Report from Dagstuhl Seminar 20031

Scalability in Multiobjective Optimization

Edited by

Carlos M. Fonseca[1], Kathrin Klamroth[2], Günter Rudolph[3], and Margaret M. Wiecek[4]

1 **University of Coimbra, PT,** cmfonsec@dei.uc.pt
2 **Universität Wuppertal, DE,** klamroth@math.uni-wuppertal.de
3 **TU Dortmund, DE,** guenter.rudolph@tu-dortmund.de
4 **Clemson University, US,** wmalgor@clemson.edu

Abstract

The Dagstuhl Seminar 20031 Scalability in Multiobjective Optimization carried on a series of six previous Dagstuhl Seminars (04461, 06501, 09041, 12041, 15031 and 18031) that were focused on Multiobjective Optimization. The continuing goal of this series is to strengthen the links between the Evolutionary Multiobjective Optimization (EMO) and the Multiple Criteria Decision Making (MCDM) communities, two of the largest communities concerned with multiobjective optimization today.

This report documents the program and the outcomes of Dagstuhl Seminar 20031 "Scalability in Multiobjective Optimization". The seminar focused on three main aspects of scalability in multiobjective optimization (MO) and their interplay, namely (1) MO with many objective functions, (2) MO with many decision makers, and (3) MO with many variables and large amounts of data.

Seminar January 12–17, 2020 – http://www.dagstuhl.de/20031
2012 ACM Subject Classification Mathematics of computing → Mathematical optimization, Applied computing → Operations research, Computing methodologies → Artificial intelligence
Keywords and phrases multiple criteria decision making, evolutionary multiobjective optimization, scalability
Digital Object Identifier 10.4230/DagRep.10.1.52
Edited in cooperation with Allmendinger, Richard

1 Executive Summary

Carlos M. Fonseca (University of Coimbra, PT)
Kathrin Klamroth (Universität Wuppertal, DE)
Günter Rudolph (TU Dortmund, DE)
Margaret M. Wiecek (Clemson University, US)

To continue being useful to society, MO has to address new challenges brought to science and engineering by big data that are continuously being produced and stored with a much lower cost than ever in the past. Since massive production of data takes place in the areas of human activity that have traditionally benefited from MCDM (e.g., social media analysis, retail sales, or high-frequency finance), MO needs to enter a new stage of development to be able to handle the high-dimensional data. Driven by this increasing availability of data and

also motivated by an unprecedented demand for efficient, reliable and robust optimization methods, research in MO has to focus on the particular difficulties arising in large-scale problems. This requires from MCDM researchers new statistical thinking and leads to an increasing demand for efficient solution methods for large-scale problems, involving *many* objective functions and/or constraints, *many* decision makers, *many* variables and large amounts of data.

In this spirit, the focus in the seminar was on *scalability* which has become a universal challenge for mathematical optimization, and for EMO *and* MCDM in particular. *Scalability* is a characteristic of a system that describes its capability to cope and perform under an increased or expanding workload. A system that scales well will be able to maintain or even increase its level of performance or efficiency when tested by larger operational demands. In an economic context, a company's scalability implies that the underlying business model offers the potential for economic growth within the company. Therefore the main goals of the seminar were the exploration and elucidation of scalability in three fundamental domains: MO with many objective functions, MO with many decision makers, and MO with many variables.

While single objective optimization problems possess (at most) one optimal objective value, biobjective optimization problems are already intractable in many cases, i.e., combinatorial problems such as, for example, shortest path and spanning tree problems, may have an exponential number of nondominated solutions. Going from two to three objectives is another major step in difficulty since there does no longer exist a complete ordering of nondominated solutions. Problems with *many objective functions* pose even greater challenges. Since the number of nondominated solutions generally grows exponentially with the number of objective functions (as long as these are conflicting), efficient strategies for the detection of redundancies, for model reduction and for metamodelling are crucial for the scalability of existing methods. In the domain of MO with many objective functions the following specific topics were addressed:

- *Model building* and the derivation of technical properties are crucial for understanding the specific challenges in many-objective optimization. The following topics were undertaken: (i) Identification of interdependencies between objective functions as compared to real conflict; (ii) Relevance of many objective functions to a given real-life decision-making situation; (iii) Exploration of mathematical or statistical tools that can compress information while retaining the important problem features. Methodological approaches in this context included data analysis, metamodelling, partial and full scalarization, and a new concept for approximation schemes with quality guarantees.
- Concise *representations* are indispensable for the development of efficient algorithms, particularly EMO algorithms, interactive approaches, and decision support tools. The scalability of quality measures and associated representations, including hypervolume, uniformity, coverage, and ε-indicator were discussed and novel representation and visualization paradigms suitable for many-objective optimization were proposed.
- *Efficient solution algorithms* that scale well with an increasing number of objective functions or computationally expensive objective functions are needed. The shortcomings of existing methods were discussed and new strategies that are specifically designed for large-scale problems were derived.
- *Scalable test cases* are needed for the evaluation of the developed approaches. This has been a concern of the EMO community to some extent. The difficulties pertaining to construction of the test cases were identified and future work in this direction was proposed.

The discussion of MO with many decision makers considered the inherent changes in the decision process as soon as there is not just a single decision maker. The focus was on building a formal framework that guarantees a fair decision respecting the preferences of all decision makers with the least total loss.

The domain of MO with many variables was discussed jointly with the domain of MO with many objective functions because large-scale optimization problems involving many variables and large amounts of data often also involve many objective functions. However, an emphasis was put on the required adaptations of EMO and MCDM approaches to handle problems with a high-dimensional decision space. While EMO algorithms often scale relatively well with an increasing dimension of the decision space (at least as long as the number of objective functions remains relatively small), this is in general not the case for MCDM approaches. In particular, the most commonly used exact solution approaches, such as dynamic programming and branch and bound, suffer from the curse of dimensionality. Complexity theoretic aspects were discussed and the use of approximation paradigms, metamodelling, hybridization, or parallelization in this situation was investigated.

During the seminar the schedule was updated on a daily basis to maintain flexibility in balancing time slots for talks, discussions, and working groups, and to integrate in the program the talks whose authors were inspired by the ongoing seminar. The working groups were established on the first day in an interactive fashion. Starting with three large working groups focused around the three central topics of the seminar (MO with many objectives, MO with many decision makers, and MO with many variables), each participant was requested to formulate her/his favorite topics and most important challenges.

The three groups then rejoined and the prevailing topics were put into groups through a natural clustering process while the participants made up initial five working groups. During the week the participants were allowed to change the working groups while some groups split. Overall, the teams remained fairly stable throughout to eventually form eight groups by the end of the seminar. Abstracts of the talks and extended abstracts of the eight working groups can be found in subsequent chapters of this report.

Further notable events during the week included: (i) an invitation to the opening of the art exhibition with paintings of the artist Lola Sprenger, (ii) a hike during a time period with the best (!) weather conditions in the entire week, (iii) a presentation session allowing the participants to share details of upcoming events in the research community, and (iv) a wine and cheese party (see Fig. 1) made possible by a donation of *Fraunhofer-Institut für Techno- und Wirtschaftsmathematik (ITWM)* represented by Karl-Heinz Küfer. The participants are pleased to announce that they made a donation to Schloss Dagstuhl to make a painting by Lola Sprenger entitled “Berg und Tal” part of the permanent art display.

Offers and Needs Market

A major innovation to this seminar was the *Offers & Needs Market* open for the entire week. The participants could write their research offers and needs regarding MO on notepads in different colors and post on pin boards (see fig. 2) to attract or find a possible collaborator. The idea was well received and the participants desired its repetition in future events.

Figure 1 Traditional wine & cheese party.

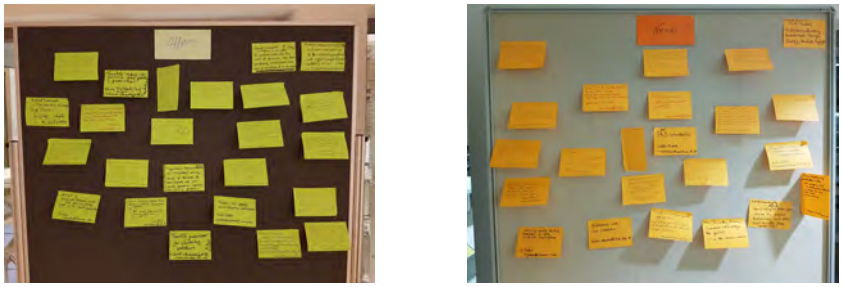


Figure 2 Offers and needs market.

Outcomes

The outcomes of each of the working groups can be seen in the sequel. Extended versions of their findings will be submitted to a Special Issue of *Computers and Operations Research* entitled "Modern Trends in Multiobjective Optimization" and guest-edited by the organizers of this Dagstuhl seminar.

This seminar resulted in a very insightful, productive and enjoyable week. It has already led to first new results and formed new cooperation, research teams and topics.

Acknowledgements

The organizers would like to express their appreciation to the Dagstuhl office and its helpful and patient staff for their professional and smooth cooperation; huge thanks to the organizers of the previous seminars in this series for setting us up for success; and thanks to all the participants, who worked hard and were amiable company all week. In a later section, we also give special thanks to Kathrin Klamroth and Günter Rudolph as they step down from the organizer role.

2 Table of Contents

3 Overview of Talks

3.1 Scaling up Multi-Objective Bayesian Optimization

Mickaël Binois (INRIA – Valbonne, FR)

Joint work of Mickaël Binois, Abderrahmane Habbal, Victor Picheny, Stefan M. Wild (Argonne), Nathan Wycoff

Bayesian optimization (BO) aims at efficiently optimizing expensive black-box functions, such as hyperparameter tuning problems in machine learning. Scaling up BO to many variables relies on structural assumptions about the underlying black-box, to alleviate the curse of dimensionality. In this talk, we review several options to tackle this challenge. We also discuss the use of the Kalai-Smorodinski solution when the number of objectives increases, for which a stepwise uncertainty reduction infill criterion is detailed.

References

1 Mickaël Binois, Victor Picheny, Patrick Taillandier, Abderrahmane Habbal. *The Kalai-Smorodinski solution for many-objective Bayesian optimization*. ArXiv preprint 1902.06565. 2019
2 Nathan Wycoff, Mickaël Binois, Stefan Wild, *Sequential Learning of Active Subspaces*. ArXiv preprint 1907.11572. 2019

3.2 Output-sensitive Complexity in Multiobjective Optimization

Fritz Bökler (Universität Osnabrück, DE)

Main reference Fritz Bökler: "Output-sensitive Complexity of Multiobjective Combinatorial Optimization Problems with an Application to the Multiobjective Shortest Path Problem", PhD Thesis, TU Dortmund University, 2018
URL http://dx.doi.org/10.17877/DE290R-19130

In this talk, I summarize my core findings on output-sensitive complexity of multiobjective optimization problems from the past years. I contrast the results with current open problems in the respective areas. I also present new open problems on counting and approximation of nondominated sets.

3.3 On Set-Indicator-Based Search: Using Single-Objective Solvers for Multiobjective Problems

Dimo Brockhoff (INRIA Saclay – Palaiseau, FR)

Joint work of Dimo Brockhoff, Cheikh Touré, Anne Auger, Nikolaus Hansen
Main reference Cheikh Touré, Nikolaus Hansen, Anne Auger, Dimo Brockhoff: "Uncrowded hypervolume improvement: COMO-CMA-ES and the sofomore framework", in Proc. of the Genetic and Evolutionary Computation Conference, GECCO 2019, Prague, Czech Republic, July 13-17, 2019, pp. 638–646, ACM, 2019.
URL https://doi.org/10.1145/3321707.3321852

One approach to solve multiobjective optimization problems is to formulate them as single-objective set problems via indicators: the goal is then to find the set of solutions (of a given size) that maximizes a certain quality. The hypervolume indicator has been regularly used in this context because it has favorable theoretical properties. The "classical" definition of the hypervolume indicator and how it is used in multiobjective solvers, however, has some disadvantages: (i) the resulting single-objective set problem is of high dimension and the gradient of the hypervolume indicator is zero in dominated areas of the search space – not giving a solver enough information about where to search for good solutions.

In this talk, I discussed and visualized these disadvantages and presented a set quality criterion which is based on the hypervolume indicator but solves the mentioned disadvantages (joint work with Cheikh Touré, Anne Auger, and Nikolaus Hansen). The implementation of this idea can be combined with any existing single-objective solver with an ask&tell interface, in particular solvers that can handle expensive or large-scale problems.

3.4 A Multiobjective Trust Region Method for Expensive and Cheap Functions

Gabriele Eichfelder (TU Ilmenau, DE)

Joint work of Gabriele Eichfelder, Jana Thomann
Main reference Jana Thomann, Gabriele Eichfelder: "A Trust-Region Algorithm for Heterogeneous Multiobjective Optimization", SIAM Journal on Optimization, Vol. 29(2), pp. 1017–1047, 2019.
URL https://doi.org/10.1137/18M1173277

The talk is about multiobjective optimization problems where one or more of the objective functions are expensive, i.e. computationally heavy, see [1, 3]. In case just some of the functions are expensive while the others are cheap we speak of heterogeneous problems [3]. Such problems occur in applications for instance in the context of Lorentz force velocimetry, when the task is to find an optimal design of a magnet which minimizes the weight of the magnet and maximizes the induced Lorentz force. The latter might be computable only by a time-consuming simulation. We discuss the reasons why classical methods as scalarizations, descent methods and so on are not a suitable approach here. We present the basic concepts of trust region approaches which are often used in case of expensive functions. The main idea is to restrict the computations in every iteration to a local area and to replace the objective functions by suitable models. The number of function evaluations also depends on the models chosen (linear vs. quadratic), and we give recommendations for the model choice based on numerical experiments [3]. Moreover, we discuss acceptation criteria for the iteration steps which are suitable for an application, the electromagnetic mixing in a liquid metal [4].

References

1 Jana Thomann and Gabriele Eichfelder. *A Trust-Region Algorithm for Heterogeneous Multiobjective Optimization*. SIAM Journal on Optimization, 29(2), 1017 – 1047, 2019.
2 Jana Thomann and Gabriele Eichfelder. *Numerical Results for the Multi-Objective Trust Region Algorithm MHT*. Data in Briefs, 25 , 2019.
3 Jana Thomann and Gabriele Eichfelder. *Representation of the Pareto front for heterogeneous multi-objective optimization*. Journal of Applied and Numerical Optimization, 1(3), 293-323, 2019.
4 Sebastian Prinz, Jana Thomann, Gabriele Eichfelder, Thomas Boeck, Jörg Schumacher. *Expensive multi-objective optimization of electromagnetic mixing in a liquid metal* . OptimizationOnline, 2019

3.5 Multi-Objective Multi-Scale Optimization with Massively Large Number of Variables: Design of Graded Components

Georges Fadel (Clemson University – Clemson, US)

Joint work of Georges Fadel, Anthony Garland

Additive Manufacturing processes that deposit material on a point-by-point basis are able to control the material type that is placed at every location within a single object, which enables customizing the object's internal material composition on a region-by-region or cell-by-cell basis. Furthermore, Additive Manufacturing can create complex shapes that cannot be manufactured using other manufacturing techniques. The optimal object design should then seek to find both the optimal part topology and an optimal internal material composition. In particular, the design for many parts requires considering both thermal and elastic loading. This presentation describes how an object's optimal topology and optimal internal composition can be generated using multi-scale optimization, and how the emphasis on one objective versus the other affects the final solution. Since the design process and the associated finite element analyses must be carried out at the macro-cell level, the discretization of the object results in a number of variables in the order of thousands. Additionally, each macro-cell is again discretized into a large number of smaller micro level cells, again in the order of thousands each, and a multi-level coordination method (ATC) is required to obtain an optimal solution for each level yielding an optimal solution for the two scales. The optimization problem is solved as a weighted bi-objective problem at the macro level with the topological objective converted to a constraint. It is solved as a single objective problem at the micro-level using parallel computers. Example designed structures in 2D are shown.

References

1 Anthony Garland. *Optimal Design of Gradient Materials and Bi-Level Optimization of Topology Using Targets (BOTT)*. Ph.D. dissertation, Department of Mechanical Engineering, Clemson University, August 2017
2 Anthony Garland and Georges M. Fadel. *Optimizing Topology and Gradient Orthotropic Material Properties Under multiple Loads*. J. Comput. Inf. Sci. Eng. Jun 2019, 19(2): 021007

3.6 Multi-Objective Simulation Optimization: Theory and Practice

Susan R. Hunter (Purdue University, US)

Main reference Susan R. Hunter, Eric A. Applegate, Viplove Arora, Bryan Chong, Kyle Cooper, Oscar Rincón-Guevara, Carolina Vivas-Valencia: "An Introduction to Multiobjective Simulation Optimization", ACM Trans. Model. Comput. Simul., Vol. 29(1), Association for Computing Machinery, 2019.
URL https://doi.org/10.1145/3299872

The multi-objective simulation optimization (MOSO) problem is a nonlinear multi-objective optimization problem in which the objective functions can only be observed with stochastic error, e.g., through a Monte Carlo simulation oracle. To date, these difficult problems have seen little theoretical development in the literature. We discuss the MOSO problem statement, existing MOSO methods, and promising directions for the future development of MOSO theory and practice.

References

1 Susan R. Hunter, Eric A. Applegate, Viplove Arora, Bryan Chong, Kyle Cooper, Oscar Rincón-Guevara, and Carolina Vivas-Valencia. An introduction to multi-objective simulation optimization. *ACM Transactions on Modeling and Computer Simulation*, 29 (1): 7:1–7:36, January 2019. 10.1145/3299872.
2 Kyle Cooper, Susan R. Hunter, and Kalyani Nagaraj. Bi-objective simulation optimization on integer lattices using the epsilon-constraint method in a retrospective approximation framework. *INFORMS Journal on Computing*, 2020. 10.1287/ijoc.2019.0918.
3 Kyle Cooper and Susan R. Hunter. PyMOSO: Software for multi-objective simulation optimization with R-PERLE and R-MinRLE. *INFORMS Journal on Computing*, 2020. 10.1287/ijoc.2019.0902.

3.7 Chances and Challenges of Multimodality in Multi-Objective Continuous Optimization Problems

Pascal Kerschke (Universität Münster, DE)

Joint work of Pascal Kerschke, Christian Grimme, Heike Trautmann, Michael Emmerich, Hao Wang, Andre Deutz, Mike Preuss
Main reference Christian Grimme, Pascal Kerschke, Heike Trautmann: "Multimodality in Multi-objective Optimization – More Boon than Bane?", in Proc. of the Evolutionary Multi-Criterion Optimization, pp. 126–138, Springer International Publishing, 2019.
URL https://doi.org/10.1007/978-3-030-12598-1_11

As the quality of multi-objective (MO) optimization algorithms needs to be assessed, they are usually tested on a variety of MO benchmark problems (DTLZ, ZDT, bi-objective BBOB, etc.). Those problems have usually been designed with some expectations regarding their difficulty for a MO algorithm, which is usually inferred from our understanding of structures in the single-objective domain. As such, structural properties like multimodality are assumed to be rather challenging – especially for local search algorithms. However, recent developments on the visualization of MO test problems revealed interesting insights, which question those assumptions [3, 2]. Therefore – in order to get away from fine-tuning algorithms and/or designing problems without actually understanding the challenges (and chances) of MO

optimization – we should look for new approaches that actually improve our understanding of such problems [1]. Some possible ideas would be:

(1) How can we visualize more than 2-3 search and/or objective variables (ideally simultaneously)?

