally expensive wastewater treatment problem.

The experimental design in this paper was unique: Because our decision maker had already used the IND-NIMBUS<sup>®</sup> to study the same wastewater



treatment problem, he was able to compare the experiences of using the IND-NIMBUS software with and without the PAINT method. According to the decision maker's opinion, the PAINT method provided a significant improvement. This implies that the PAINT method should be also applicable to other computationally expensive problems.

The IND-NIMBUS<sup>®</sup> PAINT module is still in the development phase and, thus, it lacks some essential functionality (like the implementations of the other scalarizations of the synchronous NIMBUS method) and, also, it still has some bugs. If there had been no bugs in the software, the investigation of the problem with the PAINT module would have been even more fluent. Further effort has to be put in correcting these bugs.

The PAINT method requires one to use additional methods and software to generate the given set of Pareto optimal outcomes and to solve the mixed integer linear surrogate problem. In this paper, we used the UPS-EMO algorithm to generate the Pareto optimal outcomes and the IND-NIMBUS software with the PAINT module and the CPLEX solver to solve the surrogate problem. In future research, other applicable methods and software can be also used together with the PAINT method.

## 7 Acknowledgements

The authors of this paper were financially supported by the Academy of Finland (grant number 128495). The authors are indebted to Mr. Kristian Sahlstedt from Pöyry Environment Ltd for his expertise on the problem and for acting as the decision maker. In addition, the authors wish to thank Drs. Jussi Hakanen and Timo Aittokoski from the Research Group in Industrial Optimization at the University of Jyväskylä, Department of Mathematical Information Technology for helping to solve the problem. During the writing of this paper, Prof. Mieltinen provided her helpful comments that helped the authors improve this paper.

## References

- [1] H. Ackermann, A. Newman, H. Röglin, and B. Vöcking. Decision-making Based on Approximate and Smoothed Pareto Curves. *Theoretical Computer Science*, 378:253–270, 2007.
- [2] T. Aittokoski and K. Miettinen. Cost Effective Simulation-Based Multiobjective Optimization in the Performance of an Internal Combustion Engine. *Engineering Optimization*, 40:593–612, 2008.
- [3] T. Aittokoski and K. Miettinen. Efficient Evolutionary Approach to Approximate the Pareto-Optimal Set in Multiobjective Optimization, UPS-EMOA. *Optimization Methods and Software*, 25:841 – 858, 2010.
- [4] V. E. Bezerkin, G. K. Kamenev, and A. V. Lotov. Hybrid Adaptive Methods for Approximating a Nonconvex Multidimensional Pareto Fron-

- tier. *Computational Mathematics and Mathematical Physics*, 46:1918–1931, 2006.
- [5] J. W. Eaton. *GNU Octave Manual*. Network Theory Limited, 2002.
- [6] P. Eskelinen, K. Miettinen, K. Klamroth, and J. Hakanen. Pareto Navigator for Interactive Nonlinear Multiobjective Optimization. *OR Spectrum*, 32:211–227, 2010.
- [7] GLPK (GNU Linear Programming Kit) Home Page, Last accessed: September 16, 2011. url = <http://www.gnu.org/s/glpk/>.
- [8] GPS-X Simulator Home Page, Last accessed: September 16, 2011. url = [http://www.technotrade.com.pk/20/GPS-X\\_Waste\\_Water\\_Treatment\\_Simulation\\_Software/](http://www.technotrade.com.pk/20/GPS-X_Waste_Water_Treatment_Simulation_Software/).
- [9] J. Hakanen, Y. Kawajiri, K. Miettinen, and L.T. Biegler. Interactive Multi-Objective Optimization for Simulated Moving Bed Processes. *Control and Cybernetics*, 36(2):282–320, 2007.
- [10] J. Hakanen, K. Miettinen, and K. Sahlstedt. Wastewater Treatment: New Insight Provided by Interactive Multiobjective Optimization. *Decision Support Systems*, 51:328–337, 2011.
- [11] J.H. Hämäläinen, K. Miettinen, P. Tarvainen, and J. Toivanen. Interactive Solution Approach to a Multiobjective Optimization Problem in Paper Machine Headbox Design. *Journal of Optimization Theory and Applications*, 116:265–281, 2003.
- [12] M. Hartikainen, K. Miettinen, and M. M. Wiecek. Constructing a Pareto Front Approximation for Decision Making. *Mathematical Methods of Operations Research*, 73:209–234, 2011.
- [13] M. Hartikainen, K. Miettinen, and M. M. Wiecek. PAINT: An Interpolation Method for Computationally Expensive Multiobjective Optimization Problems, 2011. University of Jyväskylä, Reports of the Department of Mathematical Information Technology, Series B. Scientific Computing, Number B3/2011.
- [14] M. Hartikainen, K. Miettinen, and M. M. Wiecek. Pareto Front Approximations for Decision Making with Inherent Non-dominance. In Y. Shi, S. Wang, G. Kou, and J. Wallenius, editors, *New State of MCDM in the 21st Century, Selected Papers of the 20th International Conference on Multiple Criteria Decision Making 2009*, pages 35–46. Springer-Verlag Berlin, Heidelberg, 2011.
- [15] M. Hasenjäger and B. Sendhoff. Crawling Along the Pareto Front: Tales From the Practice. In *The 2005 IEEE Congress on Evolutionary Computation (IEEE CEC 2005)*, pages 174–181, Piscataway, NJ, 2005. IEEE Press.