(2) How to scale MO benchmarks w.r.t. structural properties (e.g., how does multimodality change with increasing dimensionality in search and/or objective space)?

(3) What would be (scalable) features to characterize continuous MO problems? (ideally those features should of course be informative and efficiently computable in high-dimensional spaces)

References

1 Christian Grimme and Pascal Kerschke and Heike Trautmann. *Multimodality in Multi-Objective Optimization–More Boon than Bane?*. International Conference on Evolutionary Multi-Criterion Optimization, pp. 126–138, East Lansing, MI, USA, Springer, 2019.

2 Pascal Kerschke and Hao Wang and Mike Preuss and Christian Grimme and André H. Deutz and Heike Trautmann and Michael T. M. Emmerich. *Search Dynamics on Multimodal Multiobjective Problems*. Evolutionary Computation, 27(4), pp. 577–609, MIT Press, 2019.

3 Pascal Kerschke and Christian Grimme. *An Expedition to Multimodal Multi-Objective Optimization Landscapes*. International Conference on Evolutionary Multi-Criterion Optimization, pp. 329–343, Münster, Germany, Springer.

3.8 On the Difficulty of Multiobjective Combinatorial Optimization Problems

Arnaud Liefooghe (University of Lille, FR)

Joint work of Arnaud Liefooghe, Fabio Daolio, Sébastien Verel, Bilel Derbel, Hernán Aguirre, Kiyoshi Tanaka, Manuel López-Ibánez, Luís Paquete

We first analyze in [1] the impact of the number of variables (n) and of the number of objectives (m) on the difficulty of multiobjective combinatorial optimization problems. Based on extensive experiments conducted on multiobjective NK landscapes, a general family of multiobjective pseudo-boolean functions, we measure the Pareto set approximation quality reached by multiobjective search algorithms with respect to n and m. Additionally, we discuss the relative importance of n and m compared against other facets of problem difficulty. Based on landscape analysis, a sound and concise summary of features characterizing the structure of a multiobjective combinatorial optimization problem are identified, including objective correlation, ruggedness and multimodality. We then expose and contrast the relation between these properties and algorithm performance, thus enhancing our understanding about why and when a multiobjective optimization algorithm is actually successful, and about the main challenges that such methods have to face.

Secondly, we report in [2] an in-depth experimental analysis on local optimal set (LO-set) under given definitions of neighborhood and preference relation among subsets of solutions, such as set-based dominance relation, hypervolume or epsilon indicator. Our results reveal that, whatever the preference relation, the number of LO-sets typically increases with the problem non-linearity, and decreases with the number of objectives. We observe that strict LO-sets of bounded cardinality under set-dominance are LO-sets under both epsilon

and hypervolume, and that LO-sets under hyper-volume are LO-sets under set-dominance, whereas LO-sets under epsilon are not. Nonetheless, LO-sets under set-dominance are more similar to LO-sets under epsilon than under hypervolume. These findings have important implications for multi-objective local search. For instance, a dominance-based approach with bounded archive gets more easily trapped and might experience difficulty to identify an LO-set under epsilon or hypervolume. On the contrary, a hypervolume-based approach is expected to perform more steps before converging to better approximations.

References

1 Arnaud Liefooghe, Fabio Daolio, Sébastien Verel, Bilel Derbel, Hernán Aguirre, Kiyoshi Tanaka. *Landscape-aware performance prediction for evolutionary multi-objective optimization*. IEEE Transactions on Evolutionary Computation, 2019 (early access)

2 Arnaud Liefooghe, Manuel López-Ibánez, Luís Paquete, Sébastien Verel. *Dominance, epsilon, and hypervolume local optimal sets in multi-objective optimization, and how to tell the difference*. Genetic and Evolutionary Computation Conference (GECCO 2018), pp 324-331, Kyoto, Japan, 2018

3.9 Robust Multiobjective Optimization Problems and an Approach for Solving them

Anita Schöbel (Fraunhofer ITWM – Kaiserslautern, DE)

Joint work of Marco Botte, Anita Schöbel
Main reference Marco Botte, Anita Schöbel: "Dominance for multi-objective robust optimization concepts", Eur. J. Oper. Res., Vol. 273(2), pp. 430–440, 2019.
URL https://doi.org/10.1016/j.ejor.2018.08.020

Multiobjective optimization problems often face uncertainties. Instead of efficient solutions there is hence a need for robust efficient solutions in many practical settings. Concepts for defining such robust efficient solutions exist, but unfortunately, they are usually hard to find.

One promising algorithmic idea is to reduce an uncertain multi-objective optimization problem to a deterministic multiobjective optimization problem with very many objective functions. The approach has been studied for single-objective optimization before (among others in Klamroth et al, 2017) and is motivated by an idea of Wiecek et al (2009). The talk is based on the paper by Botte and Schöbel (2019) mentioned above.

The talk shows in particular the need for solving multi-objective problems with very many objective functions.

4 Working groups

4.1 Many Objectives: Characterization and Structure (WG2)

Richard Allmendinger (University of Manchester, GB), Andrzej Jaszkiewicz (Poznan University of Technology, PL), Arnaud Liefooghe (University of Lille, FR), and Christiane Tammer (Martin-Luther-Universität Halle-Wittenberg, DE)

4.1.1 Topics covered

This working group has covered a number of topics related to impact that many objectives (in an optimization problem) have on algorithm design and theoretical properties of tools. The availability of suitable many-objective optimization test problems has been touched upon as well. What follows is a summary of topics covered.

Influence of the number of objectives on problem characteristics
- Types of characteristics
- Theoretical results
- Existing experimental results

Considerations on the effect of the number of objectives on the complexity of multiobjective procedures and algorithms
- Updating the archive of non-dominated solutions
- Solving scalarizing problem(s)
- Computing and approximating hypervolume
- EMO algorithms

Exemplary problems
- Location problems [11, 3]
- Distance problems [18, 19, 9]
- ρMNK-landscapes [27]
- Instance generators

4.1.2 Effect of the Number of Objectives on Problem Characteristics

In this section, we study the influence of the number of objectives on different problem characteristics such as the number of Pareto optimal solutions or of preference parameters.

Number of Pareto Optimal Solutions (Combinatorial Case)

In the combinatorial case, the number of Pareto optimal solutions grows exponentially with the number of objectives in the worst case, that is $O(c^{m-1})$, where c is a constant [5]. Furthermore, as shown in [5], this bound is tight for many classical multiobjective combinatorial optimization problems, such as selection, knapsack, shortest path, spanning tree, traveling salesperson, and s–t cut problems. Obviously, the number of Pareto optimal solutions is also bounded by the size of the whole feasible set.

In Figure 3, we report the proportion of Pareto optimal solutions in the solution space with respect to the number of objectives (from 2 to 20 objectives) for ρMNK-landscapes [27] (see Section 4.1.4). Let us focus on independent objectives ($\rho = 0$). We see that less than 5%

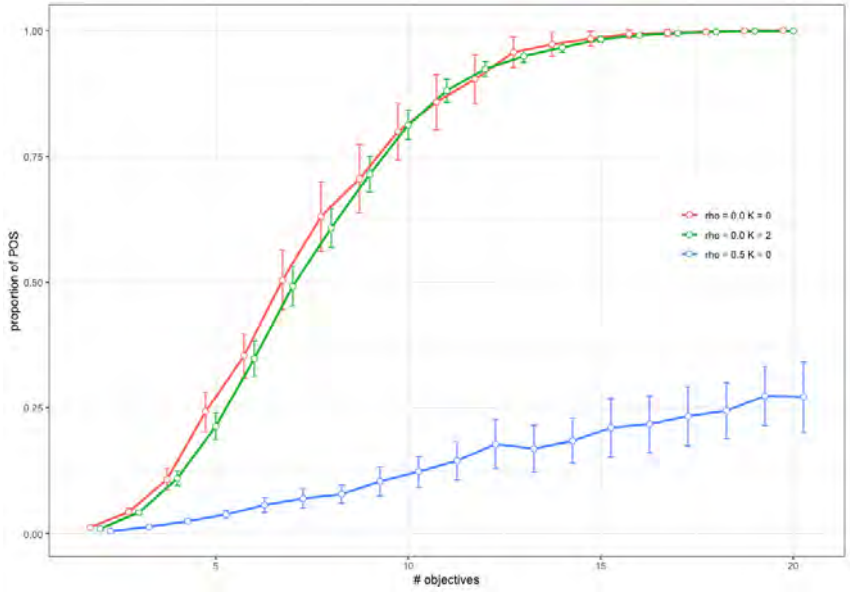


Figure 3 Proportional number of Pareto optimal solutions with respect to the number of objectives for ρMNK-landscapes with different degree of non-linearity (k) and correlations among the objectives (ρ).

of solutions are Pareto optimal for 2-objective problems ($m = 2$), whereas this proportion grows to about 50% for $m = 7$ objectives. For $m = 20$ objectives, more than 99% are Pareto optimal solutions.

Discriminative Power of the Dominance Relation

With a growing number of objectives, the dominance relation becomes less discriminative. Let us consider the comparison between two arbitrary solutions x^1 and x^2 on m objectives. Assume that the probability of equal objective values can be neglected, and that the comparison with respect to each objective is independent. For each objective there is a $1/2$ probability that x^1 has a better value for this objective, and a $1/2$ probability of the opposite situation. As such, the probability that one solution dominates the other one is $1/2^{(m-1)}$. Thus, it becomes more and more likely that two arbitrary solutions are mutually non-dominated. If the objectives are positively correlated, this probability increases, and if they are negatively correlated this probability decreases.

Probability for a Solution to be Non-Dominated in a Population

As a consequence of the reduced discriminative power of the dominance relation, the probability that a given solution is Pareto optimal increases. Consider a population of μ random solutions. The probability that a given solution is not dominated by another one is

$$1 - \frac{1}{2^m}$$

and thus the probability that one of them is not dominated by any other solution in the population is

$$\prod_{i=1}^{\mu-1}\left(1-\frac{1}{2^m}\right)=\left(1-\frac{1}{2^m}\right)^{\mu-1},$$

and the expected number of non-dominated solutions in this population is then

$$\mu\left(1-\frac{1}{2^m}\right)^{\mu-1}.$$

Dimensionality of the Objective Space

In the continuous case, the size of the set of Pareto optimal solutions formally does not grow, because it is infinite, $\mathfrak{c}$ to be precise, already for two objectives. However, the dimensionality of the objective space and the Pareto front grows. This means that more points or directions are typically required to approximately cover the Pareto front or possible search directions, respectively.

Number of Preference Parameters

In many multiobjective optimization methods, the decision maker (DM) is expected to express his/her preferences e.g. in the form of weighting coefficients or reference levels (aspiration levels/goals) specified for each objective. The number of such parameters grows just linearly with the number of objectives. In the case of some methods like AHP [26], preference parameters are expressed with respect to each pair of objectives; their number then grows quadratically.

Probability of Having Heterogeneous Objectives

By heterogeneous objectives, we here mean objectives that differ for example in their mathematical form (e.g. linear vs. non-linear), cost and/or time of evaluation (e.g. analytical form vs. simulation vs. real physical experiment). Intuitively, the higher the number of objectives, the higher the chance that some of them will differ from other objectives; i.e. that some of them will be more multimodal or costly/time-consuming in evaluation, an issue that has been investigated e.g. in [2, 6].

4.1.3 Effect of the Number of Objectives on the Complexity of Multiobjective Procedures and Algorithms

Although most studies are based on a number of pairwise comparisons of solutions, it is important to notice that the elementary operation for complexity results reported below is a pairwise comparison *per objective*. This choice is motivated by the fact that we want to highlight the effect of the number of objectives (m) on different multiobjective optimization tools and methods.

Updating the Pareto Archive

The Pareto archive is a structure used to store the set of points in the objective space (and corresponding solutions) generated by a multiobjective optimization method, in particular solutions being non-dominated with respect to all solutions generated so far. Updating the Pareto archive A with a new solution x means that all solutions dominated by x are removed from A and x is added to A if it is not dominated by any solution in A.

Updating the Pareto archive can be performed efficiently with the ND-Tree data structure [16]. The complexity of this process is as follows:

- Worst case: $O(m\ \mu)$;
- Best case: $\Theta(m\ \log(\mu))$;
- Average case: $\Theta(m\ \mu^b)$, where μ is the size of the archive, and $b \in [0, 1]$ is the probability of branching.

Note that sublinear time complexity in average case could also be obtained with another recently proposed data structure BSP Tree [12].

An interesting question is how b changes with respect to the number of objectives. In ND-Tree, the archive is recursively divided into subsets of points being close to each other in the objective space. For each subset, a local ideal point and a local nadir point are maintained. A new solution is compared first to the two points and only if the decision with respect to this subset cannot be made, then the corresponding branch of the tree is expanded. The latter situation occurs when the new solution is dominated by the local ideal point and/or when the new solution dominates the local nadir point. Given that, as discussed above, the probability that the dominance relation holds for two solutions (points) decreases with the number of objectives, these situations may become even less likely with a growing number of objectives.

The Pareto archive may be either bounded in size or unbounded, i.e. contain only some or all non-dominated solutions generated so far [10]. In the latter case, the size of the Pareto archive may, in general, grow exponentially with the number of objectives (see Section 4.1.2). Assuming that $\mu = O(c^{m-1})$, the complexity of the update process becomes:

- Worst case: $O(m\ \mu) = O(m\ c^{m-1})$;
- Best case: $\Theta(m\ \log(\mu)) = \Theta(m\ \log(c^{m-1})) = \Theta(m^2\ \log(c))$;
- Average case: $\Theta(m\ \mu^b) = \Theta(m\ c^{(m-1)b})$.

In other words, in the average case the time grows exponentially with the number of objectives, however, with a relatively low exponent, assuming that probability of branching is $b \ll 1$.

Dominance Test

In this section, we consider the process of testing if a solution x is non-dominated or dominated by a Pareto archive. The complexity analysis of this process with the use of ND-tree is the same as for updating the Pareto archive, since the dominance test is the bottleneck part of the updating process. The same holds for BSP Tree [12].

Solving Scalarizing Problem(s)

Let $c(n, m)$ be the complexity for solving one scalar problem. Assuming that the complexity grows linearly with m, then we have $c(n, m) = O(c(n)\ m)$. As such, when multiple scalar subproblems are to be solved, as in decomposition-based evolutionary multiobjective optimization (e.g., [25]), the complexity is $O(c(n)\ m\ \mu)$, with μ being the number of sub-problems. However, notice that it is often assumed that μ increases with the number objectives m in order obtain a good approximation of the Pareto set [25].

Computing and Approximating Hypervolume

When assessing the performance of multiobjective optimization algorithms, or in indicator-based evolutionary multiobjective optimization, the indicator-value of a solutions-set of size μ is to be computed multiple times. One of the recommended and most-often used indicator is the hypervolume [32]. Unfortunately, the exact hypervolume computation is known to grow exponentially with the number of objectives, more particularly: $O(\mu^{m-2}\ \log(\mu)\ m)$ [4].

Alternatively, the hypervolume can be approximated by Monte Carlo sampling [4]. In this case, the complexity is $\Theta(s\ m\ \mu^b)$, where s is the number of sampling points (see above). Since, Monte Carlo sampling is just a sequence of s independent experiments, each asking a yes/no question (dominated/non-dominated), the confidence interval can be derived from a binomial distribution and does not depend on the number of objectives. On the other hand, the question remains if the size of the confidence intervals should be reduced with growing number of objectives or growing μ. Furthermore, as discussed above, it is often assumed that μ increases with m.

EMO Algorithms

This section tries to understand the impact of many objectives on the working principles of different types of evolutionary multiobjective optimization (EMO) algorithms.

For *all* EMO algorithms that use a constant population size and no external archive:

- Algorithm performance should be evaluated in terms of representation quality.
- The distance between solutions in the objective space increases, so that the quality of representation likely decreases, with poorer coverage.
- The distance between solutions in the decision space increases, so that (blind) recombination likely becomes less effective.

For *dominance-based EMO algorithms* (e.g. NSGA-II [7]):

- The dominance relation becomes less discriminative (see Section 4.1.2). Because of this, we expect a lower selection pressure based on dominance, and then a lower quality in terms of Pareto front approximation.
- Since most individuals are mutually non-dominated, the EMO selection pressure is mostly guided by the diversity preservation mechanisms.
- Convergence is potentially affected.
- The complexity of non-dominated sorting is potentially affected.

For *decomposition/scalarization-based EMO algorithms* (e.g. MOEA/D [29]):

- Assuming a constant number of weight vectors (and population size), all issues mentioned above for constant population size hold. Moreover, the distance between weight vectors increases.
- Assuming the number of weight vectors increases with m (in order to maintain the same level of coverage), the algorithm complexity increases with the number of weight vectors. It remains unclear how the number of weight vectors shall increase, e.g. polynomially or exponentially.

For *indicator-based EMO algorithms* (e.g. IBEA [31]):

- For exact hypervolume computation, the complexity is exponential with m (see above).
- For Monte Carlo approximation, and assuming a constant population size, all issues mentioned above for constant population size hold. By contrast, assuming an population size that increases with m, it remains unclear at this stage how the number of sampling points shall be changed (or not) to reach the same level of hypervolume approximation quality.

4.1.4 Case Studies on Real and Artificial Problems

Multiobjective NK Landscapes (Artificial)

ρMNK-landscapes [27] are a problem-independent model used for constructing multiobjective multimodal combinatorial problems with objective correlation. They extend single-objective NK-landscapes [17] and multiobjective NK-landscapes with independent object-

ives [1]. Candidate solutions are binary strings of size n. The objective function vector $f = (f_1, \ldots, f_i, \ldots, f_m)$ is defined as $f\colon \{0,1\}^n \mapsto [0,1]^m$ such that each objective f_i is to be maximized. As in the single-objective case, the objective value $f_i(x)$ of a solution $x = (x_1, \ldots, x_j, \ldots, x_n)$ is an average value of the individual contributions associated with each variable x_j. Given objective f_i, $i \in 1, \ldots, m$, and each variable x_j, $j \in 1, \ldots, n$, a component function $f_{ij}\colon \{0,1\}^{k+1} \mapsto [0,1]$ assigns a real-valued contribution for every combination of x_j and its k *epistatic interactions* $x_{j_1}, \ldots, x_{j_k}$. These f_{ij}-values are uniformly distributed in $[0,1]$. Thus, the individual contribution of a variable x_j depends on its value and on the values of $k < n$ variables $x_{j_1}, \ldots, x_{j_k}$ other than x_j. The problem can be formalized as follows:

$$\begin{array}{lll} \max & f_i(x) = \dfrac{1}{n}\displaystyle\sum_{j=1}^{n} f_{ij}(x_j, x_{j_1}, \ldots, x_{j_k}) & i \in 1, \ldots, m \\ \text{s.t.} & x_j \in \{0,1\} & j \in 1, \ldots, n \end{array}$$

The epistatic interactions, i.e. the k variables that influence the contribution of x_j, are typically set uniformly at random among the $(n-1)$ variables other than x_j, following the random neighborhood model from [17]. By increasing the number of epistatic interactions k from 0 to $(n-1)$, problem instances can be gradually tuned from smooth to rugged. In ρMNK-landscapes, f_{ij}-values additionally follow a multivariate uniform distribution of dimension m, defined by an $m \times m$ positive-definite symmetric covariance matrix (c_{pq}) such that $c_{pp} = 1$ and $c_{pq} = \rho$ for all $p, q \in 1, \ldots, m$ with $p \neq q$, where $\rho > \frac{-1}{m-1}$ defines the correlation among the objectives. The positive (respectively, negative) objective correlation ρ decreases (respectively, increases) the degree of conflict between the different objective function values. Interestingly, ρMNK-landscapes exhibit different characteristics and different degrees of difficulty for multiobjective optimization methods [20]. The source code of the ρMNK-landscapes generator (and other problem classes) as well as a set of multiobjective combinatorial benchmark instances are available at the following URL: http://mocobench.sf.net.