- [16] E. Heikkola, K. Miettinen, and P. Nieminen. Multiobjective Optimization of an Ultrasonic Transducer Using NIMBUS. *Ultrasonics*, 44:368–380, 2006.
- [17] IBM ILOG CPLEX Optimization Studio Home Page, Last accessed: September 16, 2011. url = <http://www-01.ibm.com/software/integration/optimization/cplex-optimization-studio/>.
- [18] P. Korhonen and J. Wallenius. Behavioural Issues in MCDM: Neglected Research Questions. *Journal of Multi-Criteria Decision Analysis*, 5:178–182, 1996.
- [19] O. I. Larichev. Cognitive Validity in Design of Decision-Aiding Techniques. *Journal of Multi-Criteria Decision Analysis*, 3:127 – 138, 1992.
- [20] T. Laukkanen, T.-M. Tveit, V. Ojalehto, K. Miettinen, and C.-J. Fogelholm. An Interactive Multi-Objective Approach to Heat Exchanger Network Synthesis. *Computers & Chemical Engineering*, 34:943–952, 2010.
- [21] A. V. Lotov, V. A. Bushenkov, and G. A. Kamenev. *Interactive Decision Maps*. Kluwer Academic Publishers, Boston, 2004.
- [22] K. Miettinen. *Nonlinear Multiobjective Optimization*. Kluwer Academic Publishers, Boston, 1999.
- [23] K. Miettinen. IND-NIMBUS for Demanding Interactive Multiobjective Optimization. In T. Trzaskalik, editor, *Multiple Criteria Decision Making'05*, pages 137–150. The Karol Adamiecki University of Economics in Katowice, Katowice, 2006.
- [24] K. Miettinen and M. Mäkelä. Interactive Bundle-based Method for Nondifferentiable Multiobjective Optimization: NIMBUS. *Optimization*, 34:231–246, 1995.
- [25] K. Miettinen and M. Mäkelä. Interactive Multiobjective Optimization System WWW-NIMBUS on the Internet. *Computers & Operations Research*, 27:709–723, 2000.
- [26] K. Miettinen and M. M. Mäkelä. Synchronous Approach in Interactive Multiobjective Optimization. *European Journal of Operational Research*, 170:909–922, 2006.
- [27] K. Miettinen, F. Ruiz, and A. P. Wierzbicki. Introduction to Multiobjective Optimization: Interactive Approaches. In J. Branke, K. Deb, K. Miettinen, and R. Slowinski, editors, *Multiobjective Optimization: Interactive and Evolutionary Approaches*, pages 27–57. Springer-Verlag Berlin, Heidelberg, 2008.
- [28] Kaisa Miettinen. Using Interactive Multiobjective Optimization in Continuous Casting of Steel. *Materials and Manufacturing Processes*, 22:585–593, 2007.