Multiobjective Distance-based Problems (Artificial)

Distance-based optimization problems have been developed with the aim to visualize the movement of a population through the design space over time in order to understand, e.g. search bias. This is achieved by assuming a 2D design space with arbitrarily many (m) objective dimensions. More formally, in a standard visualisable distance-based test problem, the ith objective is calculated as

$$f_i(x) = \min_{v \in V_i}(\texttt{dist}(x, \mathbf{v})),$$

where $\mathcal{X}$ is a feasible design space, and $x \in \mathcal{X} \subseteq \mathbb{R}^2$ a 2D point (solution) in the design space. There are m sets of vectors defined, where the ith set, $V_i = \{\mathbf{v}_1, \ldots, \mathbf{v}_{s_i}\}$, determines the quality of a putative design vector $\mathbf{x} \in \mathcal{X}$, on the ith objective. Note as s_i is the number of elements of V_i, which depends on i, it is legal for $|V_p| \neq |V_q|$, but $|V_p| \geq 1$ for all p. The function $\texttt{dist}(x, \mathbf{v})$ typically returns the Euclidean distance between x and $\mathbf{v}$.

Figure 4 illustrates the simplest distance-based problem formulation using points, where $|V_i| = 1$ for all i. This means that there is a single connected Pareto set, and no additional locally Pareto optimal regions.

There are two approaches to setting the set the elements of V_i: (i) setting the elements of V_i directly by fixing $2 \times m$ parameters or (ii) using a centre (in a 2D space), a circle radius and an angle to each objective minimizing vector resulting in $3 + m$ parameters to fix. The

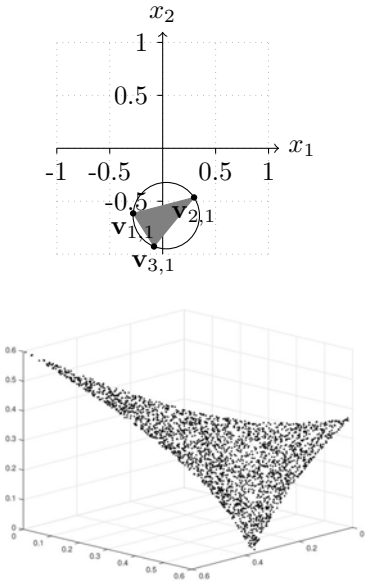


Figure 4 A problem with three objectives, $V_i = \{\mathbf{v}_{i,j}\}$, $|V_i| = 1$; figure taken from [9]. *Left*: The three locations in $\mathcal{X}$, which lie on the circumference of the black circle, determine the objective value minima. They describe a three-sided polygonal Pareto set (coloured grey). *Right*: Samples on the corresponding Pareto front generated by Monte Carlo sampling the Pareto set.

latter approach is more convenient as it requires fixing fewer (or same number) parameters for problems with $m > 2$ objectives compared to the former approach. That is, in the context of many-objective optimization, it is advisable to adopt this approach.

Distance-based problems have been proposed initially in 2005 [18, 19]. Since then the community has suggested a number of extensions to the problem formulation to replicate different complex problem characteristics, such as arbitrarily large decision spaces that could be projected back to the 2D visualization space [22], disconnected Pareto sets of the same [15] or different shapes [14], non-identical disconnected Pareto sets [14], alternative distance metrics, e.g. the Manhattan distance [30, 28], dominance resistance regions [8], local fronts [21], and variants of real-world constraints [24]. To automate the design of feature rich distance-based problems and thus increase the uptake of these problems within the community, Fieldsend et al. [9] have also proposed a configurable test problem generator. The source code of the generator is available at the following URL: https://github.com/fieldsend/DBMOPP_generator.

Multiobjective Location Problems (Real)

The aim of this section is to discuss our investigations for multiobjective optimization problems with a special structure. Especially, we consider point-objective location problems. Using the special special structure of these multiobjective optimization problems, [3] and [11] derived suitable duality assertions and corresponding algorithms for generating the whole set of Pareto optimal solutions in the decision space.

Let m points $a^1, \ldots, a^m \in \mathbb{R}^n$ be a priori given. The distance from the new facility $x \in \mathbb{R}^n$ to a given existing facility $a^i \in \mathbb{R}^n$ will be measured by the metric induced by a norm $||\cdot||$, especially by the Euclidean norm $||\cdot||_2 : \mathbb{R}^n \to \mathbb{R}$, i.e., $||x||_2 := (\sum_{j=1}^n x_j^2)^{\frac{1}{2}}$ or by the Manhattan norm $||\cdot||_1 : \mathbb{R}^n \to \mathbb{R}$, i.e., $||x||_1 := |x_1| + \cdots + |x_n|$ or by the maximum norm $||\cdot||_\infty : \mathbb{R}^n \to \mathbb{R}$, i.e., $||x||_\infty := \max\{|x_1|, \cdots, |x_n|\}$.

The constrained point-objective location problem involving a certain norm $||\cdot||$ is formulated as:

$$\begin{cases} f(x) = (||x - -a^1||, \ldots, ||x - -a^m||) \to \min \quad \text{w.r.t. } \mathbb{R}^m_+ \\ x \in X, \end{cases} \tag{POLP$_m$}$$

where the feasible set X is a nonempty and closed set in $\mathbb{R}^n$.

Pareto Optimal Solutions: A point $x \in X$ is called Pareto optimal solution for (POLP$_m$) if

$$\nexists\, x' \in X \text{ s.t. } \begin{cases} \forall\, i \in I_m : \ ||x' - -a^i|| \leq ||x - -a^i||, \\ \exists\, j \in I_m : \ ||x' - -a^j|| < ||x - -a^j||, \end{cases}$$

where $I_m := \{1, 2, \cdots, m\}$.

The set of all Pareto optimal solutions is denoted by $\mathrm{PO}(X \mid f)$. We have

$$\mathrm{PO}(X \mid f) = \{x \in X \mid f[X] \cap (f(x) - -\mathbb{R}^m_+ \setminus \{0\}) = \emptyset\}.$$

For generating the set of all Pareto optimal solutions $\mathrm{PO}(X \mid f)$ of (POLP$_m$), we can use algorithms based on duality statements [3, 11]. These algorithms and many other algorithms for solving scalar as well as multiobjective location problems are implemented in the software *FLO* freely available at https://project-flo.de; see [13].

Generating the Solution Set: Using the software *FLO*, we generate in Figure 5 the set $\mathrm{PO}(X \mid f)$ of (POLP$_7$) with seven existing facilities a^i $(i \in I_7)$, $X = \mathbb{R}^2$ and $||\cdot|| = ||\cdot||_\infty$.

If we consider one additional existing facility a^8, i.e., we consider a multiobjective location problem with one more objective function, we generate the set of all Pareto optimal solutions $\mathrm{PO}(X \mid f)$ of (POLP$_8$) using the duality based algorithm included in *FLO*. The set of all Pareto optimal solutions $\mathrm{PO}(X \mid f)$ of (POLP$_8$) is given in Figure 6. It is possible to see that the solution set of (POLP$_8$) is very different from the solution set of (POLP$_7$).

Furthermore, if we consider the multiobjective location problem (POLP$_8$) where the *Manhattan norm* is involved, we get the set of all Pareto optimal solutions reported in Figure 7.

At last, Figure 8 shows $\mathrm{PO}(X \mid f)$ of (POLP$_{11}$) with different norms.

Summary: Using the special structure of the multiobjective location problem, it is of interest to study the influence of the number of objectives (i.e., the number of existing facilities) from the theoretical as well as numerical point of view. Experimental results could be derived

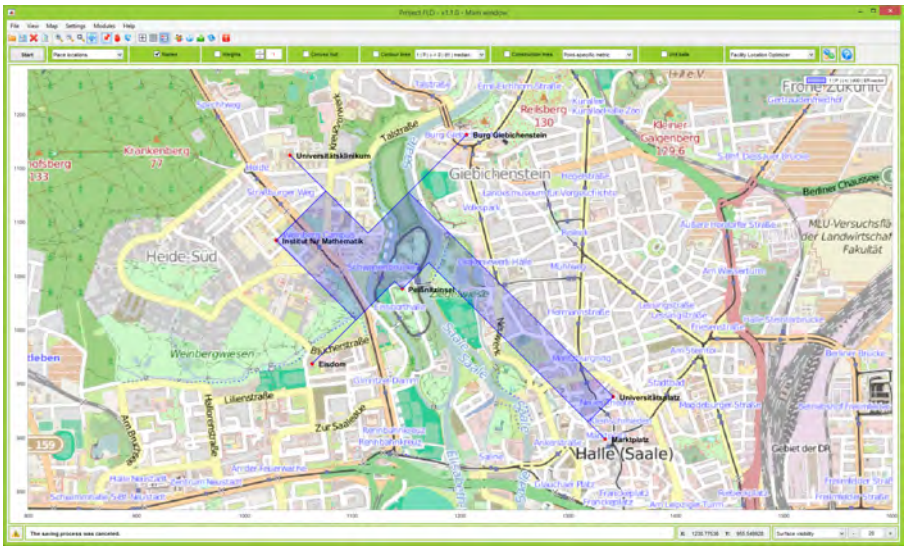


Figure 5 PO($X \mid f$) of (POLP$_7$) where the maximum norm is involved.

for multiobjective location problems using the software *FLO*. Especially, the influence of the number of existing facilities on the algorithm is of interest. Decomposition methods and scalarization based algorithms for generating the set PO($X \mid f$) of (POLP$_m$) should be derived. Approaches based on multiple scalar subproblems could be developed for solving multiobjective location problems.

Furthermore, it would be interesting to consider variable domination structures in (POLP$_m$).

The results can be used in several fields of applications:

- **Location theory**.
- **Economics:** Considering models in utility theory (Cobb-Douglas-function) and production theory.
- **Bioinformatics:** Considering entropy maximization models (based on entropies by [23] for DNA sequence analysis.

4.1.5 Conclusions

Summary

This working group has analyzed the impact an increase in the number of objectives has on (i) problem characteristics and the (ii) complexity of multiobjective procedures and algorithms. Several existing feature-rich test instance generators for problems with many objectives where covered as well. These generators can be used, for example, to validate the observations made but also test the performance of many-objective optimization algorithms on problems with different properties. Our main findings in terms of scaling efficiencies can be summarized as:

- *Good* scaling behavior (i.e., polynomially):
  - Single scalarizations,
  - Approximate hypervolume (though the approximation quality itself may be affected).
- *Relatively good* scaling behavior:
  - Updating the archive of non-dominated solutions.

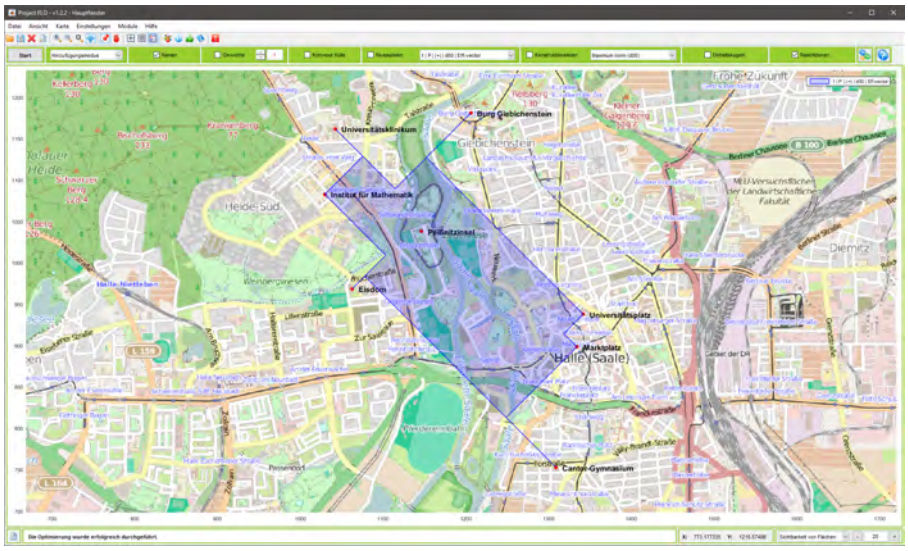


Figure 6 PO($X \mid f$) of (POLP_8) where the maximum norm is involved.

- *Poor* scaling behavior (i.e., exponential complexity, decreased quality):
  - Exact hypervolume computation,
  - Approximating the whole PF with guaranteed quality.

Future/Ongoing work

The following directions for future work have been identified:

- Verify theoretical properties on artificial and real problems, problems with special structures.
- Investigate a proper setting for the population size or the number of weights (approximation quality) w.r.t the number of objectives and the correlation among them.
- Investigate the accuracy of Monte Carlo estimation for the hypervolume, how many sampling points w.r.t the number of objectives?
- Changing DM preferences for generating preferred solutions.

References

1 Hernan. Aguirre and Kiyoshi Tanaka. Working principles, behavior, and performance of MOEAs on MNK-landscapes. *European Journal of Operational Research*, 181(3):1670–1690, 2007.

2 Richard Allmendinger, Julia Handl, and Joshua Knowles. Multiobjective optimization: When objectives exhibit non-uniform latencies. *European Journal of Operational Research*, 243(2):497–513, 2015.

3 Shaghaf Alzorba, Christian Günther, Nicolae Popovici, and Christiane Tammer. A new algorithm for solving planar multiobjective location problems involving the manhattan norm. *European Journal of Operational Research*, 258(1):35–46, 2017.

4 Johannes Bader and Eckart Zitzler. HypE: An algorithm for fast hypervolume-based many-objective optimization. *Evolutionary Computation*, 19(1):45–76, 2011.

5 Cristina Bazgan, Florian Jamain, and Daniel Vanderpooten. On the number of non-dominated points of a multicriteria optimization problem. *Discrete Applied Mathematics*, 161(18):2841 – 2850, 2013.

6 Tinkle Chugh, Richard Allmendinger, Vesa Ojalehto, and Kaisa Miettinen. Surrogate-assisted evolutionary biobjective optimization for objectives with non-uniform latencies.

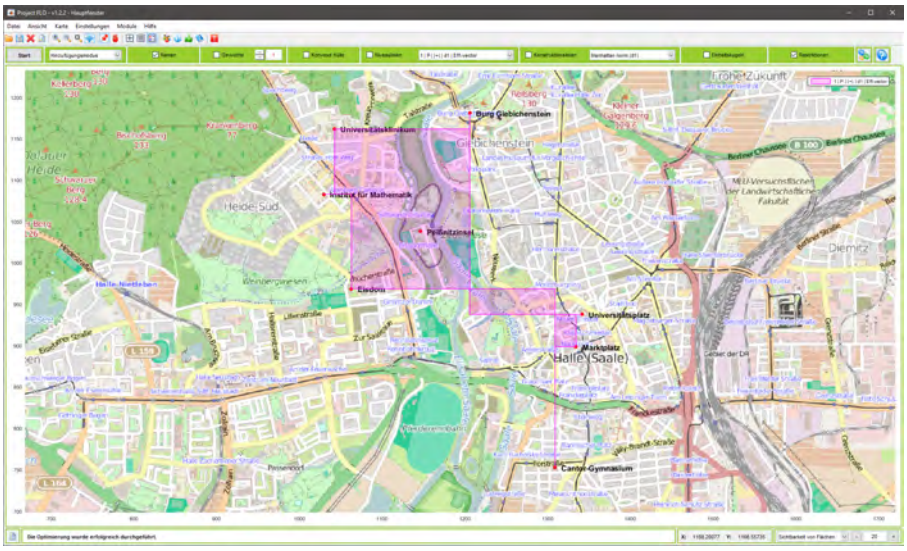


Figure 7 PO($X \mid f$) of (POLP_8) where the Manhattan norm is involved.

In *Proceedings of the Genetic and Evolutionary Computation Conference*, pages 609–616, 2018.

7 Kalyanmoy Deb, Amrit Pratap, Sameer Agarwal, and TAMT Meyarivan. A fast and elitist multiobjective genetic algorithm: Nsga-ii. *IEEE transactions on evolutionary computation*, 6(2):182–197, 2002.

8 Jonathan E. Fieldsend. Enabling dominance resistance in visualisable distance-based many-objective problems. In *Proceedings of the 2016 on Genetic and Evolutionary Computation Conference Companion*, pages 1429–1436. ACM, 2016.

9 Jonathan E. Fieldsend, Tinkle Chugh, Richard Allmendinger, and Kaisa Miettinen. A feature rich distance-based many-objective visualisable test problem generator. In *Proceedings of the Genetic and Evolutionary Computation Conference*, GECCO ,äô19, page 541,äì549, New York, NY, USA, 2019. Association for Computing Machinery.

10 Jonathan E. Fieldsend, Richard. M. Everson, and Sameer Singh. Using unconstrained elite archives for multiobjective optimization. *IEEE Transactions on Evolutionary Computation*, 7(3):305–323, June 2003.

11 Christiane Gerth and Karin Pöhler. Dualität und algorithmische anwendung beim vektoriellen standortproblem. *Optimization*, 19(4):491–512, 1988.

12 Tobias Glasmachers. A fast incremental bsp tree archive for non-dominated points. In Heike Trautmann, Günter Rudolph, Kathrin Klamroth, Oliver Schütze, Margaret Wiecek, Yaochu Jin, and Christian Grimme, editors, *Evolutionary Multi-Criterion Optimization*, pages 252–266, Cham, 2017. Springer International Publishing.

13 Christian Günther and Christiane Tammer. Relationships between constrained and unconstrained multi-objective optimization and application in location theory. *Mathematical Methods of Operations Research*, 84(2):359–387, 2016.

14 Hisao Ishibuchi, Naoya Akedo, and Yusuke Nojima. A many-objective test problem for visually examining diversity maintenance behavior in a decision space. In *Proceedings of the 13th Annual Conference on Genetic and Evolutionary Computation*, pages 649–656. ACM, 2011.