- [29] M. Monz, K. H. Kufer, T. R. Bortfeld, and C. Thieke. Pareto Navigation - Algorithmic Foundation of Interactive Multi-Criteria IMRT Planning. *Physics in Medicine and Biology*, 53:985–998, 2008.
- [30] H. Nakayama, Y. Yun, and M. Yoon. *Sequential Approximate Multiobjective Optimization Using Computational Intelligence*. Springer-Verlag Berlin, Heidelberg, 2009.
- [31] Octave Home Page. url = [www.gnu.org/software/octave/](http://www.gnu.org/software/octave/), Last accessed: September 16, 2011.
- [32] H. M. Phillips, K. E. Sahlstedt, K. Frank, J. Bratby, W. Brennan, S. Rogowski, D. Pier, W. Anderson, M. Mulas, J. B. Copp, and N. Shirodkar. Wastewater Treatment Modelling in Practice: a Collaborative Discussion of the State of the Art. *Water Science and Technology*, 59:695–704, 2009.
- [33] W. L. Price. Global Optimization by Controlled Random Search. *Journal of Optimization Theory and Applications*, 40:333–348, 1983. 10.1007/BF00933504.
- [34] H. Ruotsalainen, E. Boman, K. Miettinen, and J. Tervo. Nonlinear Interactive Multiobjective Optimization Method for Radiotherapy Treatment Planning with Boltzmann Transport Equation. *Contemporary Engineering Sciences*, 2:391–422, 2009.
- [35] H. Ruotsalainen, K. Miettinen, J.-E. Palmgren, and T. Lahtinen. Interactive Multiobjective Optimization for Anatomy based Three-Dimensional HDR Brachytherapy. *Physics in Medicine and Biology*, 55:4703–4719, 2010.
- [36] S. Ruzika and M. M. Wiecek. Approximation Methods in Multiobjective Programming. *Journal of Optimization Theory and Applications*, 126:473–501, 2005.
- [37] K. Sahlstedt, J. Hakanen, and K. Miettinen. Interactive Multiobjective Optimization in Wastewater Treatment Plant Operation and Design. In *Proceedings of the ECWATECH-2010, the IWA Specialist Conference 'Water and Wastewater Treatment Plants in Towns and Communities of the XXI Century: Technologies, Design and Operation', CD-ROM*, 2010.
- [38] Y. Sawaragi, H. Nakayama, and T. Tanino. *Theory of Multiobjective Optimization*. Academic Press, Orlando, 1985.
- [39] P. van Hentenryck. *The OPL Optimization Programming Language*. The MIT Press, Cambridge, 1999.
- [40] A. P. Wierzbicki. The Use of Reference Objectives in Multiobjective Optimization. In G. Fandel and T. Gal, editors, *Multiple Criteria Decision Making Theory and Applications*, pages 468–486. Springer-Verlag Berlin, Heidelberg, 1980.

- [41] A. P. Wierzbicki. On the Completeness and Constructiveness of Parametric Characterizations to Vector Optimization Problems. *OR Spectrum*, 8:73–87, 1986.
- [42] B. Wilson, D. Cappelleri, T. W. Simpson, and M. Frecker. Efficient Pareto Frontier Exploration Using Surrogate Approximations. *Optimization and Engineering*, 2:31–50, 2001.
- [43] L. Xu, T. Reinikainen, W. Ren, B. P. Wang, Z. Han, and D. Agonafer. A Simulation-Based Multi-Objective Design Optimization of Electronic Packages Under Thermal Cycling and Bending. *Microelectronics Reliability*, 44:1977 – 1983, 2004.
- [44] H. Yapicioglu, H. Liu, A. E. Smith, and G. Dozier. Hybrid Approach for Pareto Front Expansion in Heuristics. *Journal of Operational Research Society*, 62:348–359, 2011.