15 Hisao Ishibuchi, Yasuhiro Hitotsuyanagi, Noritaka Tsukamoto, and Yusuke Nojima. Many-objective test problems to visually examine the behavior of multiobjective evolution in a decision space. In R. Schaefer, C. Cotta, J. Kolodziej, and G. Rudolph, editors, *Parallel*

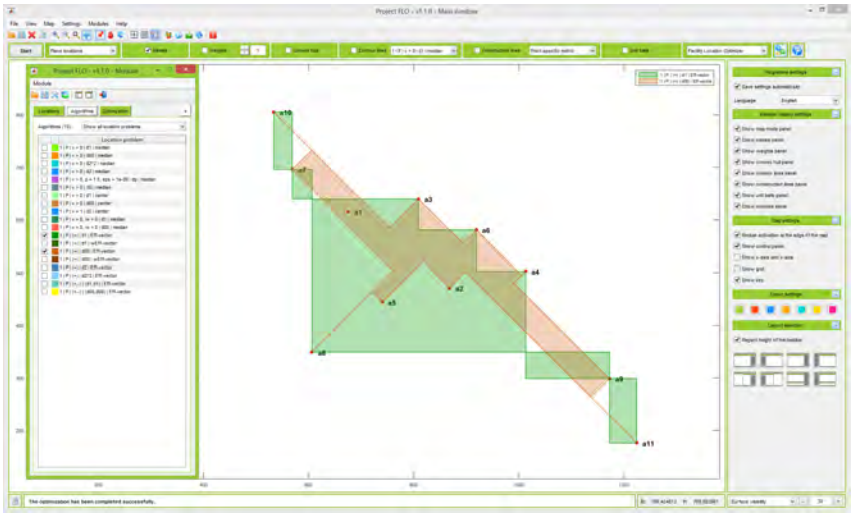


Figure 8 PO($X \mid f$) of (POLP$_{11}$) where the Manhattan norm (green) and the maximum norm (red) are involved (generated using *FLO*).

Problem Solving from Nature, PPSN XI, 11th International Conference, Proceedings, Part II, pages 91–100, Berlin, Heidelberg, 2010. Springer.

16 Andrzej Jaszkiewicz and Thibaut Lust. ND-Tree-based update: A fast algorithm for the dynamic nondominance problem. *IEEE Transactions on Evolutionary Computation*, 22(5):778–791, Oct 2018.

17 Stuart A. Kauffman. *The Origins of Order*. Oxford University Press, 1993.

18 Mario Köppen, Raul Vicente-Garcia, and Bertram Nickolay. Fuzzy-Pareto-dominance and its application in evolutionary multi-objective optimization. In C. A. Coello Coello, A. H. Aguirre, and E. Zitzler, editors, *Evolutionary Multi-Criterion Optimization, Third International Conference, Proceedings*, pages 399–412, Berlin, Heidelberg, 2005. Springer.

19 Mario Köppen and Kaori Yoshida. Visualization of Pareto-sets in evolutionary multi-objective optimization. In *Proceedings of the 7th International Conference on Hybrid Intelligent Systems*, pages 156–161, 2007.

20 Aarnaud Liefooghe, Fabio Daolio, Sebastien Verel, Bilel Derbel, Hernan Aguirre, and Kiyoshi Tanaka. Landscape-aware performance prediction for evolutionary multi-objective optimization. *IEEE Transactions on Evolutionary Computation*, page (early access), 2019.

21 Yiping Liu, Hisao Ishibuchi, Yusuke Nojima, Naoki Masuyama, and Ke Shang. A double-niched evolutionary algorithm and its behavior on polygon-based problems. In *International Conference on Parallel Problem Solving from Nature*, pages 262–273. Springer, 2018.

22 Hiroyuki Masuda, Yusuke Nojima, and Hisao Ishibuchi. Visual examination of the behavior of EMO algorithms for many-objective optimization with many decision variables. In *IEEE Congress on Evolutionary Computation (CEC), Proceedings*, pages 2633–2640. IEEE, 2014.

23 Tomasz Maszczyk and Włodzisław Duch. Comparison of shannon, renyi and tsallis entropy used in decision trees. In *International Conference on Artificial Intelligence and Soft Computing*, pages 643–651. Springer, 2008.

24 Yusuke Nojima, Yiping Liu Takafumi Fukase, Naoki Masuyama, and Hisao Ishibuchi. Constrained multiobjective distance minimization problems. In *Proceedings of the 2019 on Genetic and Evolutionary Computation Conference Companion*, pages 1–1. ACM, 2019.

25 Qingfu Zhang and Hui Li. MOEA/D: A Multiobjective Evolutionary Algorithm Based on Decomposition. *IEEE Transactions on Evolutionary Computation*, 11(6):712–731, 2007.

26 Roseanna W Saaty. The analytic hierarchy process,äîwhat it is and how it is used. *Mathematical modelling*, 9(3-5):161–176, 1987.

27 Sébastien Verel, Arnaud Liefooghe, Laetitia Jourdan, and Clarisse Dhaenens. On the structure of multiobjective combinatorial search space: MNK-landscapes with correlated objectives. *European Journal of Operational Research*, 227(2):331–342, 2013.

28 Justin Xu, Kalyanmoy Deb, and Abhinav Gaur. Identifying the Pareto-optimal solutions for multi-point distance minimization problem in Manhattan space. Technical Report COIN Report Number 2015018, Michigan State University, 2015.

29 Qingfu Zhang and Hui Li. Moea/d: A multiobjective evolutionary algorithm based on decomposition. *IEEE Transactions on evolutionary computation*, 11(6):712–731, 2007.

30 Heiner Zille and Sanaz Mostaghim. Properties of scalable distance minimization problems using the Manhattan metric. In *2015 IEEE Congress on Evolutionary Computation (CEC), Proceedings*, pages 2875–2882. IEEE, 2015.

31 Eckart Zitzler and Simon Künzli. Indicator-based selection in multiobjective search. In *International conference on parallel problem solving from nature*, pages 832–842. Springer, 2004.

32 Eckart Zitzler, Lothar Thiele, Marco Laumanns, Carlos M. Fonseca, and Viviane Grunert da Fonseca. Performance assessment of multiobjective optimizers: An analysis and review. *IEEE Transactions on Evolutionary Computation*, 7(2):117–132, 2003.

4.2 The Output-sensitive Complexity of the BUCO Problem

Fritz Bökler (Universität Osnabrück, DE), Matthias Ehrgott (Lancaster University, GB), José Rui Figueira (IST – Lisbon, PT), Andreia P. Guerreiro (IST – Lisbon, PT), Kathrin Klamroth (Universität Wuppertal, DE), Britta Schulze (Universität Wuppertal, DE), and Daniel Vanderpooten (University Paris-Dauphine, FR)

The question how the running-times of algorithms scale with respect to certain parameters of the input lies at the heart of computational complexity theory. Traditionally, we are interested in how the worst-case running-time of a given algorithm for a given problem scales in the size of the input. Especially, whether this function scales polynomially. For multiobjective combinatorial optimization (MOCO) problems, we often know a negative answer. In particular, for multiobjective variants of the shortest path, spanning tree, assignment, and many more problems, there is no algorithm with a running-time scaling polynomially in the input size [3].

But what happens, if we investigate the running-time of an algorithm as a function of more than just the input size? A new approach in multiobjective optimization is to investigate the running-time as a function of the input size and the output size. Other possible viewpoints include fixed-parameter tractability, wherein the running-time is studied in the input-size and a set of parameters that are assumed to be fixed (cf., e.g., [2]).

In our group, we studied a problem which is easy to describe, but a negative or positive answer has far-reaching consequences. Our object of study was the biobjective unconstrained combinatorial optimization (BUCO) problem given in the following mathematical form:

$$\begin{aligned} &\max\ {c^1}^{\mathsf{T}} x \\ &\min\ {c^2}^{\mathsf{T}} x \\ &\text{s.t.} \quad x \in \{0,1\}^n \end{aligned}$$

Where $c^1, c^2 \in \mathbb{N}^n$ and $n \in \mathbb{N}$. We are interested in the set of *nondominated points* of this problem, i.e., $\mathcal{Y}_N \min\left\{(-c^1, c^2)^x \mid x \in \{0,1\}^n\right\}$, where min denotes the set of minimal elements with respect to the component-wise less-or-equal order on vectors.

An intuitive description of the problem is the following: We are given n items. Item $i \in \{1, \ldots, n\}$ has profit c_i^1 and weight c_i^2. A solution is a filling of a knapsack with a subset of these items. The profit of a filling is the sum of the profits of its items and the weight of a filling is the sum of the weights of its items. We want to maximize the profit and minimize the weight but these objectives may be conflicting. Consequently, we want to find the set of best compromises between the profit and weight of those knapsack fillings.

In [1] it is proven that if there is no output-sensitive algorithm for the BUCO problem then there is also none for the multiobjective spanning tree problem. Thus, a negative result is especially interesting, since the output-sensitive complexity of the multiobjective spanning-tree problem is open. On the other hand, as this problem is of very general nature, a positive result for the BUCO problem may result in new solution methods for MOCO problems in general. We investigated several angles to attack the problem:

The Nemhauser-Ullmann Algorithm

With a given BUCO instance (c^1, c^2), we associate a set of knapsack (KP) instances parameterized by $k \in \mathbb{N}$:

$$\begin{aligned} \max\ & {c^1}^{\mathsf{T}} x \\ \text{s.t.}\ & {c^2}^{\mathsf{T}} m x \leqslant k \\ & x \in \{0,1\}^n \end{aligned}$$

One algorithm to solve KP is the Nemhauser-Ullmann (NU) algorithm [4, 5]. The basic idea is to iteratively consider a growing subset of items. We start with the empty set that only allows for the empty solution and the cost-vector $(0,0)^{\mathsf{T}}$. Iteratively, we consider one more item in the order given by the input, add its weight and profit to all previous vectors, join these new vectors with the old ones, and delete the dominated vectors. This computes the nondominated set of the BUCO problem. Moreover, for each given k, we can find the maximum value of the corresponding KP instance among this nondominated set. It is unknown whether the NU algorithm is output-sensitive for the BUCO problem.

The main difficulty arises as follows: For each index $i \in \{0, \ldots, n\}$, we define $T(i)$ as the set of solutions after the ith iteration. Thus, $T(0) = \left\{(0,0)^{\mathsf{T}}\right\}$ and $T(n) = \mathcal{Y}_N$. Can it happen that one of the $T(i)$ for $i \in \{1, \ldots, n-1\}$ is much larger, say more than any polynomial in n and $|\mathcal{Y}_N|$ larger, than $T(n)$? Moreover, while changing the ordering of the items does not change $T(n)$, the sets $T(i)$ for $i \in \{1, \ldots n-1\}$ can change significantly. We thus discussed the behavior of the algorithm with respect to several orderings, including

- ordering by $\frac{c_i^1}{c_i^2}$,
- lexicographic ordering, and
- ordering by $\left\| \begin{pmatrix} c_i^1 \\ c_i^2 \end{pmatrix} \right\|$.

Parameter: Number of large items. The NU algorithm solves the BUCO problem in pseudo-polynomial time. An implication of pseudo-polynomiality is the following: If all numbers in the input are polynomially bounded by the input size, these instances are polynomial-time solvable. We asked the question: What happens if we allow only a small number of super-polynomial items? More formally: Let $p\colon \mathbb{N} \to \mathbb{N}$ be a polynomial. For a

given instance, let n be the number of items, and let k be the number of items that have at least one of c_i^1, c_i^2 with value larger than $p(n)$. Is there an algorithm solving BUCO with a running-time bounded by $\mathcal{O}(f(k) \cdot \text{poly}(n))$ for any computable function f? We answered this question affirmatively.

Computing supported nondominated points. The *weighted-sum scalarization* for a given $\lambda \in \mathbb{R}^2_>$ of the BUCO problem is the following problem:

$$\begin{aligned} 2\max \quad & \lambda_1 {c^1}^\mathsf{T} x - -\lambda_2 {c^2}^\mathsf{T} x \\ \text{s.t.} \quad & x \in \{0,1\}^n \end{aligned}$$

A point $y \in \mathcal{Y}_N$ is called a *supported nondominated point*, if there is a $\lambda \in \mathbb{R}^2_>$ such that there is an optimal solution x to the weighted-sum-scalarization with weight λ and ${c^1}^\mathsf{T} x = y_1$ and ${c^2}^\mathsf{T} x = y_2$. A point $y \in \mathcal{Y}_N$ is called an *extreme nondominated point*, if there is a $\lambda \in \mathbb{R}^2_>$ with the above property that leads to no other point $y' \in \mathcal{Y}_N \backslash \{y\}$.

In [6] it is proven that the set of extreme nondominated points of BUCO can be found in polynomial time. However, the number of supported nondominated points can be exponential in the input size. As a third question we thus asked, if we can find all of the supported nondominated points in output-sensitive running-time. While the set of Pareto-optimal supported solutions can be computed in an output-sensitive way by a local search scheme, this is not clear for its image. It can happen that many of the Pareto-optimal supported solutions map to the same point in the objective space, prohibiting an output-sensitive running-time.

We also answered this question in the affirmative. The general idea is to use the characterization in [6] and enumerate solutions comprised of objects with the same profit-to-weight ratios in an output-sensitive way. We also discussed further ideas for the generalized problem, where more than two objectives are present and we aim to continue this line of research in the future.

Acknowledgements

Andreia P. Guerreiro and José Rui Figueira acknowledge the support from DOME (Discrete Optimization Methods for Energy management) FCT Research Project (Ref: PTDC/CCI-COM/31198/2017).

References

1 Bökler, Fritz Output-sensitive Complexity of Multiobjective Combinatorial Optimization Problems with an Application to the Multiobjective Shortest Path Problem. PhD Thesis, Department of Computer Science, TU Dortmund University, 2018.
2 Downey, Rodney G. and Michael R. Fellows. Parameterized Complexity. Springer, 1999.
3 Ehrgott, Matthias. Multicriteria Optimization. Springer, 2005.
4 Kellerer, Hans, Ulrich Pferschy, and David Pisinger. Knapsack Problems. Springer, 2004.
5 Nemhauser, Georg L., Zev Ullmann. Discrete dynamic programming and capital allocation. Management Science 15, pp. 494–263, 1994.
6 Schulze, Britta, Michael Stiglmayr, and Kathrin Klamroth. Multi-objective unconstrained combinatorial optimization: a polynomial bound on the number of extreme supported solutions. Journal of Global Optimization 74(3), pp. 495-–522, 2019.

4.3 Computationally Expensive Functions and Large Scale Test Instances

Dimo Brockhoff (INRIA Saclay – Palaiseau, FR), Gabriele Eichfelder (TU Ilmenau, DE), Carlos M. Fonseca (University of Coimbra, PT), Susan R. Hunter (Purdue University, US), Enrico Rigoni (ESTECO SpA – Trieste, IT), Michael Stiglmayr (Universität Wuppertal, DE)

4.3.1 Introduction

The group first discussed general issues and current challenges related to scaling up to solve large-scale problems. The group identified a number of issues and open research questions on which we give an overview in Section 4.3.2. The group examined in more detail the issue of computationally heavy problems, i.e. problems where function evaluations are computationally demanding, see Section 4.3.3. Further, when developing new algorithms for large-scale problems, the group emphasized the importance of being able to test the algorithms well. The group identified a lack of test instances which are not just simple extensions from the single-objective to the multiobjective setting but which really transfer single-objective challenges to the mutliobjective case or highlight challenges inherent to the multiobjective setting that may not exist in the single-objective setting. We summarize this discussion and provide some solution approaches in Section 4.3.4.

4.3.2 Large Scale Issues and Current Challenges

In application problems and thus in optimization problems of interest there can be many factors that imply a multiobjective problem to be of "large scale." These factors include a large number of variables, a large amount of computation time to evaluate the objective functions which may be outputs of a black box [16], a large number of processors that must evaluate the objectives, the existence of a large Pareto front (especially in discrete problems), or the solution approach which must navigate a large number of local Pareto fronts.

The group identified several open research topics that relate to the identified factors implying a multiobjective problem is large-scale. These research questions apply to either very specific problem formulations or to a more general setting. Some of these open research topics include the following:

- *Many decision variables:* If the multiobjective optimization problem has a large number of decision variables, and particularly if there are many integer variables, constructive methods such as Branch-and-Bound may be too computationally burdensome. For instance, in [3] the proposed Branch-and-Bound procedure could only solve mixed-integer convex multiobjective problems up to 30 integer variables. New theory and methods are needed to handle this scenario. In particular, the group identified column generation for multiobjective combinatorial problems (MOCO) and complexity theory for multiobjective unconstrained combinatorial optimization (MUCO) as possible research topics.
- *Computationally heavy problems:* If the objective functions are the output of a black-box, such as a deterministic black-box oracle [17] or a *Monte-Carlo* simulation oracle [13], one or more of the objective or constraint function evaluations may be time consuming. Thus, only a limited number of function evaluations may be possible. Furthermore, in the case of a Monte-Carlo simulation oracle, more than one processor may be required to obtain a sufficiently accurate estimator of the objective function value [10, 8, 9, 14]. How should

Table 1 Comparison of trust region methods with surrogate methods.

Trust Region Methods	Surrogate Methods
uses a local model, determined only with information along a "path" in the area you trust	uses a global model
basically a local solver, i.e. aims to find locally optimal solutions	aims to find a globally optimal solution
refined when coming close to an optimum	refined in promising areas where you do not trust the model (high variance)
requires model assumptions, e.g., trust the model	requires model assumptions, e.g. Gaussian process

such problems be approached? Trust-region [24, 23] or surrogate modeling may be useful tools in this context; how can they be modified to ensure efficiency?

- *Many local Pareto sets:* In many single-objective settings, local methods are used in the search for global optima (e.g.[18, 19, 15]). Does a similar framework make sense in a multiobjective setting, and if so, how should solvers handle many local Pareto sets? How can one avoid calculating unnecessary Pareto sets, and how might one go about storing the data generated when solving such problems? In particular, mixed-integer nonlinear multiobjective optimization problems might have a huge number of local Pareto sets, one for each fixed setting of the integer variables. Then, the overall Pareto set is the non-dominated points among them, which can be identified by comparing sets. How can such calculations be conducted efficiently?
- *Test instances:* While a large number of test instances exist for multiobjective optimization (e.g. [2, 1, 12, 6, 5, 4]), there is a lack of scalable test instances that truly extend the challenges observed in a single-objective setting to the multiobjective setting. Such test instances are required to test algorithms developed for large-scale problems.

4.3.3 Computationally Heavy Problems

The group decided first to focus specifically on computationally heavy problems, i.e., on optimization problems where the functions are large-scale in terms of computational time. Furthermore, the group envisaged to identify how such problems might be approached using surrogate or trust-region methods. First, we summarize similarities and differences between these methods, which we display in Table 1.

Especially the idea of a local model which is refined close to the optimum can be used in many fields. There are also different possibilities for the model itself: one can built a model for each objective individually, as done in the trust region approaches [22, 24], or one can define a global one based on scalarization, as for instance for the hypervolume or for a multiplicative reformulation [22].

When such a model approach should be chosen for computationally heavy objectives, then one has also to keep in mind the following aspects. First, the question is whether all function evaluations are expensive, or whether it will be cheaper to calculate some value $f(y)$ for y close to some x in case one has already calculated $f(x)$. An example for that might be mesh adaptation or just slight changes in a mesh. It is still an open research question how this can be used for efficient algorithms, for instance for the numerical approximation of derivatives. Of course, this issue is also a task for single-objective expensive optimization.

Another aspect is that the objective functions might be of different type (heterogeneous), like, e.g., different simulation times (3h vs 20 minutes; 2h vs 1 second). See for example [7, 24].

Moreover, in the case that there are different levels of granularities or accuracies of the model an efficient algorithm should choose adaptively. Such approaches already exist, but theoretical proofs are always a true challenge.

Hence, when dealing with computationally heavy problems, several aspects have to be taken into account which are not directly related to the presence of multiple objectives. Moreover, there are many different problem types (heterogeneous, stochastic, locally cheaper, ...) and methods have to be developed for each problem type individually.