JYVÄSKYLÄ STUDIES IN COMPUTING

- 1 ROPPONEN, JANNE, Software risk management - foundations, principles and empirical findings. 273 p. Yhteenveto 1 p. 1999.
- 2 KUZMIN, DMITRI, Numerical simulation of reactive bubbly flows. 110 p. Yhteenveto 1 p. 1999.
- 3 KARSTEN, HELENA, Weaving tapestry: collaborative information technology and organisational change. 266 p. Yhteenveto 3 p. 2000.
- 4 KOSKINEN, JUSSI, Automated transient hypertext support for software maintenance. 98 p. (250 p.) Yhteenveto 1 p. 2000.
- 5 RISTANIEMI, TAPANI, Synchronization and blind signal processing in CDMA systems. - Synkronointi ja sokea signaalinkäsittely CDMA järjestelmässä. 112 p. Yhteenveto 1 p. 2000.
- 6 LAITINEN, MIKA, Mathematical modelling of conductive-radiative heat transfer. 20 p. (108 p.) Yhteenveto 1 p. 2000.
- 7 KOSKINEN, MINNA, Process metamodelling. Conceptual foundations and application. 213 p. Yhteenveto 1 p. 2000.
- 8 SMOLIANSKI, ANTON, Numerical modeling of two-fluid interfacial flows. 109 p. Yhteenveto 1 p. 2001.
- 9 NAHAR, NAZMUN, Information technology supported technology transfer process. A multi-site case study of high-tech enterprises. 377 p. Yhteenveto 3 p. 2001.
- 10 FOMIN, VLADISLAV V., The process of standard making. The case of cellular mobile telephony. - Standardin kehittämisen prosessi. Tapaus-tutkimus solukoverkkoon perustuvasta matkapuhelintekniikasta. 107 p. (208 p.) Yhteenveto 1 p. 2001.
- 11 PÄIVÄRINTA, TERO, A genre-based approach to developing electronic document management in the organization. 190 p. Yhteenveto 1 p. 2001.
- 12 HÄKKINEN, ERKKI, Design, implementation and evaluation of neural data analysis environment. 229 p. Yhteenveto 1 p. 2001.
- 13 HIRVONEN, KULLERVO, Towards better employment using adaptive control of labour costs of an enterprise. 118 p. Yhteenveto 4 p. 2001.
- 14 MAJAVA, KIRSI, Optimization-based techniques for image restoration. 27 p. (142 p.) Yhteenveto 1 p. 2001.
- 15 SAARINEN, KARI, Near infra-red measurement based control system for thermo-mechanical refiners. 84 p. (186 p.) Yhteenveto 1 p. 2001.
- 16 FORSELL, MARKO, Improving component reuse in software development. 169 p. Yhteenveto 1 p. 2002.
- 17 VIRTANEN, PAULI, Neuro-fuzzy expert systems in financial and control engineering. 245 p. Yhteenveto 1 p. 2002.
- 18 KOVALAINEN, MIKKO, Computer mediated organizational memory for process control. Moving CSCW research from an idea to a product. 57 p. (146 p.) Yhteenveto 4 p. 2002.
- 19 HÄMÄLÄINEN, TIMO, Broadband network quality of service and pricing. 140 p. Yhteenveto 1 p. 2002.
- 20 MARTIKAINEN, JANNE, Efficient solvers for discretized elliptic vector-valued problems. 25 p. (109 p.) Yhteenveto 1 p. 2002.
- 21 MURSU, ANJA, Information systems development in developing countries. Risk management and sustainability analysis in Nigerian software companies. 296 p. Yhteenveto 3 p. 2002.
- 22 SELEZNYOV, ALEXANDR, An anomaly intrusion detection system based on intelligent user recognition. 186 p. Yhteenveto 3 p. 2002.
- 23 LENSU, ANSSI, Computationally intelligent methods for qualitative data analysis. 57 p. (180 p.) Yhteenveto 1 p. 2002.
- 24 RYABOV, VLADIMIR, Handling imperfect temporal relations. 75 p. (145 p.) Yhteenveto 2 p. 2002.
- 25 TSYMBAL, ALEXEY, Dynamic integration of data mining methods in knowledge discovery systems. 69 p. (170 p.) Yhteenveto 2 p. 2002.
- 26 AKIMOV, VLADIMIR, Domain decomposition methods for the problems with boundary layers. 30 p. (84 p.) Yhteenveto 1 p. 2002.
- 27 SEYUKOVA-RIVKIND, LUDMILA, Mathematical and numerical analysis of boundary value problems for fluid flow. 30 p. (126 p.) Yhteenveto 1 p. 2002.
- 28 HÄMÄLÄINEN, SEPPO, WCDMA Radio network performance. 235 p. Yhteenveto 2 p. 2003.
- 29 PEKKOLA, SAMULI, Multiple media in group work. Emphasising individual users in distributed and real-time CSCW systems. 210 p. Yhteenveto 2 p. 2003.
- 30 MARKKULA, JOUNI, Geographic personal data, its privacy protection and prospects in a location-based service environment. 109 p. Yhteenveto 2 p. 2003.
- 31 HONKARANTA, ANNE, From genres to content analysis. Experiences from four case organizations. 90 p. (154 p.) Yhteenveto 1 p. 2003.
- 32 RAITAMÄKI, JOUNI, An approach to linguistic pattern recognition using fuzzy systems. 169 p. Yhteenveto 1 p. 2003.
- 33 SAALASTI, SAMI, Neural networks for heart rate time series analysis. 192 p. Yhteenveto 5 p. 2003.
- 34 NIEMELÄ, MARKETTA, Visual search in graphical interfaces: a user psychological approach. 61 p. (148 p.) Yhteenveto 1 p. 2003.
- 35 YOU, YU, Situation Awareness on the world wide web. 171 p. Yhteenveto 2 p. 2004.
- 36 TAAATILA, VESA, The concept of organizational competence - A foundational analysis. - Perusteanalyysi organisaation kompetenssin käsitteestä. 111 p. Yhteenveto 2 p. 2004.