4.3.4 Consistent and scalable extensions of test instances for large scale

There is already a large number of test instances available for multiobjective solvers. Nevertheless, we see a lack in instances which truly transfer the single-objective challenges to the multiobjective setting. Often, the difficulties known from single-objective optimization are just transferred to one of the objective functions, while the other objective functions are just chosen to be of a quite simple structure to obtain a simple extension.

An example in this direction is the test instance known as DTLZ7 [5], denoted MaF7 in [2], with

$$\min_{x\in\mathbb{R}^n} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_{m-1} \\ f_m(x) \end{pmatrix}$$

with $1 \leq m \leq n$ and functions $f_m, g\colon \mathbb{R}^n \to \mathbb{R}$,

$$f_m(x) = m - \sum_{i=1}^{m-1} \left(\frac{x_i}{1+g(x)} (1 + \sin(3\pi x_i)) \right)$$

and $g(x) = 1 + \frac{9}{n-m+1} \sum_{i=m}^{n} x_i$. Other instances with many objectives are for instance obtained by taking sinus and cosinus values of the components x_i and by a multiplication of them.

Hence, we suggest to construct new multiobjective test instances which extend the classical test functions as the *Rosenbrock* [20] or the *Himmelblau function* [11]. Thereby, we want to use a consistent extension, i.e., all objective functions should be of the same type. Moreover, to be usable for large scale issues, the test instances should be scalable w.r.t. the number of objective functions and the number of variables. Of course, as expected from test instances, they should have predefined mathematical properties as, e.g., a known Pareto set.

Another aspect is that real-world problems often exhibit a complex correlation structure between objectives: conflict and agreement among objective functions are in general local properties. For example, Figure 9 shows the correlation structure of the crankshaft problem [21] on the Pareto front. Available multiobjective test instances consider in general only problems with independent objectives, missing the complexity of real-world scenarios.

Consequently, test instances should have the property that when adding functions the problem should keep the structure in the following sense:

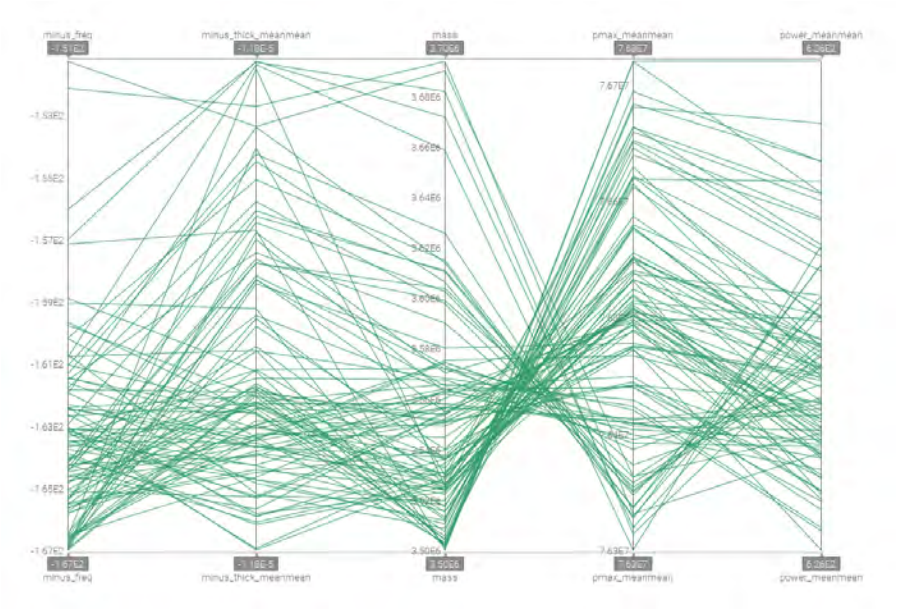


Figure 9 Parallel coordinates chart of an empirical Pareto set of the crankshaft problem. Signs have been changed so that all five objectives are to be minimized. Both conflict and agreement among different objectives are visible.

Assume there are k objectives $f_1, \ldots, f_k \colon \mathbb{R}^n \to \mathbb{R}$. Then these define some kind of ordering in the pre-image space in the sense that we say

$$x \leq_f \tilde{x} \ :\iff \ (f_1(x), \ldots, f_k(x)) \leq (f_1(\tilde{x}), \ldots, f_k(\tilde{x})). \tag{1}$$

Now we aim on defining for the test instances a new objective $f_{k+1} \colon \mathbb{R}^n \to \mathbb{R}$, which is not just a copy of the previous objective functions, but that also keeps this ordering, meaning

$$x \leq_f \tilde{x} \iff (f_1(x), \ldots, f_k(x), f_{k+1}(x)) \leq (f_1(\tilde{x}), \ldots, f_k(\tilde{x}), f_{k+1}(\tilde{x})).$$

The overall idea is that if we want to study scalability with respect to the number of objectives, we should keep everything else fixed. Furthermore, this requirement introduces a rich correlation structure between objectives as a fringe benefit.

We illustrate this with a biobjective and a triobjective example based on the Rosenbrock function [20]. First, recall that the Rosenbrock function $f \colon \mathbb{R}^2 \to \mathbb{R}$ is defined by

$$f(x, y) = (a - x)^2 + b(y - x^2)^2$$

and it has a unique global minimum at $(x, y) = (a, a^2)$, where $f(x, y) = 0$. Usually the parameters are set such that $a = 1$ and $b = 100$. Then the global minimum, located at $(1, 1)$, is inside a narrow, parabolic shaped valley, as shown in Figure 10.

▶ **Example 1.** Let $a, c, d \in \mathbb{R}$ and $b > 0$ be scalars and define $f_a \colon \mathbb{R}^2 \to \mathbb{R}$ by

$$f_a(x, y) = (a - x)^2 + b(y - x^2)^2.$$

We define a second function by $f_d \colon \mathbb{R}^2 \to \mathbb{R}$,

$$f_d(x, y) = (d - x)^2 + b(y - x^2)^2.$$

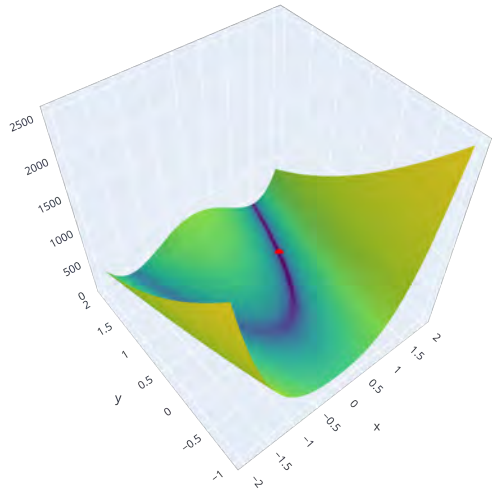


Figure 10 Plot of the Rosenbrock function with $a = 1$ and $b = 100$. The unique global minimum, outlined with a red point, is at $(1, 1)$.

Figure 11 shows that the shape of the function and the position of the global minimum are affected by the value of a (or d).

Then a simple calculation yields that the set of efficient points of the biobjective optimization problem

$$\min_{(x,y)\in\mathbb{R}^2} (f_a(x, y), f_d(x, y))$$

is

$$\mathbb{S} := \{(x, y) \in \mathbb{R}^2 \mid x = (1 - \lambda)a + \lambda d,\ y = x^2,\ \lambda \in [0, 1]\}.$$

An example of optimal solution set $\mathbb{S}$ is shown in Figure 12, where the two Rosenbrock functions presented in Figure 11 are used as objectives for a biobjective optimization problem. The relevant Pareto front is presented in Figure 13.

These two objectives f_a, f_d define now an ordering as given in (1). Now we can add a third new objective $f_c \colon \mathbb{R}^2 \to \mathbb{R}$ with the same structure, i.e.

$$f_c(x, y) = (c - x)^2 + b(y - x^2)^2$$

and as long as $c \in [a, d]$ the optimal solution set of the three objective problem

$$\min_{(x,y)\in\mathbb{R}^2} (f_a(x, y), f_d(x, y), f_c(x, y))$$

is still equal to $\mathbb{S}$.

Note that the discussion in the above example would cover the quadratic case for $b = 0$. This can be seen even better with the forthcoming Example 3. For the moment, let us show with the next example that also with quadratic functions $f \colon \mathbb{R}^n \to \mathbb{R}$ of the form

$$f(x) = \sum_{i=1}^{n} (a_i - x_i)^2$$

with parameters $a_i \in \mathbb{R}$, $i = 1, \ldots, n$ we get a similar structure (which is of course less challenging from an algorithmic point of view).

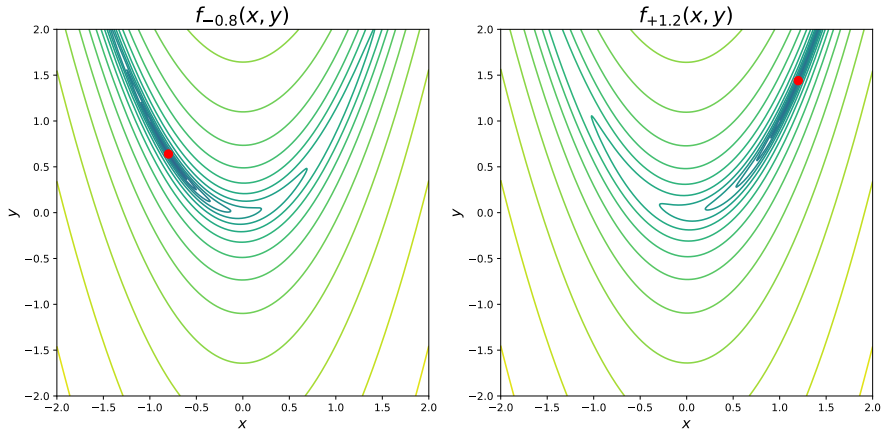


Figure 11 Contour plots of two Rosenbrock functions (in logarithmic scale) with different values for the parameter a (or d) and fixed $b = 100$: the position of the unique global minimum, outlined with a red point, changes accordingly to the value of a.

▶ **Example 2.** We consider a triobjective optimization problem with three objective functions of the type $f_j \colon \mathbb{R}^2 \to \mathbb{R}$ with

$$f_j(x, y) = (x - \alpha_j)^2 + (y - \alpha_j)^2$$

with $\alpha_j \in \mathbb{R}$, $j = 1, 2, 3$. Each individual objective function has the unique minimal solution $(x, y) = (\alpha_j, \alpha_j)$. Now let $A = (\alpha_1, \alpha_1)$, $B = (\alpha_2, \alpha_2)$, $C = (\alpha_3, \alpha_3)$. Figure 14 shows the optimal solution set for such a triobjective optimization problem. The optimal solution set is

$$\mathbb{S} := \big\{(x, y) \in \mathbb{R}^2 \mid \alpha_{\min} := \min_{j=1,2,3} \alpha_j \leq x \leq \max_{j=1,2,3} \alpha_j =: \alpha_{\max},\ y = x\big\}.$$

The Pareto front is presented in Figure 15. The solution set would not change in case one adds objectives of the above type as long as for the additional functions f_j we have $\alpha_{\min} \leq \alpha_j \leq \alpha_{\max}$. Figure 16 shows the correlation structure between objectives on the Pareto front.

The test instances should also be scalable w.r.t. the number of variables. This can be easily achieved by following the approach from the following example:

▶ **Example 3.** Let a function $g \colon \mathbb{R}^n \to \mathbb{R}$ be defined by

$$g(x) = \sum_{i=1}^{k} (a_i - x_i)^2 + \sum_{j=k+1}^{n} b_j \Big(x_j - \sum_{i \in S_j} x_i^2\Big)^2$$

where $a_i \in \mathbb{R}$, $i = 1, \ldots, k$, and $b_j > 0$, $j = k + 1, \ldots, n$, are scalars and $S_j \subseteq \{1, \ldots, k\}$ are index sets with $S_j \neq \emptyset$ for all $j = k + 1, \ldots, n$. Then the single-objective unconstrained optimization problem

$$\min_{x \in \mathbb{R}^n} g(x)$$

has the optimal solution set

$$\Big\{x \in \mathbb{R}^n \mid x_i = a_i,\ i = 1, \ldots, k,\ x_j = \sum_{i \in S_j} x_i^2\Big\}$$

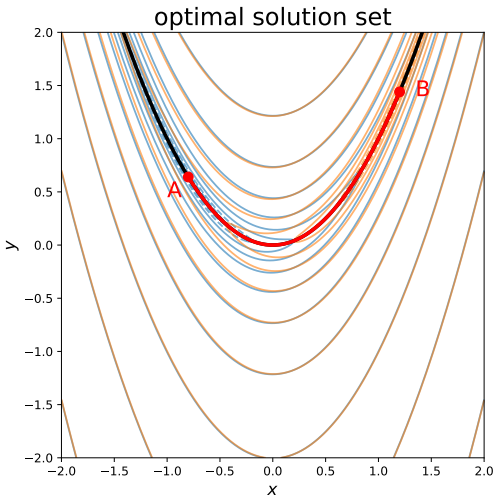


Figure 12 Optimal solution set (outlined with a red parabolic arch) for the biobjective optimization problem defined by the two objectives $f_{-0.8}(x, y)$ and $f_{+1.2}(x, y)$ shown in Figure 11. The contour plots of two objectives functions are represented in blue and in orange, respectively. The minima of the objective functions (shown as red points) are, respectively, $A = \arg\min f_{-0.8}(x, y)$ and $B = \arg\min f_{+1.2}(x, y)$.

4.3.5 Conclusion

Future work contains the formulation of clear test instances with full information on the parameters to choose and on the Pareto set. It is also of interest to discuss the above procedures on how they can be extended to other functions as to construct a multiobjective version of the Himmelblau function [11] $h\colon \mathbb{R}^2 \to \mathbb{R}$

$$h(x, y) = (x^2 + y - 11)^2 + (x + y^2 - 7)^2$$

and also to see how general the proposed approach is to generate other types of level sets and shapes of the Pareto set. Most of all on how the dimensionality of the Pareto front can be influenced and how the choice of the parameters influences the conditioning of the problem.

References

1 D. Brockhoff, T. Tusar, A. Auger, and N. Hansen. Using well-understood single-objective functions in multiobjective black-box optimization test suites. *arXiv preprint arXiv:1604.00359*, 2016.

2 R. Cheng, M. Li, Y. Tian, X. Zhang, S. Yang, Y. Jin, and X. Yao. A benchmark test suite for evolutionary many-objective optimization. *Complex & Intelligent Systems*, 3(1):67–81, 2017.

3 M. de Santis, G. Eichfelder, J. Niebling, and S. Rocktäschel. Solving multiobjective mixed integer convex optimization problems. *Optimization Online*, 2019.

4 K. Deb. Multi-objective genetic algorithms: Problem difficulties and construction of test problems. *Evolutionary Computation*, 7(3):205–230, 1999.

5 K. Deb, L. Thiele, M. Laumanns, and E. Zitzler. Scalable test problems for evolutionary multi-objective optimization. Technical report, Institut für Technische Informatik und Kommunikationsnetze, ETH Zürich, 2001. TIK-Technical Report No. 112.

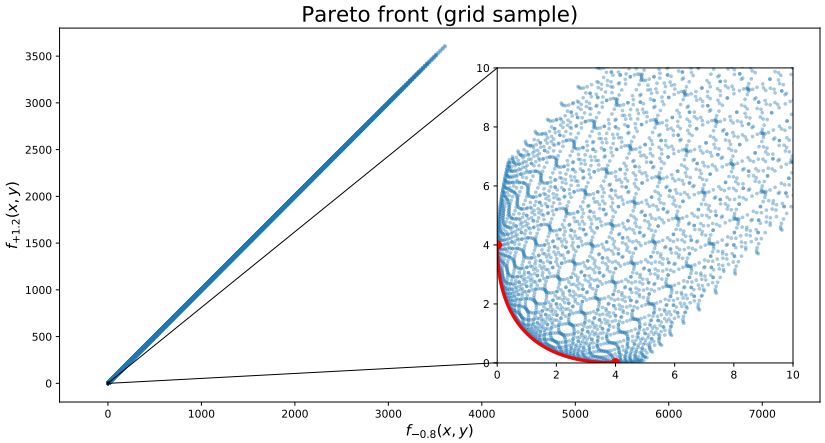


Figure 13 Objective space of the biobjective optimization problem presented in Figure 12: the Pareto front is outlined with a red curve. The blue points represent the image of the objective functions when the domain is sampled on a regular grid. It is evident that the two objectives are highly correlated.

6 K. Deb, L. Thiele, M. Laumanns, and E. Zitzler. Scalable multi-objective optimization test problems. In *Proceedings of the 2002 Congress on Evolutionary Computation*, volume 1, pages 825–830, Piscataway, NJ, May 2002. IEEE.

7 G. Eichfelder, X. Gandibleux, M.J. Geiger, J. Jahn, A. Jaszkiewicz, J. Knowles, P.K. Shukla, H. Trautmann, and S. Wessing. Heterogeneous functions. *In: Understanding Complexity in Multiobjective Optimization, Dagstuhl Seminar 15031*, pages 121–129, 2015.

8 P. W. Glynn and P. Heidelberger. Bias properties of budget constrained simulations. *Operations Research*, 38(5):801–814, 1990.

9 P. W. Glynn and P. Heidelberger. Analysis of parallel replicated simulations under a completion time constraint. *ACM Transactions on Modeling and Computer Simulations*, 1(1):3–23, 1991.

10 P. Heidelberger. Discrete event simulations and parallel processing: statistical properties. *Siam J. Stat. Comput.*, 9(6):1114–1132, 1988.

11 D. M. Himmelblau. *Applied nonlinear programming.* McGraw-Hill, 1972.

12 S. Huband, P. Hingston, L. Barone, and L. While. A review of multiobjective test problems and a scalable test problem toolkit. *IEEE Transactions on Evolutionary Computation*, 10(5):477–506, 2006.

13 S. R. Hunter, E. A. Applegate, V. Arora, B. Chong, K. Cooper, O. Rincón-Guevara, and C. Vivas-Valencia. An introduction to multi-objective simulation optimization. *ACM Transactions on Modeling and Computer Simulation*, 29(1):7:1–7:36, January 2019.

14 S. R. Hunter and B. L. Nelson. Parallel ranking and selection. In A. Tolk, J. Fowler, G. Shao, and E. Yücesan, editors, *Advances in Modeling and Simulation: Seminal Research from 50 Years of Winter Simulation Conferences*, Simulation Foundations, Methods and Applications, chapter 12, pages 249–275. Springer International, Switzerland, 2017.

15 J. Larson and S. M. Wild. A batch, derivative-free algorithm for finding multiple local minima. *Optimization and Engineering*, 17(1):205–228, March 2016.

16 Y. Nesterov. *Introductory lectures on convex optimization: A basic course*, volume 87 of *Applied Optimization*. Springer, 2004.

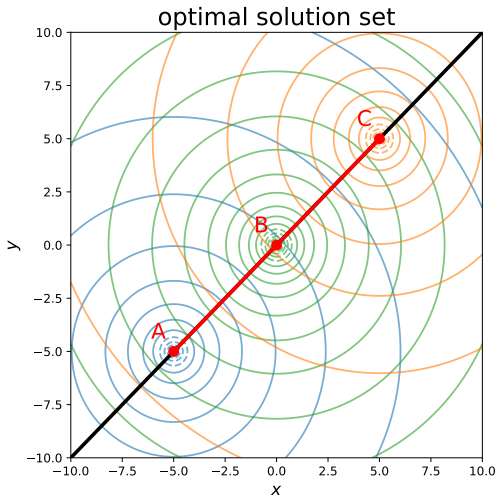


Figure 14 Optimal solution set (outlined with a red segment) for a triobjective optimization problem defined by three quadratic functions as objectives. The contour plots of the tree quadratic functions are represented in blue, green, and orange. The minima of the objective functions (shown as red points) are, respectively, A, B, and C.