- 37 LYYTIKÄINEN, VIRPI, Contextual and structural metadata in enterprise document management. - Konteksti- ja rakennemetatieto organisaation dokumenttien hallinnassa. 73 p. (143 p.) Yhteenveto 1 p. 2004.
- 38 KAARIO, KIMMO, Resource allocation and load balancing mechanisms for providing quality of service in the Internet. 171 p. Yhteenveto 1 p. 2004.
- 39 ZHANG, ZHEYING, Model component reuse. Conceptual foundations and application in the metamodeling-based systems analysis and design environment. 76 p. (214 p.) Yhteenveto 1 p. 2004.
- 40 HAARALA, MARJO, Large-scale nonsmooth optimization variable metric bundle method with limited memory. 107 p. Yhteenveto 1 p. 2004.
- 41 KALVINE, VIKTOR, Scattering and point spectra for elliptic systems in domains with cylindrical ends. 82 p. 2004.
- 42 DEMENTIEVA, MARIA, Regularization in multistage cooperative games. 78 p. 2004.
- 43 MAARANEN, HEIKKI, On heuristic hybrid methods and structured point sets in global continuous optimization. 42 p. (168 p.) Yhteenveto 1 p. 2004.
- 44 FROLOV, MAXIM, Reliable control over approximation errors by functional type a posteriori estimates. 39 p. (112 p.) 2004.
- 45 ZHANG, JIAN, QoS- and revenue-aware resource allocation mechanisms in multiclass IP networks. 85 p. (224 p.) 2004.
- 46 KUJALA, JANNE, On computation in statistical models with a psychophysical application. 40 p. (104 p.) 2004.
- 47 SOLBAKOV, VIATCHESLAV, Application of mathematical modeling for water environment problems. 66 p. (118 p.) 2004.
- 48 HIRVONEN, ARI P., Enterprise architecture planning in practice. The Perspectives of information and communication technology service provider and end-user. 44 p. (135 p.) Yhteenveto 2 p. 2005.
- 49 VARTIAINEN, TERO, Moral conflicts in a project course in information systems education. 320 p. Yhteenveto 1p. 2005.
- 50 HUOTARI, JOUNI, Integrating graphical information system models with visualization techniques. - Graafisten tietojärjestelmävausten integrointi visualisointitekniikoilla. 56 p. (157 p.) Yhteenveto 1p. 2005.
- 51 WALLENIUS, EERO R., Control and management of multi-access wireless networks. 91 p. (192 p.) Yhteenveto 3 p. 2005.
- 52 LEPPÄNEN, MAURI, An ontological framework and a methodical skeleton for method engineering - A contextual approach. 702 p. Yhteenveto 2 p. 2005.
- 53 MATYUKEVICH, SERGEY, The nonstationary Maxwell system in domains with edges and conical points. 131 p. Yhteenveto 1 p. 2005.
- 54 SAYENKO, ALEXANDER, Adaptive scheduling for the QoS supported networks. 120 p. (217 p.) 2005.
- 55 KURJENNIEMI, JANNE, A study of TD-CDMA and WCDMA radio network enhancements. 144 p. (230 p.) Yhteenveto 1 p. 2005.
- 56 PECHENIZKIY, MYKOLA, Feature extraction for supervised learning in knowledge discovery systems. 86 p. (174 p.) Yhteenveto 2 p. 2005.
- 57 IKONEN, SAMULI, Efficient numerical methods for pricing American options. 43 p. (155 p.) Yhteenveto 1 p. 2005.
- 58 KÄRKKÄINEN, KARI, Shape sensitivity analysis for numerical solution of free boundary problems. 83 p. (119 p.) Yhteenveto 1 p. 2005.
- 59 HELFENSTEIN, SACHA, Transfer. Review, reconstruction, and resolution. 114 p. (206 p.) Yhteenveto 2 p. 2005.
- 60 NEVALA, KALEVI, Content-based design engineering thinking. In the search for approach. 64 p. (126 p.) Yhteenveto 1 p. 2005.
- 61 KATASONOV, ARTEM, Dependability aspects in the development and provision of location-based services. 157 p. Yhteenveto 1 p. 2006.
- 62 SARKKINEN, JARMO, Design as discourse: Representation, representational practice, and social practice. 86 p. (189 p.) Yhteenveto 1 p. 2006.
- 63 ÄYRÄMÖ, SAMI, Knowledge mining using robust clustering. 296 p. Yhteenveto 1 p. 2006.
- 64 IFINEDO, PRINCELY EMILI, Enterprise resource planning systems success assessment: An integrative framework. 133 p. (366 p.) Yhteenveto 3 p. 2006.
- 65 VIINIKAINEN, ARI, Quality of service and pricing in future multiple service class networks. 61 p. (196 p.) Yhteenveto 1 p. 2006.
- 66 WU, RUI, Methods for space-time parameter estimation in DS-CDMA arrays. 73 p. (121 p.) 2006.
- 67 PARKKOLA, HANNA, Designing ICT for mothers. User psychological approach. - Tieto- ja viestintätekniikoiden suunnittelu äideille. Käyttäjäpsykologinen näkökulma. 77 p. (173 p.) Yhteenveto 3 p. 2006.
- 68 HAKANEN, JUSSI, On potential of interactive multiobjective optimization in chemical process design. 75 p. (160 p.) Yhteenveto 2 p. 2006.
- 69 PUUTONEN, JANI, Mobility management in wireless networks. 112 p. (215 p.) Yhteenveto 1 p. 2006.
- 70 LUOSTARINEN, KARI, Resource , management methods for QoS supported networks. 60 p. (131 p.) 2006.
- 71 TURCHYN, PAVLO, Adaptive meshes in computer graphics and model-based simulation. 27 p. (79 p.) Yhteenveto 1 p.
- 72 ZHOVTBRYUKH, DMYTRO, Context-aware web service composition. 290 p. Yhteenveto 2 p. 2006.