17 S. Prinz, J. Thomann, G. Eichfelder, T. Boeck, and J. Schumacher. Expensive multi-objective optimization of electromagnetic mixing in a liquid metal. *Optimization Online*, 2019.

18 A. H. G. Rinnooy Kan and G. T. Timmer. Stochastic global optimization methods part I: Clustering methods. *Mathematical Programming*, 39:27–56, 1987.

19 A. H. G. Rinnooy Kan and G. T. Timmer. Stochastic global optimization methods part II: Multi level methods. *Mathematical Programming*, 39:57–78, 1987.

20 H. H. Rosenbrock. An automatic method for finding the greatest or least value of a function. *The Computer Journal*, 3(3):175–184, 01 1960.

21 Rosario Russo, Alberto Clarich, and Marco Carriglio. A multi-objective optimization of engine crankshaft design using modeFRONTIER. *International Review of Mechanical Engineering*, 6(3), 2012.

22 J.-H. Ryu and S. Kim. A derivative-free trust-region method for biobjective optimization. *SIAM Journal on Optimization*, 24:334,äì362, 2014.

23 J. Thomann and G. Eichfelder. Representation of the pareto front for heterogeneous multi-objective optimization. *Journal of Applied and Numerical Optimization*, 1(2):293–323, 2019.

24 J. Thomann and G. Eichfelder. A trust-region algorithm for heterogeneous multiobjective optimization. *SIAM Journal on Optimization*, 29(2):1017–1047, 2019.

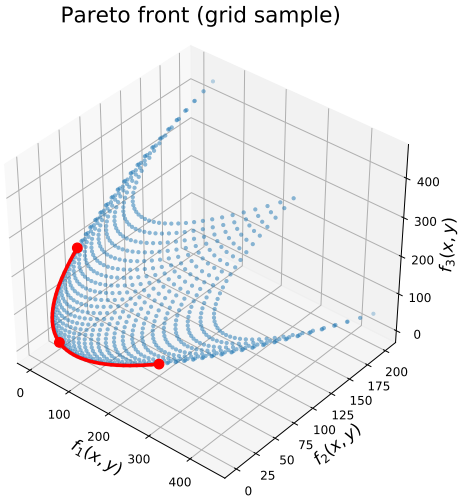


Figure 15 Objective space of the triobjective optimization problem presented in Figure 14: The Pareto front is outlined with a red curve. The blue points represent the image of the objective functions when the domain is sampled on a regular grid.

4.4 Performance Indicators

Carlos A. Coello Coello (CINVESTAV – Mexico, MX), Hisao Ishibuchi (Southern Univ. of Science and Technology – Shenzen, CN), Pascal Kerschke (Universität Münster, DE), Boris Naujoks (TH Köln, DE), and Tea Tušar (Jožef Stefan Institute – Ljubljana, SI)

4.4.1 Introduction

Within any aspect of scalability in multi-objective optimisation, performance indicators play a critical role. These indicators are important when comparing solutions and evaluating the performance of different algorithmic approaches.

Within many-objective optimisation, one special aspect of scalability in multi-objective optimisation, performance indicators recently received a lot of interest. With increasing objective space dimension, this aspect becomes more and more complex since such indicators need to fulfil different properties. Such indicators are wanted to be[1]

- Fast to compute
- Independent from the optimal Pareto set/front
- Independent from the shape of the Pareto front
- Pareto-compliant
- Not emphasizing boundary points
- Measures spread in the decision and objective space
- Scalable in the number of objectives
- Interpretable
- Featuring only a few parameters (that are easy to understand)
- Sharing the ability to represent preference of the DM
- Sharing biases of the indicator that can be understood (are known)

[1] List compiled by a working group on the topic "Performance Indicators" at the Lorentz Center workshop MACODA (MAny Criteria Optimization and Decision Analysis, 16. – 20. September 2020, http://lorentzcenter.nl/lc/web/2019/1160/info.php3?wsid=1160&venue=Oort)

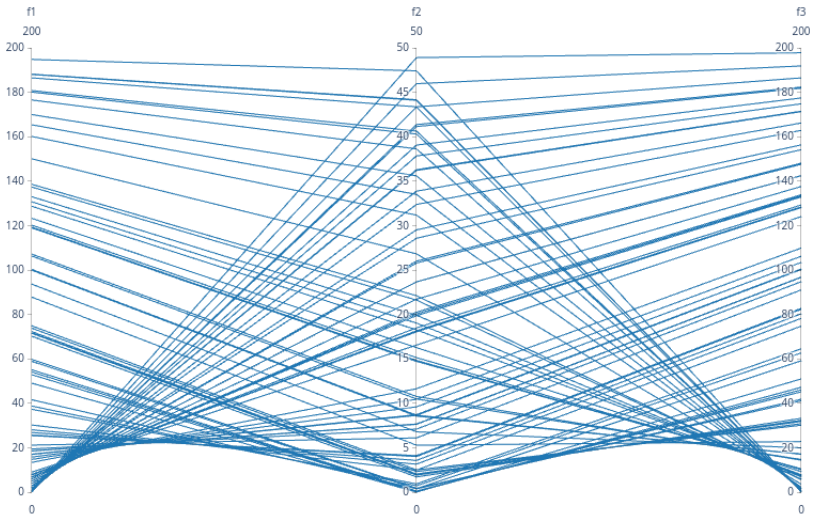


Figure 16 Parallel coordinates chart of a random sample of the Pareto front of the triobjective optimization problem presented in Figure 14. This chart complements the information already shown in Figure 15. There is a rich correlation structure between objectives: Both conflict and agreement among objectives are visible. This pattern closely resembles the features observed in the real-world problem shown in Figure 9.

The idea to discuss the aspect of performance indicators at the Dagstuhl seminar origins in the understanding that no single indicator exists that fulfils all these properties. Moreover, recent work indicates that optimally distributed points generated in the sense of different indicators are not optimal in any case [28]. Since the distribution of points is a critical aspect, the working group started with the idea to use statistical method to generate these points and build a new performance indicator based on the corresponding methodology.

The remainder of the text is organised as follows: The following part briefly summarises the history of performance indicators in multi- and many-objective optimisation. It thus summarises the preliminary work that lead to the one at hand. This is followed by a detailled description of involved methods and the implementation derived during the Dagstuhl seminar. Finally, first result are presented and a short overview on conclusion and future work is provided.

4.4.2 Preliminary Work

In the early days of multi-objective evolutionary algorithms (MOEAs), no performance indicators were adopted to assess performance, and comparisons of results were based purely on graphical representations of the approximations generated by two or more MOEAs. The first performance indicators were proposed in the mid-1990s. Some examples are: Distributed Spacing (ι) [27], Attainment Functions [12] and Efficient Set Spacing [24]. However, it was in the late 1990s when a wide variety of performance indicators were introduced. For example, David Van Veldhuizen [29] proposed: Generational Distance, Error Ratio, Maximum Pareto Front Error, Average Pareto Front Error, Overall Nondominated Vector Generation, Overall Nondominated Vector Generation Ratio and Generational Nondominated Vector Generation. Zitzler et al. [36, 34] proposed: Relative Coverage Comparison of Two Sets, Size of the Space Covered and Hypervolume. Later on, Zitzler also proposed the ϵ-indicator [35].

During this period (late 1990s and early 2000s), there were also several concerns regarding the appropriate methodology to assess the performance of a MOEA (see for example [16]). But a more important issue that soon arose was Pareto compliance. In 2003, Zitzler et al. [37] showed that most of the performance indicators that were in common use at that time were Pareto non-compliant, which meant that their results were unreliable. The hypervolume was identified as the only unary indicator which is Pareto compliant, but its high computational cost when dealing with problems having a high number of objectives triggered a significant amount of research in the last 15 years [31, 2, 23, 15].

Throughout the years, many other performance indicators have been proposed (see for example [10, 11, 6, 26, 18, 3, 7, 25, 20, 14]). Also, a number of surveys on performance indicators are currently available (see for example [21, 5, 22, 17]). Also, some researchers have proposed other interesting ideas such as the use of an ensemble of performance indicators [32].

Although the development of performance measures for assessing convergence and diversity of the approximations generated by a MOEA is a fundamental topic in evolutionary multi-objective optimization, in recent years, there have been few papers focusing on the development of new performance measures. The emphasis in recent years has been on the development of performance measures to assess performance in problems having a large number of objectives. In this case, diversity is of particular interest, and some interesting proposals have been made in that regard (see for example [13, 30]).

While the work of Harding and Saff [13] is more theory driven, there are some recent publications that address the construction of weights vectors [33] and the easy creation of any arbitrary number of uniformly distributed reference points [8]. Finally, [28] analyzes nine quality indicators using their approximated optimal distributions for $p = 3$. The analysis demonstrates that uniformly-distributed objective vectors over the entire Pareto front are not optimal in many cases. Each quality indicator has its own optimal distribution for each Pareto front.

4.4.3 Methods and Implementation

We propose a performance indicator based on reference vectors. The advantages of the latter are manifold: for instance, they (i) allow to incorporate a decision maker's preferences, (ii) are independent from the shape of the true Pareto front, and (iii) are scalable in the number of objectives.

Reference vectors $\mathbf{r} = (w_1, \ldots, w_p) \in [0, 1]^p$ are p-dimensional vectors (in the objective space) and their elements (or weights) w_i indicate how strong the respective vector is affected by each of the p underlying objectives $f_1, \ldots, f_p$. W.l.o.g. we further assume $\sum_{i=1}^{p} w_i = 1$. In most applications, those reference vectors are uniformly distributed across the $(p-1)$-dimensional Pareto front (see, e.g., [4]). However, a variety of research on efficient sampling strategies has shown that evenly spaced structures (like a grid layout in ≥ 2 dimensions) are suboptimal. Therefore, we decided to utilize two sophisticated sampling strategies – *Latin Hypercube Sampling (LHS)* [19] and *Maximally Sparse Selection (MSS)* [9] – for finding promising weight configurations (which in turn will hopefully result in a suitable alignment of the reference vectors across the Pareto front).

In the first strategy, we use LHS to generate a set of K coefficient vectors $\boldsymbol{\theta}^{(k)} = \left(\theta_1^{(k)}, \theta_2^{(k)}, \ldots, \theta_{p-1}^{(k)}\right) \in \left[0, \frac{\pi}{2}\right]^{p-1}$ with $k = 1, \ldots, K$ in the $(p-1)$-dimensional box with lower bounds 0 and upper bounds $\pi/2$. We treat those vectors as angles in the $(p-1)$-dimensional

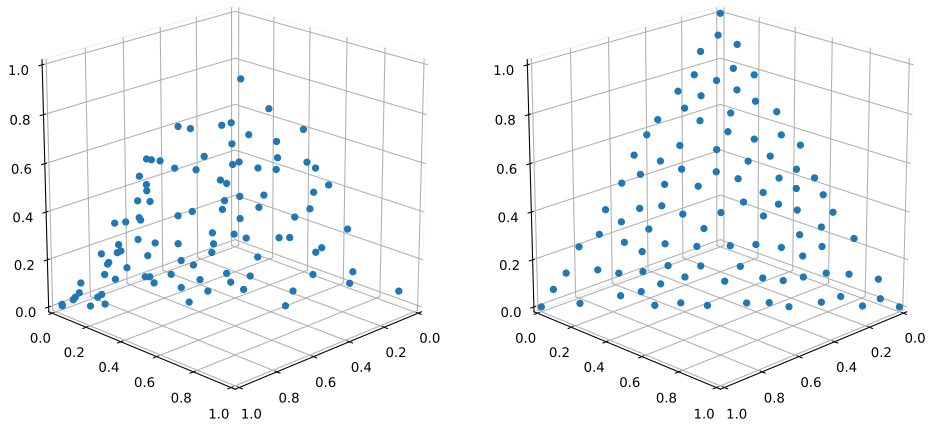


Figure 17 Schematic example of our proposed approach for generating 100 weight vectors in a 3-objective problem. The left image shows the results using LHS, and the right one is based on MSS.

space and use them to compute the corresponding p-dimensional (polar) coordinates $\mathbf{x}^{(k)} = (x_1^{(k)}, \ldots, x_p^{(k)})$ (in the so-called hyperspherical coordinate system [1]):

$$\begin{aligned} x_1^{(k)} &= \cos(\theta_1^{(k)}) \\ x_j^{(k)} &= \cos(\theta_j^{(k)}) \cdot \Pi_{i=1}^{j-1} \sin(\theta_i^{(k)}), \qquad j = 2, \ldots, p-1 \\ x_p^{(k)} &= \Pi_{i=1}^{p-1} \sin(\theta_i^{(k)}). \end{aligned}$$

The resulting vectors are located on the $(p-1)$-dimensional unit sphere and thus contradict the desired property of all weights summing up to one. Thus, we project the coordinates onto the $(p-1)$-dimensional simplex by normalizing the polar coordinates (with the L1-norm of the respective vector's distance to the origin), i.e., $w_i^{(k)} := x_i^{(k)} / \sum_{j=1}^{p} x_j^{(k)}$.

On the other hand, the MSS sampling strategy already starts with points on the $(p-1)$-dimensional simplex. Each of the initial $\overline{K}$ simplex points $\mathbf{s}^{(k)} = \left(s_1^{(k)}, \ldots, s_p^{(k)}\right) \in [0,1]^p$, where $\sum_{i=1}^{p} s_i^{(k)} = 1$ and $k = 1, \ldots, \overline{K}$, is constructed by first sorting $p - -1$ uniformly distributed random numbers $a_i^{(k)} \in [0,1]$, $i = 1, \ldots, p-1$, so that $0 \leq a_1^{(k)} \leq a_2^{(k)} \leq \cdots \leq a_{p-1}^{(k)} \leq 1$. We interpret the numbers 0 and 1 as $a_0^{(k)}$ and $a_p^{(k)}$, respectively. Then, the coordinates of the simplex points are computed as $s_i^{(k)} = a_i^{(k)} - -a_{i-1}^{(k)}$, for $i = 1, \ldots, p$.

The final set of K simplex points, where $\overline{K} \gg K$, is constructed iteratively according to the maximally sparse selection. In a first step, all p unit vectors are added to this set. Then, until the size of the set reaches K, the point $\mathbf{s}^{(k)}$ that has the greatest sum of distances to all already chosen points is added to the set. By increasing the number of initial points $\overline{K}$, the resulting set has a more uniform distribution of points, but it also takes longer to compute.

4.4.4 Results

Figure 17 depicts two exemplary samples of reference vectors using our proposed approach for a 3-objective problem. The left image shows 100 reference vectors generated using the LHS approach, and the right image depicts the 100 reference vectors created using MSS as a sampling strategy.

In the depicted scenario, the distribution of the reference vectors generated by the MSS approach (i.e., right image) looks more promising than the one based on LHS (left). Therefore, we will consider only reference vectors generated by the MSS approach in the further study. Therein, we will use our reference vectors in combination with the R2 indicator (called our R2-MSS indicator in the following) in order to assess the quality of a selection of approximated optimal distribution sets that were obtained in [28]. The assessment will be performed both from a qualitative (visual) and quantitative (indicator-based) perspective.

We will use the approximated optimal distribution sets for nine different performance indicators (HV, IGD, IGD+, R2, NR2, ϵ+, SE, Δ, PD), across MOO problems with six types of Pareto fronts (linear, concave, convex, and their inverted versions), and three, five and eight objectives, generated by Tanabe and Ishibuchi [28]. The performance of these sets will be evaluated by the R2 indicator where 100 reference vectors were chosen by the MSS approach from an initial selection of 1000 random reference vectors.

Tables 2 and 3 show the performance of our R2-MSS indicator that uses 100 reference vectors sampled with the MSS approach (see Section 4.4.3) on the approximated optimal distribution sets from [28]. The columns contain sets of the same shape and the rows those that correspond to the same indicator. For each set we show our R2-MSS indicator value and its rank among all sets of the same shape and dimension. In addition, we visualize the approximated optimal distribution sets for the 3-objective case. We have summed up the ranks for sets optimizing each indicator (across the three considered dimensions and all six shapes) and ordered the sets in the tables from the best (top) to the worst (bottom) indicator performance according to this sum.

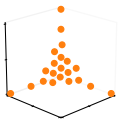
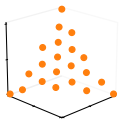
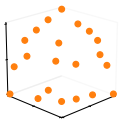
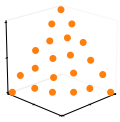
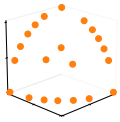
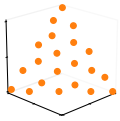
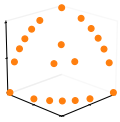
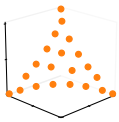
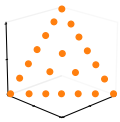
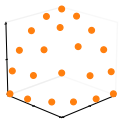
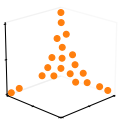
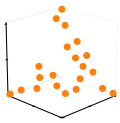
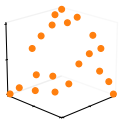
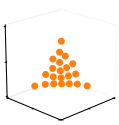
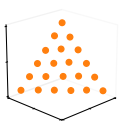
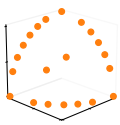
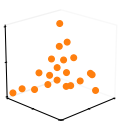
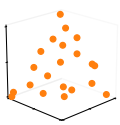
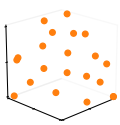
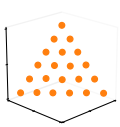
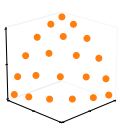
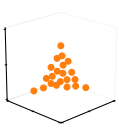
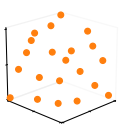
We can see that, unsurprisingly, the best results are almost always achieved on sets optimizing the regular R2 indicator, which are closely followed by those optimizing the NR2 indicator. As indicated by the plots, the sets for the HV indicator resemble those for the NR2 indicator and generally yield similar values. The sets optimizing the SE indicator perform really well on linear shapes and poorly on the others and are therefore scored fourth overall. The sets for indicators Δ, PD and ϵ+ look unevenly distributed and are appropriately given poor scores. On the other hand, sets corresponding to the IGD and IGD+ indicators often have an even distribution of points, which is not in line with their performance as evaluated by our indicator. A further inspection suggests that the main reason for the poor performance of these sets is that they are missing points at the location of the unit vectors on non-inverted fronts (see Table 2), which are always included in our 100 selected reference vectors and are well approximated by some of the other sets. In fact, a short experiment has shown that if we add such extreme points to all sets from Table 2, the order of the indicators changes and sets optimizing the IGD+ and IGD indicators are scored more favourably.

4.4.5 Conclusions and Future Work

First of all we have to admit that the definition of a performance indicator needs much more than a neat idea. All aspects that needed to be considered to finally derive a proper performance indicator was more work and needed more discussion than expected. In addition, more compromises than expected with respect to different aspects like sampling strategy, reference-based or not etc., needed to define such an indicator had to be made.

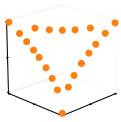
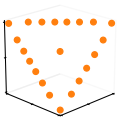
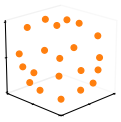
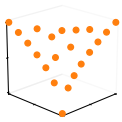
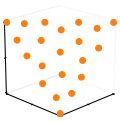
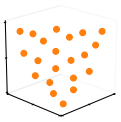
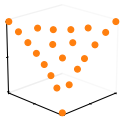
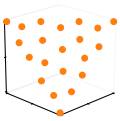
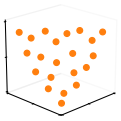
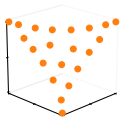
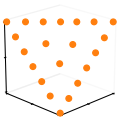
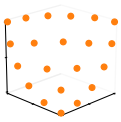
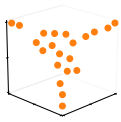
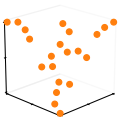
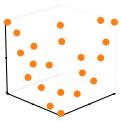
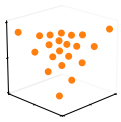
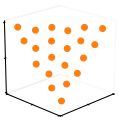
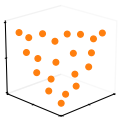
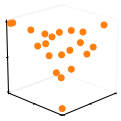
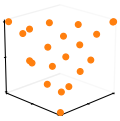
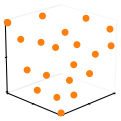
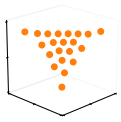
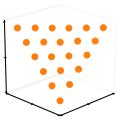
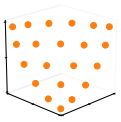
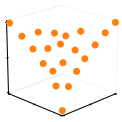
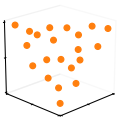
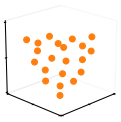
Nevertheless, we managed to define a new indicator based on statistical methods, namely LHS and MSS, to generate equally distributed points in high-dimensional spaces. Two such methods were involved in the implementation of a new indicator and first results supported the decision to continue investigations using MSS. Incorporating this method into an R2 like indicator led to our new indicator R2-MSS. This was tested on already existing approximately optimal distributed sets yielding promising results.

Table 2 The values (and their ranks) of our R2-MSS indicator that uses 100 MSS sampled reference vectors on the approximated optimal distribution sets for indicator I with convex, linear and concave shapes in dimensions $p = 3, 5, 8$ (see textual description for more details).

	convex				linear				concave			
I	$p=3$	p	value	rank	$p=3$	p	value	rank	$p=3$	p	value	rank
R2		3	0.03497	1		3	0.06009	1		3	0.08219	4
		5	0.00814	1		5	0.01559	1		5	0.01872	3
		8	0.00269	2		8	0.00890	2		8	0.00980	3
NR2		3	0.03672	2		3	0.06169	4		3	0.08183	1
		5	0.00849	2		5	0.01807	4		5	0.01882	4
		8	0.00259	1		8	0.00953	3		8	0.01138	4
HV		3	0.03682	3		3	0.06159	3		3	0.08210	3
		5	0.00863	3		5	0.01956	5		5	0.01858	2
		8	0.00323	3		8	0.01002	4		8	0.00835	1
SE		3	0.03907	6		3	0.06186	5		3	0.08230	5
		5	0.01082	7		5	0.01596	2		5	0.01979	5
		8	0.00604	9		8	0.00784	1		8	0.01703	6
Δ		3	0.03803	5		3	0.06121	2		3	0.08329	6
		5	0.01049	6		5	0.01789	3		5	0.02113	6
		8	0.00358	5		8	0.01100	5		8	0.01393	5
IGD+		3	0.04144	8		3	0.07305	8		3	0.08193	2
		5	0.01135	8		5	0.04167	9		5	0.01854	1
		8	0.00368	6		8	0.02460	9		8	0.00843	2
PD		3	0.03718	4		3	0.06262	6		3	0.08926	8
		5	0.00990	4		5	0.02514	6		5	0.04626	7
		8	0.00348	4		8	0.01633	6		8	0.03326	7
IGD		3	0.04048	7		3	0.07284	7		3	0.10066	9
		5	0.01039	5		5	0.03929	7		5	0.06322	9
		8	0.00389	7		8	0.02457	7		8	0.05234	9
$\epsilon+$		3	0.04192	9		3	0.07699	9		3	0.08896	7
		5	0.01268	9		5	0.04149	8		5	0.04834	8
		8	0.00423	8		8	0.02457	7		8	0.03820	8

It seems clear that this approach needs to be examined on different solutions sets, in particular steaming from algorithm runs in the future. Different scenarios need to be investigated using different algorithms, different benchmarking functions yielding different Pareto front shapes, and compared to other performance indicators for different objective space dimensions. Finally, the new indicator should be investigated with respect to all of the

Table 3 The values (and their ranks) of our R2-MSS indicator that uses 100 MSS sampled reference vectors on the approximated optimal distribution sets for indicator I with inverted convex, linear and concave shapes in dimensions $p = 3, 5, 8$ (see textual description for more details).

	inverted convex				inverted linear				inverted concave			
I	$p = 3$	p	value	rank	$p = 3$	p	value	rank	$p = 3$	p	value	rank
R2		3	0.22936	1		3	0.19256	1		3	0.12929	1
		5	0.23385	3		5	0.20172	2		5	0.15192	1
		8	0.23359	2		8	0.21235	2		8	0.16376	1
NR2		3	0.23084	2		3	0.19670	4		3	0.13240	4
		5	0.23140	1		5	0.21995	5		5	0.16292	8
		8	0.24153	4		8	0.26378	7		8	0.17817	7
HV		3	0.23172	3		3	0.19625	3		3	0.13174	3
		5	0.23332	2		5	0.23400	7		5	0.16039	6
		8	0.23175	1		8	0.26654	8		8	0.18751	9
SE		3	0.23285	4		3	0.19450	2		3	0.13281	7
		5	0.23534	4		5	0.20003	1		5	0.15728	5
		8	0.23975	3		8	0.20727	1		8	0.17949	8
Δ		3	0.24822	7		3	0.21063	9		3	0.13613	8
		5	0.29475	7		5	0.21630	3		5	0.16178	7
		8	0.36511	8		8	0.22674	3		8	0.16612	3
IGD+		3	0.24816	6		3	0.20900	8		3	0.13274	5
		5	0.27227	6		5	0.23800	8		5	0.15403	3
		8	0.26911	5		8	0.25465	5		8	0.17334	5
PD		3	0.24967	8		3	0.20092	5		3	0.13280	6
		5	0.35168	9		5	0.23380	6		5	0.15457	4
		8	0.38488	9		8	0.27272	9		8	0.16828	4
IGD		3	0.28437	9		3	0.20827	7		3	0.13146	2
		5	0.33947	8		5	0.24139	9		5	0.15319	2
		8	0.33546	7		8	0.26131	6		8	0.16597	2
$\epsilon+$		3	0.23383	5		3	0.20549	6		3	0.13654	9
		5	0.24714	5		5	0.21832	4		5	0.16309	9
		8	0.27915	6		8	0.22893	4		8	0.17647	6

properties listed above. This investigation is expected to prove which properties are owned by our indicator and which are not. Based on all these results we expect some hints on how to improve our algorithm further and, thus, to hopefully improve performance measurement and comparison in many-objective optimisation.

References

1 L. E. Blumenson. A Derivation of n-Dimensional Spherical Coordinates. *The American Mathematical Monthly*, 67(1):63 – 66, 1960.

2 Lucas Bradstreet, Lyndon While, and Luigi Barone. A Fast Incremental Hypervolume Algorithm. *IEEE Transactions on Evolutionary Computation*, 12(6):714–723, December 2008.

3 Dimo Brockhoff, Tobias Wagner, and Heike Trautmann. On the Properties of the R2 Indicator. In *2012 Genetic and Evolutionary Computation Conference (GECCO'2012)*, pages 465–472, Philadelphia, USA, July 2012. ACM Press. ISBN: 978-1-4503-1177-9.

4 Ran Cheng, Yaochu Jin, Markus Olhofer, and Bernhard Sendhoff. A Reference Vector Guided Evolutionary Algorithm for Many-Objective Optimization. *IEEE Transactions on Evolutionary Computation*, 20(5):773 – 791, 2016.

5 Shi Cheng, Yuhui Shi, and Quande Qin. On the Performance Metrics of Multiobjective Optimization. In Ying Tan, Yuhui Shi, and Zhen Ji, editors, *Advances in Swarm Intelligence, Third International Conference, ICSI 2012*, pages 504–512, Shenzhen, China, June 17-20 2012. Springer. Lecture Notes in Computer Science Vol. 7331.

6 Carlos A. Coello Coello and Nareli Cruz Cortés. Solving Multiobjective Optimization Problems using an Artificial Immune System. *Genetic Programming and Evolvable Machines*, 6(2):163–190, June 2005.

7 Dilip Datta and Jose Rui Figueira. Some convergence-based M-ary cardinal metrics for comparing performances of multi-objective optimizers. *Computers & Operations Research*, 39(7):1754–1762, July 2012.

8 Kalyanmoy Deb, Sunith Bandaru, and Haitham Seada. Generating uniformly distributed points on a unit simplex for evolutionary many-objective optimization. In Kalyanmoy Deb, Erik Goodman, Carlos A. Coello Coello, Kathrin Klamroth, Kaisa Miettinen, Sanaz Mostaghim, and Patrick Reed, editors, *Evolutionary Multi-Criterion Optimization*, pages 179–190, Cham, 2019. Springer International Publishing.

9 Kalyanmoy Deb, Sunith Bandaru, and Haitham Seada. Generating Uniformly Distributed Points on a Unit Simplex for Evolutionary Many-Objective Optimization. In Kalyanmoy Deb, Erik Goodman, Carlos A. Coello Coello, Kathrin Klamroth, Kaisa Miettinen, Sanaz Mostaghim, and Patrick Reed, editors, *Evolutionary Multi-Criterion Optimization*, pages 179 – 190. Springer, 2019.

10 Ali Farhang-Mehr and Shapour Azarm. Diversity Assessment of Pareto Optimal Solution Sets: An Entropy Approach. In *Congress on Evolutionary Computation (CEC'2002)*, volume 1, pages 723–728, Piscataway, New Jersey, May 2002. IEEE Service Center.

11 Ali Farhang-Mehr and Shapour Azarm. Entropy-Based Multi-Objective Genetic Algorithm for Design Optimization. *Structural and Multidisciplinary Optimization*, 24(5):351–361, November 2002.

12 Carlos M. Fonseca and Peter J. Fleming. On the Performance Assessment and Comparison of Stochastic Multiobjective Optimizers. In *Parallel Problem Solving from Nature,äiPPSN IV*, Lecture Notes in Computer Science, pages 584–593, Berlin, Germany, September 1996. Springer-Verlag.

13 Douglas P. Hardin and Edward B. Saff. Discretizing Manifolds via Minimum Energy Points. *Notices of the American Mathematical Society*, 51(10):1186 – 1194, 2004.

14 Hisao Ishibuchi, Hiroyuki Masuda, Yuki Tanigaki, and Yusuke Nojima. Modified Distance Calculation in Generational Distance and Inverted Generational Distance. In António Gaspar-Cunha, Carlos Henggeler Antunes, and Carlos A. Coello Coello, editors, *Evolutionary Multi-Criterion Optimization, 8th International Conference, EMO 2015*, pages 110–125. Springer. Lecture Notes in Computer Science Vol. 9019, Guimarães, Portugal, March 29 – April 1 2015.

15 Andrzej Jaszkiewicz. Improved Quick Hypervolume Algorithm. *Computers & Operations Research*, 90:72–83, February 2018.

16 Joshua Knowles, Lothar Thiele, and Eckart Zitzler. A Tutorial on the Performance Assessment of Stochastic Multiobjective Optimizers. 214, Computer Engineering and Networks Laboratory (TIK), ETH Zurich, Switzerland, 2006. revised version.

17 Miqin Li and Xin Yao. Quality Evaluation of Solution Sets in Multiobjective Optimisation: A Survey. *ACM Computing Surveys*, 52(2), May 2019. Article number: 26.

18 Giovanni Lizárraga Lizárraga, Arturo Hernández Aguirre, and Salvador Botello Rionda. G-Metric: an M-ary Quality Indicator for the Evaluation of Non-dominated Sets. In *2008 Genetic and Evolutionary Computation Conference (GECCO'2008)*, pages 665–672, Atlanta, USA, July 2008. ACM Press. ISBN 978-1-60558-131-6.

19 Michael D. McKay, Richard J. Beckman, and William J. Conover. A Comparison of Three Methods for Selecting Values of Input Variables in the Analysis of Output from a Computer Code. *Technometrics*, 42(1):55 – 61, 2000.

20 Seyedali Mirjalili and Andrew Lewis. Novel Performance Metrics for Robust Multi-Objective Optimization Algorithms. *Swarm and Evolutionary Computation*, 21:1–23, April 2015.

21 Tatsuya Okabe, Yaochu Jin, and Bernhard Sendhoff. A Critical Survey of Performance Indices for Multi-Objective Optimization. In *Proceedings of the 2003 Congress on Evolutionary Computation (CEC'2003)*, volume 2, pages 878–885, Canberra, Australia, December 2003. IEEE Press.

22 Nery Riquelme, Christian Von Lücken, and Benjamin Baran. Performance Metrics in Multi-Objective Optimization. In *2015 Latin American Computing Conference (CLEI)*, pages 1 – 11. IEEE Press, October 2015.

23 Luis M. S. Russo and Alexandre P. Francisco. Quick Hypervolume. *IEEE Transactions on Evolutionary Computation*, 18(4):481–502, August 2014.

24 Jason R. Schott. Fault Tolerant Design Using Single and Multicriteria Genetic Algorithm Optimization. Master's thesis, Department of Aeronautics and Astronautics, Massachusetts Institute of Technology, Cambridge, Massachusetts, May 1995.

25 Oliver Schütze, Xavier Esquivel, Adriana Lara, and Carlos A. Coello Coello. Using the Averaged Hausdorff Distance as a Performance Measure in Evolutionary Multiobjective Optimization. *IEEE Transactions on Evolutionary Computation*, 16(4):504–522, August 2012.

26 Vinícius L. S. Silva, Elizabeth F. Wanner, Sérgio A. A. G. Cerqueira, and Ricardo H. C. Takahashi. A New Performance Metric for Multiobjective Optimization: The Integrated Sphere Counting. In *2007 IEEE Congress on Evolutionary Computation (CEC'2007)*, pages 3625 – 3630, Singapore, September 2007. IEEE Press.

27 N. Srinivas and Kalyanmoy Deb. Multiobjective Optimization Using Nondominated Sorting in Genetic Algorithms. *Evolutionary Computation*, 2(3):221–248, Fall 1994.

28 Ryoji Tanabe and Hisao Ishibuchi. An Analysis of Quality Indicators Using Approximated Optimal Distributions in a Three-dimensional Objective Space. *IEEE Transactions on Evolutionary Computation*, pages 1–15, 2020.

29 David A. Van Veldhuizen. *Multiobjective Evolutionary Algorithms: Classifications, Analyses, and New Innovations.* PhD thesis, Department of Electrical and Computer Engineering. Graduate School of Engineering. Air Force Institute of Technology, Wright-Patterson AFB, Ohio, USA, May 1999.

30 Handing Wang, Yaochu Jin, and Xin Yao. Diversity Assessment in Many-Objective Optimization. *IEEE Transactions on Cybernetics*, 47(6):1510–1522, June 2017.

31 Lyndon While, Phil Hingston, Luigi Barone, and Simon Huband. A Faster Algorithm for Calculating Hypervolume. *IEEE Transactions on Evolutionary Computation*, 10(1):29–38, February 2006.

32 Gary G. Yen and Zhenan He. Performance Metric Ensemble for Multiobjective Evolutionary Algorithms. *IEEE Transactions on Evolutionary Computation*, 18(1):131–144, February 2014.

33 Saul Zapotecas-Martínez, Hernán E. Aguirre, Kiyoshi Tanaka, and Carlos A. Coello Coello. On the Low-Discrepancy Sequences and Their Use in MOEA/D for High-Dimensional Objective Spaces. In *2015 IEEE Congress on Evolutionary Computation (CEC)*, pages 2835–2842, May 2015.

34 Eckart Zitzler, Kalyanmoy Deb, and Lothar Thiele. Comparison of Multiobjective Evolutionary Algorithms: Empirical Results. *Evolutionary Computation*, 8(2):173–195, 2000.

35 Eckart Zitzler, Marco Laumanns, and Stefan Bleuler. A Tutorial on Evolutionary Multiobjective Optimization. In Xavier Gandibleux, Marc Sevaux, Kenneth Sörensen, and Vincent T'kindt, editors, *Metaheuristics for Multiobjective Optimisation*, Lecture Notes in Economics and Mathematical Systems, pages 3 – 37, Berlin, 2004. Springer.

36 Eckart Zitzler and Lothar Thiele. Multiobjective Evolutionary Algorithms: A Comparative Case Study and the Strength Pareto Approach. *IEEE Transactions on Evolutionary Computation*, 3(4):257–271, November 1999.

37 Eckart Zitzler, Lothar Thiele, Marco Laumanns, Carlos M. Fonseca, and Viviane Grunert da Fonseca. Performance Assessment of Multiobjective Optimizers: An Analysis and Review. *IEEE Transactions on Evolutionary Computation*, 7(2):117–132, April 2003.

4.5 KaKaRaKe – User-Friendly Visualization for Multiobjective Optimization with High-Dimensional Objective Vectors

Kerstin Dächert (Fraunhofer ITWM – Kaiserslautern, DE), Kathrin Klamroth (Universität Wuppertal, DE), Kaisa Miettinen (University of Jyväskylä, FI), and Ralph E. Steuer (University of Georgia, US)

4.5.1 Introduction

Multiobjective optimization problems involve multiple conflicting objective functions to be optimized simultaneously. When solving such problems, we generate Pareto optimal solutions reflecting different trade-offs among the objectives. Typically, some type of preference information coming from a decision maker, a domain expert, is needed to identify the most preferred Pareto optimal solution as the final one.

By studying different Pareto optimal solutions and providing preference information, the decision maker can learn about interdependencies among the objectives. This task can be supported by visualizations. Visualizations can also be used to represent the progress of the solution process.

Surveys of various visualization techniques to represent a set of Pareto optimal objective vectors include [9, 12, 13, 18]. They discuss widely-used visualization techniques like parallel coordinate plots (sometimes known as value paths), spider web charts, petal diagrams, star coordinate plots, and glyphs. Examples of more recently proposed visualization methods for multiobjective optimization are heatmaps [6], knowCube [16], interactive decision maps [11], the prosection method [17], and 3d-radvis [7]. Visualization aspects in multiobjective optimization are also discussed in [5].

Questions involved in developing and applying visualizations include how to scale visualizations with increasing numbers of objectives; how to support the decision maker to gain understanding of the progress of the solution process; how to conduct "sensitivity analysis" by which the decision maker can understand the consequences of actions; can visualizations point to directions where small sacrifices in some objective can provide good improvement in some other objectives; how to find solutions with "robustness" properties; and how to communicate this to the decision maker. Overall, the decision maker should be able to affect the appearance of the visualization.

Our motivation is to develop visualizations to provide support to decision makers by identifying interesting aspects of Pareto optimal objective vectors as solution candidates and of the solution process itself. We aim at handling larger numbers of objectives (say between five and nine) in a comprehensible way without cognitively burdening the DM too much.