- 73 KOHVAKKO, NATALIYA, Context modeling and utilization in heterogeneous networks. 154 p. Yhteenveto 1 p. 2006.
- 74 MAZHELIS, OLEKSIY, Masquerader detection in mobile context based on behaviour and environment monitoring. 74 p. (179 p.) Yhteenveto 1 p. 2007.
- 75 SILTANEN, JARMO, Quality of service and dynamic scheduling for traffic engineering in next generation networks. 88 p. (155 p.) 2007.
- 76 KUUVVA, SARI, Content-based approach to experiencing visual art. - Sisältöperustainen lähestymistapa visuaalisen taiteen kokemiseen. 203 p. Yhteenveto 3 p. 2007.
- 77 RUOHONEN, TONI, Improving the operation of an emergency department by using a simulation model. 164 p. 2007.
- 78 NAUMENKO, ANTON, Semantics-based access control in business networks. 72 p. (215 p.) Yhteenveto 1 p. 2007.
- 79 WAHLSTEDT, ARI, Stakeholders' conceptions of learning in learning management systems development. - Osallistujien käsitykset oppimisesta oppimisympäristöjen kehittämässä. 83 p. (130 p.) Yhteenveto 1 p. 2007.
- 80 ALANEN, OLLI, Quality of service for triple play services in heterogeneous networks. 88 p. (180 p.) Yhteenveto 1 p. 2007.
- 81 NERI, FERRANTE, Fitness diversity adaptation in memetic algorithms. 80 p. (185 p.) Yhteenveto 1 p. 2007.
- 82 KURHINEN, JANI, Information delivery in mobile peer-to-peer networks. 46 p. (106 p.) Yhteenveto 1 p. 2007.
- 83 KILPELÄINEN, TURO, Genre and ontology based business information architecture framework (GOBIAF). 74 p. (153 p.) Yhteenveto 1 p. 2007.
- 84 YEVSJEVA, IRYNA, Solving classification problems with multicriteria decision aiding approaches. 182 p. Yhteenveto 1 p. 2007.
- 85 KANNISTO, ISTO, Optimized pricing, QoS and segmentation of managed ICT services. 45 p. (111 p.) Yhteenveto 1 p. 2007.
- 86 GORSHKOVA, ELENA, A posteriori error estimates and adaptive methods for incompressible viscous flow problems. 72 p. (129 p.) Yhteenveto 1 p. 2007.
- 87 LEGRAND, STEVE, Use of background real-world knowledge in ontologies for word sense disambiguation in the semantic web. 73 p. (144 p.) Yhteenveto 1 p. 2008.
- 88 HÄMÄLÄINEN, NIINA, Evaluation and measurement in enterprise and software architecture management. - Arviointi ja mittaaminen kokonais- ja ohjelmistoarkkitehtuurin hallinnassa. 91 p. (175 p.) Yhteenveto 1 p. 2008.
- 89 OJALA, ARTO, Internationalization of software firms: Finnish small and medium-sized software firms in Japan. 57 p. (180 p.) Yhteenveto 2 p. 2008.
- 90 LAITILA, ERKKI, Symbolic Analysis and Atomistic Model as a Basis for a Program Comprehension Methodology. 321 p. Yhteenveto 3 p. 2008.
- 91 NIHTILÄ, TIMO, Performance of Advanced Transmission and Reception Algorithms for High Speed Downlink Packet Access. 93 p. (186 p.) Yhteenveto 1 p. 2008.
- 92 SETÄMAA-KÄRKKÄINEN, ANNE, Network connection selection-solving a new multiobjective optimization problem. 52 p. (111p.) Yhteenveto 1 p. 2008.
- 93 PULKKINEN, MIRJA, Enterprise architecture as a collaboration tool. Discursive process for enterprise architecture management, planning and development. 130 p. (215 p.) Yhteenveto 2 p. 2008.
- 94 PAVLOVA, YULIA, Multistage coalition formation game of a self-enforcing international environmental agreement. 127 p. Yhteenveto 1 p. 2008.
- 95 NOUSIAINEN, TUULA, Children's involvement in the design of game-based learning environments. 297 p. Yhteenveto 2 p. 2008.
- 96 KUZNETSOV, NIKOLAY V., Stability and oscillations of dynamical systems. Theory and applications. 116 p. Yhteenveto 1 p. 2008.
- 97 KHRIYENKO, OLEKSIY, Adaptive semantic Web based environment for web resources. 193 p. Yhteenveto 1 p. 2008.
- 98 TIRRONEN, VILLE, Global optimization using memetic differential evolution with applications to low level machine vision. 98 p. (248 p.) Yhteenveto 1 p. 2008.
- 99 VALKONEN, TUOMO, Diff-convex combinations of Euclidean distances: A search for optima. 148 p. Yhteenveto 1 p. 2008.
- 100 SARAFANOV, OLEG, Asymptotic theory of resonant tunneling in quantum waveguides of variable cross-section. 69 p. Yhteenveto 1 p. 2008.
- 101 POZHARSKIY, ALEXEY, On the electron and phonon transport in locally periodical waveguides. 81 p. Yhteenveto 1 p. 2008.
- 102 AITTOKOSKI, TIMO, On challenges of simulation-based globaland multiobjective optimization. 80 p. (204 p.) Yhteenveto 1 p. 2009.
- 103 YALAHO, ANICET, Managing offshore outsourcing of software development using the ICT-supported unified process model: A cross-case analysis. 91 p. (307 p.) Yhteenveto 4 p. 2009.
- 104 KOLLANUS, SAMI, Tarkastuskäytänteiden kehittäminen ohjelmistoja tuottavissa organisaatioissa. - Improvement of inspection practices in software organizations. 179 p. Summary 4 p. 2009.
- 105 LEIKAS, JAANA, Life-Based Design. 'Form of life' as a foundation for ICT design for older adults. - Elämälähtöinen suunnittelu. Elämänmuoto ikääntyville tarkoitettujen ICT tuotteiden ja palvelujen suunnittelun lähtökohtana. 218 p. (318 p.) Yhteenveto 4 p. 2009.



- 106 VASILYEVA, EKATERINA, Tailoring of feedback in web-based learning systems: Certitude-based assessment with online multiple choice questions. 124 p. (184 p.) Yhteenveto 2 p. 2009.
- 107 KUDRYASHOVA, ELENA V., Cycles in continuous and discrete dynamical systems. Computations, computer assisted proofs, and computer experiments. 79 p. (152 p.) Yhteenveto 1 p. 2009.
- 108 BLACKLEDGE, JONATHAN, Electromagnetic scattering and inverse scattering solutions for the analysis and processing of digital signals and images. 297 p. Yhteenveto 1 p. 2009.
- 109 IVANNIKOV, ANDRIY, Extraction of event-related potentials from electroencephalography data. - Herätepotentiaalien laskennallinen eristäminen EEG-havaintoaineistosta. 108 p. (150 p.) Yhteenveto 1 p. 2009.
- 110 KALYAKIN, IGOR, Extraction of mismatch negativity from electroencephalography data. - Poikkeavuusnegatiivisuuden erottaminen EEG-signaalista. 47 p. (156 p.) Yhteenveto 1 p. 2010.
- 111 HEIKKILÄ, MARIKKA, Coordination of complex operations over organisational boundaries. 265 p. Yhteenveto 3 p. 2010.
- 112 FEKETE, GÁBOR, Network interface management in mobile and multihomed nodes. 94 p. (175 p.) Yhteenveto 1 p. 2010.
- 113 KUJALA, TUOMO, Capacity, workload and mental contents - Exploring the foundations of driver distraction. 146 p. (253 p.) Yhteenveto 2 p. 2010.
- 114 LUGANO, GIUSEPPE, Digital community design - Exploring the role of mobile social software in the process of digital convergence. 253 p. (316 p.) Yhteenveto 4 p. 2010.
- 115 KAMPYLIS, PANAGIOTIS, Fostering creative thinking. The role of primary teachers. - Luovaa ajattelua kehittämässä. Alakoulun opettajien rooli. 136 p. (268 p.) Yhteenveto 2 p. 2010.
- 116 TOIVANEN, JUKKA, Shape optimization utilizing consistent sensitivities. - Muodon optimointi käyttäen konsistentteja herkkyyksiä. 55 p. (130 p.) Yhteenveto 1 p. 2010.
- 117 MATTILA, KEIJO, Implementation techniques for the lattice Boltzmann method. - Virtausdynamiiikan tietokonesimulaatioita Hila-Boltzmann -menetelmällä: implementointi ja reunaehdot. 177 p. (233 p.) Yhteenveto 1 p. 2010.
- 118 CONG, FENGYU, Evaluation and extraction of mismatch negativity through exploiting temporal, spectral, time-frequency, and spatial features. - Poikkeavuusnegatiivisuuden (MMN) erottaminen aivosähkönauhotuksista käyttäen ajallisia, spektraalisia, aika-tila- ja tilapiirteitä. 57 p. (173 p.) Yhteenveto 1 p. 2010.
- 119 LIU, SHENGHUA, Interacting with intelligent agents. Key issues in agent-based decision support system design. 90 p. (143 p.) Yhteenveto 2 p. 2010.
- 120 AIRAKSINEN, TUOMAS, Numerical methods for acoustics and noise control. - Laskennallisia menetelmiä akustisiin ongelmiin ja melunvaimennukseen. 58 p. (133 p.) Yhteenveto 2 p. 2010.
- 121 WEBER, MATTHIEU, Parallel global optimization Structuring populations in differential evolution. - Rinnakkainen globaali optimointi. Populaation rakenteen määrittäminen differentiaalievoluutiossa. 70 p. (185 p.) Yhteenveto 2 p. 2010.
- 122 VÄÄRÄMÄKI, TAPIO, Next generation networks, mobility management and appliances in intelligent transport systems. - Seuraavan sukupolven tietoverkot, liikkuvuuden hallinta ja sovellutukset älykkäässä liikenteessä. 50 p. (111 p.) Yhteenveto 1 p. 2010.
- 123 VIUKARI, LEENA, Tieto- ja viestintätekniikkavälitteisen palvelun kehittämisen kolme diskurssia. - Three discourses for an ICT-service development. 304 p. Summary 5 p. 2010.
- 124 PUURTINEN, TUOMAS, Numerical simulation of low temperature thermal conductance of corrugated nanofibers. - Poimutettujen nanokuitujen lämmönjohtavuuden numeerinen simulointi matalissa lämpötiloissa. 114 p. Yhteenveto 1 p. 2010.
- 125 HILTUNEN, LEENA, Enhancing web course design using action research. - Verkko-opetuksen suunnittelun kehittäminen toimintatutkimuksen keinoin. 192 p. Yhteenveto 2 p. 2010.
- 126 AHO, KARI, Enhancing system level performance of third generation cellular networks through VoIP and MBMS services. 121 p. (221 p.) Yhteenveto 2 p. 2010.
- 127 HÄKKINEN, MARKKU, Why alarms fail. A cognitive explanatory model. 102 p. (210 p.) Yhteenveto 1 p. 2010.
- 128 PENNANEN, ANSSI, A graph-based multigrid with applications. - Graafipohjainen monihilamenetelmä sovelluksineen. 52 p. (128 p.) Yhteenveto 2 p. 2010.
- 129 AHLGREN, RIIKKA, Software patterns, organizational learning and software process improvement. 70 p. (137 p.) Yhteenveto 1 p. 2011.
- 130 NIKITIN, SERGIY, Dynamic aspects of industrial middleware architectures 52 p. (114 p.) Yhteenveto 1 p. 2011.
- 131 SINDHYA, KARTHIK, Hybrid Evolutionary Multi-Objective Optimization with Enhanced Convergence and Diversity. 64 p. (160 p.) Yhteenveto 1 p. 2011.

- 132 MALI, OLLI, Analysis of errors caused by incomplete knowledge of material data in mathematical models of elastic media. 111 p. Yhteenveto 2 p. 2011.
- 133 MÖNKÖLÄ, SANNA, Numerical Simulation of Fluid-Structure Interaction Between Acoustic and Elastic Waves. 136 p. Yhteenveto 2 p. 2011.
- 134 PURANEN, TUUKKA, Metaheuristics Meet Metamodels. A Modeling Language and a Product Line Architecture for Route Optimization Systems. 270 p. Yhteenveto 1 p. 2011.
- 135 MÄKELÄ, JUKKA, Mobility Management in Heterogeneous IP-networks. 86 p. (145 p.) Yhteenveto 1 p. 2011.
- 136 SAVOLAINEN, PAULA, Why do software development projects fail? Emphasising the supplier's perspective and the project start-up. 81 p. (167 p.) Yhteenveto 2 p. 2011.
- 137 KUZNETSOVA, OLGA, Lyapunov quantities and limit cycles in two-dimensional dynamical systems: analytical methods, symbolic computation and visualization. 80 p. (121 p.) Yhteenveto 1 p. 2011.
- 138 KOZLOV, DENIS, The quality of open source software and its relation to the maintenance process. 125 p. (202 p.) Yhteenveto 1 p. 2011.
- 139 IACCA, GIOVANNI, Memory-saving optimization algorithms for systems with limited hardware. 100 p. (236 p.) Yhteenveto 1 p. 2011.
- 140 ISOMÖTTÖNEN, VILLE, Theorizing a one-semester real customer student software project course. 189 p. Yhteenveto 1 p. 2011.
- 141 HARTIKAINEN, MARKUS, Approximation through interpolation in nonconvex multiobjective optimization. 74 p. (164 p.) Yhteenveto 1 p. 2011.