When using traditional visualization techniques, e.g., radar plots or parallel coordinates plots, the decision maker can conveniently grasp important interrelationships among the objectives when they are placed close to each other yet finding the best order in which to present the objectives may require some trial and error. Our goal is to assist the decision maker in identifying aspects of interest by pre-processing the set of solution candidates. We have two goals: to detect which objectives are strongly correlated or uncorrelated with one another, and to reduce the number of solutions presented when desirable by detecting similarities among them. The latter can be understood as finding good representative solutions.

One can detect correlations among objectives, for example, to reduce the computational burden by decreasing the number of objectives (see, e.g. [1, 2]). However, we are not aware of any such approach that is applied in a combined way to reduce the visual effort of the decision maker by clustering both objectives and solution candidates simultaneously.

We propose a new visualization technique by applying bi- or co-clustering to the set of Pareto optimal objective vectors. In this way, we can simultaneously visualize similarities and differences among objective functions and solution candidates. To communicate the information visually, we modify the idea of parallel coordinate plots so that the distances between correlated objective functions are shorter than otherwise and similar solutions are given the same color. Thanks to this kind of visualization, the decision maker can handle higher numbers of objectives and solution candidates and concentrate on aspects that are of greatest interest.

As for the following, in Section 4.5.2 we describe how data for a study like this can be obtained. In Section 4.5.3 we survey clustering techniques applicable to our needs, and in Section 4.5.4 the new visualization technique of this study is proposed. Finally, we conclude in Section 4.5.5.

4.5.2 Data generation

As mentioned above, we want to support the decision maker in handling larger numbers of objectives than is usually the case and by this we mean problems with 5 or more objectives. Of course, if one thinks long enough, one can probably imagine problems with almost any number of objectives, but because of the almost total lack of work in the area, we will in this paper only focus on problems with up to 9 objectives to get things started. However, there is a kind of "Catch-22" with problems in this area. On one hand, people are reluctant to attempt applications with many more than 5 objectives as there are essentially no tools for processing points from such high-dimensional Pareto fronts, and on the other hand, people have been slow to begin work on tools for processing high-dimensional solution vectors because of the lack of data from problems upon which to test such tools.

To get around this we describe two methods for randomly generating nondominated vectors in objective space. The first method is for the generation of Pareto optimal solutions as if coming from a problem in which all of the objectives have no especial correlations with one another. This method uses the random multiple objective linear program (MOLP) generator described in the documentation for Adbase [15] and then uses that code to solve the resulting MOLPs for all nondominated vertices of the problem's feasible region in the objective space. Data sets of any size for any number of objectives can be generated in this way by adjusting the parameters (numbers of objectives, constraints, variables) of the generator. Once a data set of Pareto solution vectors is generated, it is immaterial how it was generated for testing.

The second method is for the generation of Pareto solutions as if coming from a problem in which there are groups of objectives that have within group correlations with one another. For instance, consider a problem with 9 objectives such that 4 of the objectives are clustered in one group, 3 are in another, and the last two are in a third. In this method, the feasible region of a problem is generated in the same way as the first method, but instead of letting the MOLP generator generate the gradients of the objectives, the gradients of the objectives are randomly generated by a special routine that assures the clusters desired and their within group correlations. In this way, we can develop as many data sets of the types desired as needed. As for the example of this report, only the second method was necessary to generate data for it.

4.5.3 Clustering techniques

We now assume a given two-dimensional data set where each row corresponds to a Pareto optimal objective vector (solution candidate) and each column corresponds to an objective function. Our aim is to aggregate the given data matrix into solution-objective clusters according to the following rules.

1. Objectives are clustered if the values in any solution are "similar".
2. Solutions are clustered if the values in any objective are "similar".

"Similar" means as close as possible with respect to a specified distance, e.g., a Manhattan or Euclidean distance. The clustering is to be done simultaneously in both dimensions, rows and columns.

In the literature, there exist the concepts of biclustering and co-clustering which seem to denote the same, but are used in different communities and/or applications. A helpful survey is presented in [10].

Biclustering was first applied to bioinformatics, in particular to identify co-expressed genes under a subset of all conditions/samples, see [19] for a recent overview. A mathematical review of successful biclustering techniques is given in [3]. The authors particularly show that most of them are based on singular value decomposition (SVD). However, algorithms differ in the definition of biclusters. Some assume constant values in the data on rows within one cluster, some on columns, some on both. Others are more flexible and allow coherent values along rows and/or columns.

There are also "spectral" variants, i.e., "spectral biclustering" and "spectral co-clustering", that make use of the underlying graph structure of the problem. Spectral algorithms are common in graph partitioning problems and refer to algorithms that compute eigenvalues, eigenvectors, and singular values to solve the underlying graph problem, see e.g., the lecture notes https://courses.cs.washington.edu/courses/cse521/16sp/521-lecture-11.pdf. There, we

also find an illustrative example in which common clustering methods like k-means fail while a spectral algorithm detects a meaningful cluster. In [4], a spectral co-clustering algorithm is proposed for a text mining problem.

In the next section, we use a Python implementation of the biclustering algorithm of [8]. One should note that even though the terms bi- and co-clustering are often regarded as synonyms, they refer to different algorithms in the scikit-learn package employed. More details are given below.

4.5.4 Preliminary results

We present tentative numerical results for a data set containing 88 mutually nondominated objective vectors with nine objective functions. The data were generated by the second method described in Section 4.5.2, aiming at three objective clusters. The tests were implemented in Python 3.7 using the scikit-learn package for data clustering [14] and plotly.express to generate the parallel coordinate plots. The data were clustered w.r.t. objective vectors (rows) and objective functions (columns) using spectral biclustering [8]. This includes an automatic data normalization which was implemented as log-normalization, see the documentation on https://scikit-learn.org/stable/modules/generated/sklearn.cluster.SpectralBiclustering.html for further details.

The original, unclustered data are shown in a parallel coordinates plot in Figure 18a, while Figures 18b to 18d show the results obtained with spectral biclustering using log-normalization for different numbers of solution clusters (S) and objective clusters (O). Each solution cluster is identified by a specific color, and the objective clusters are distinguished by larger distances between the coordinate lines of the different clusters.

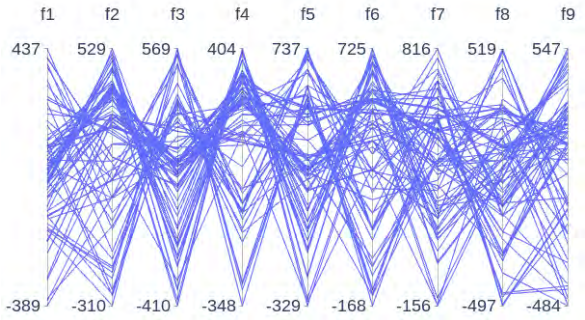
The visualizations nicely demonstrate that biclustering is a valuable tool to reveal tendencies among the solutions and correlations between the objective functions. For appropriately chosen values of S and O, the three objective clusters that are present in the data are retrieved. However, we note that the outcome largely depends on the parameter settings, in particular on the choices of S and O, but also on the employed normalization method.

In the future, we would like to use a measure to evaluate the quality of the visualization. Moreover, it would be interesting to automatically test different sizes of row and column clusters and present the "best" (with respect to a quality measure) result or results to the decision maker. Another open question is the way the columns are ordered. So far, we simply use the output of the biclustering algorithm. In the figures presented above we manually inserted gaps between the different clusters. By varying the sizes of these gaps, they could also serve as a source of information for the decision maker, e.g. by linking larger distances to a lower correlations between clusters. This and other technical issues like an automation are left for further improvements in the future.

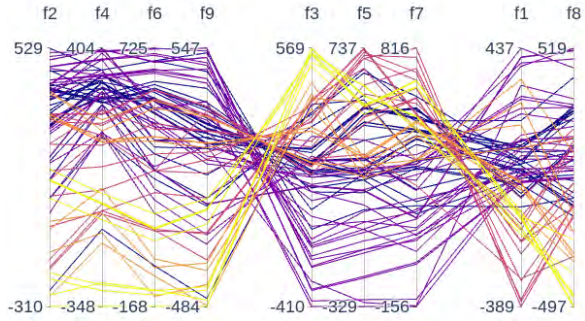
4.5.5 Conclusions

We have proposed a novel way of visualizing sets of Pareto optimal objective vectors by applying bi- or co-clustering and modifying parallel coordinate plots. Thanks to these visualizations, the decision maker can gain insight in the correlations among both objective functions and objective vectors simultaneously.

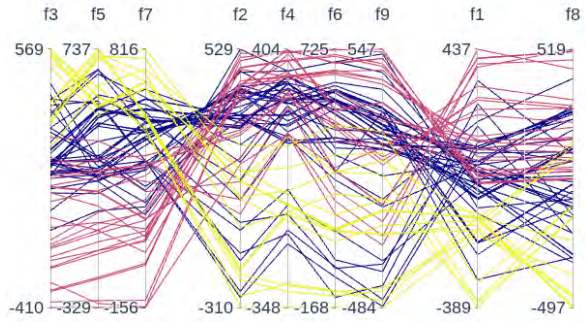
The novel visualizations can be applied to analyze any set of objective vectors. They can also be applied as a part of an interactive solution process. Our future research direction is to apply the findings and develop visualization assistance for solving multiobjective optimization problems with more than three objective functions.



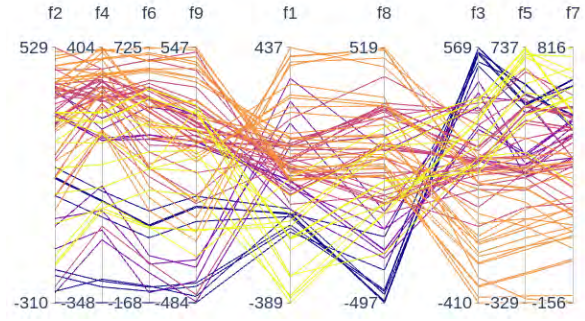
(a) Original data set



(b) BC with $S = 5$ and $O = 3$



(c) BC with $S = 3$ and $O = 4$



(d) BC with $S = 5$ and $O = 4$

Figure 18 Spectral biclustering applied to a data set containing 88 nondominated points (a). Subfigures (b)-(d) show biclustering results for S solution clusters and O objective clusters.

Acknowledgements

We would like to warmly thank Bhupinder Singh Saini for his valuable implementation support. This research is related to the thematic research area DEMO (Decision Analytics utilizing Causal Models and Multiobjective Optimization, jyu.fi/demo) of the University of Jyvaskyla.

References

1 D. Brockhoff and E. Zitzler. Are all objectives necessary? On dimensionality reduction in evolutionary multiobjective optimization. In T. P. Runarsson, H. G. Beyer, E. Burke, J. J. Merelo-Guervós, L. D. Whitley, and X. Yao, editors, *Parallel Problem Solving from Nature, PPSN IX, Proceedings*, pages 533–542, Berlin, Heidelberg, 2006. Springer.

2 D. Brockhoff and E. Zitzler. Dimensionality reduction in multiobjective optimization: The minimum objective subset problem. In K. H. Waldmann and U. M. Stocker, editors, *Operations Research Proceedings 2006*, pages 423–429, Berlin, Heidelberg, 2007. Springer.

3 S. Busygin, O. Prokopyev, and P. M. Pardalos. Biclustering in data mining. *Computers & Operations Research*, 35(9):2964–2987, 2008.

4 I. S. Dhillon. Co-clustering documents and words using bipartite spectral graph partitioning. In *Proceedings of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 269–274, 2001.

5 C. M. Fonseca, C. A. Antunes, R. Lacour, K. Miettinen, P. M. Reed, and T. Tusar. Visualization in multiobjective optimization. In S. Greco, K. Klamroth, J. D. Knowles, and G. Rudolph, editors, *Understanding Complexity in Multiobjective Optimization, Report from Dagstuhl Seminar 15031*, pages 129–139, Dagstuhl, 2015.

6 J. Hettenhausen, A. Lewis, and S. Mostaghim. Interactive multi-objective particle swarm optimization with heatmap-visualization-based user interface. *Engineering Optimization*, 42(2):119–139, 2010.

7 A. Ibrahim, S. Rahnamayan, M. V. Martin, and K. Deb. 3d-radvis: Visualization of Pareto front in many-objective optimization. In *2016 IEEE Congress on Evolutionary Computation (CEC)*, pages 736–745. IEEE, 2016.

8 Y. Kluger, R. Basri, J. T. Chang, and M. Gerstein. Spectral biclustering of microarray data: Coclustering genes and conditions. *Genome Research*, 13:703–716, 2003.

9 P. Korhonen and J. Wallenius. Visualization in the multiple objective decision-making framework. In J. Branke, K. Deb, K. Miettinen, and R. Slowinski, editors, *Multiobjective Optimization: Interactive and Evolutionary Approaches*, pages 195–212. Springer, Berlin, 2008.

10 H.-P. Kriegel, P. Kröger, and A. Zimek. Clustering high dimensional data: A survey on subspace clustering, pattern-based clustering, and correlation clustering. *ACM Transactions on Knowledge Discovery from Data*, 3(1):1–58, 2009.

11 A. V. Lotov, V. A. Bushenkov, and G. K. Kamenev. *Interactive Decision Maps: Approximation and Visualization of Pareto Frontier*. Kluwer Academic Publishers, Boston, 2004.

12 A. V. Lotov and K. Miettinen. Visualizing the Pareto frontier. In J. Branke, K. Deb, K. Miettinen, and R. Slowinski, editors, *Multiobjective Optimization: Interactive and Evolutionary Approaches*, pages 213–243. Springer, Berlin, 2008.

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

14 F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

15 R. E. Steuer. "ADBASE: A multiple objective linear programming solver for all efficient extreme points and unbounded efficient edges," Version 17.1, Terry College of Business, University of Georgia, Athens, Georgia, 2017.
16 H. L. Trinkaus and T. Hanne. knowCube: A visual and interactive support for multicriteria decision making. *Computers and Operations Research*, 32(5):1289–1309, 2005.
17 T. Tusar and B. Filipic. Visualization of Pareto front approximations in evolutionary multiobjective optimization: A critical review and the prosection method. *IEEE Transactions on Evolutionary Computation*, 19(2):225–245, 2015.
18 M. J. Woodruff, P. M. Reed, and T. W. Simpson. Many objective visual analytics: Rethinking the design of complex engineered systems. *Structural and Multidisciplinary Optimization*, 48(1):201–219, 2013.
19 J. Xie, A. Ma, A. Fennell, Q. Ma, and J. Zhao. It is time to apply biclustering: A comprehensive review of biclustering applications in biological and biomedical data. *Briefings in Bioinformatics*, 20(4):1450–1465, 2018.

4.6 Supporting Problem Solving with Many Decision Makers in Multi-objective Optimization

Michael Emmerich (Leiden University, NL), Jussi Hakanen (University of Jyväskylä, FI), Patrick Reed (Cornell University, US), Pradyumn Kumar Shukla (KIT – Karlsruher Institut fürTechnologie, DE), Jürgen Branke (University of Warwick, GB), Günter Rudolph (TU Dortmund, DE), Sanaz Mostaghim (Universität Magdeburg, DE), Heike Trautmann (Universität Münster, DE)

4.6.1 Introduction

The problem of multiobjective optimization changes qualitatively as soon as many decision makers are involved. All problems that occur when a single decision maker is involved are inherited, but new problem aspects are added. Three main additional aspects are:

1. The different decision makers can differ on the constraints and objective functions that are relevant for the problem, and the way they are computed.
2. They may have different preferences for the different objective functions and this way the problem of fairness arises, that is the problem of considering different preferences in a balanced manner.
3. There is a possibility of negotiations and group dynamics that should be considered in designing decision making processes. Moreover, decision makers might form coalitions and there might be different types of (power) relations between decision makers and hidden objectives/agendas.

In this report we consider the somewhat ideal situation of a group of equal decision makers that are able and willing to express their preferences. In such cases computer systems can be used to find out solutions that are non-dominated with respect to all objectives considered by the decision makers and among them present solutions that achieve a high performance in fairness on the one side, and total gain in terms as being close in average to the decision makers' preferred solutions or reference points on the other side.

4.6.2 Problem formulation

General decision making problem

There are three main steps for finding a consensus solution when multiple decision makers are involved:

1. Agreement on the model/goal formulation(s)
2. Selecting a subset of interesting solutions from a large set of alternatives, consensus on solvers
3. Selecting a single solution in mediated negotiation

To start with, all the DMs need to agree on the optimization problem formulation which is not a trivial task as such since different DMs may have different opinions on what is important to take into account. Therefore, this first step is usually an iterative process where alternative formulations are considered and some illustrative results are then shown to the DMs in a facilitated manner. One alternative approach to consider here is the *value focused thinking* approach by Prof. Ralph Keeney [3]. There the idea is to start from the values that the DM actually cares and, then, identify suitable criteria to measure them and define their relative importance. In case of multiple decision makers in all of these steps consensus needs to be achieved among DMs, making the value focused thinking process particularly challenging. In this report, we will not consider the first step in more detail.

When the problem formulation has been agreed, the aim of the second step is to provide solution candidates for the DMs based on their preferences. There exist different ways of providing preferences but, in this report for simplicity, we assume that every DM specifies preferences in the form of a reference point consisting of aspiration levels for each objective. In the case of multiple DMs, it can be assumed that the reference points can vary a lot meaning that the preferences have clear conflicts. In such a case, it is not clear what kind of solution candidates should be presented to the DMs. We feel that this step can be supported by computational tools and that will be our focus in this report. Our idea is based on using measures for fairness and gain when evaluating the solution candidates.

Keeping in mind that the overall goal is to find a single solution for the problem such that all the DMs can agree to that, the last step requires also a facilitated process. Finding the consensus can be difficult especially in cases where the DMs have very conflicting preferences.

Fairness and gain measures

As opposed to multi-objective optimization with a single DM, in the case of many-decision makers a solution can be unfair in the sense that it is closer to one decision maker's preference than to that of another decision maker.

But how can we measure the preference of a decision maker? Examples of preference modeling techniques are

- the specification of a preference by reference points, that are ideal points for each of the decision makers. There apply certain constraints of setting such ideal points. For instance, in the examples in this paper (*NSGA-II for teams*) we suggest to choose the reference point from the Pareto front approximation. It may also be possible to allow arbitrary reference points that our approach would then map onto the Pareto front approximation.
- the assessment of preferences by means of achievement scalarization functions [5]
- the definition of *desirability functions* for each objective and decision maker and aggregating *desirablility indices*, that are products of desirability functions over all objectives considered by the decision makers (cf. [2]).

In the following we illustrate fairness and gain measures by a relative straightforward, yet useful approach of preference formulation. The approach can be transferred easily to other, potentially more sophisticated means of preference modelling.

The basis of modeling gain and fairness can be the notion of losses with respect to the ideal solution a DM can obtain. This loss is also termed *Pareto regret* ([4]). In the case that we define the loss by means of distance to reference vectors the Pareto regret can be computed as:

$$PR_j(x) = \sum_{i=1}^{m} (f_i(x) - r_{ij})$$

where j is the index of the decision maker, $j = 1, ..., d$, m is the number of objectives, and x is the final solution selected. In the following we will introduce some new definitions that are related to the notion of Pareto regret w.r.t. reference points which we have described above. These notions generalise easily to other notions of Pareto regret.

▶ **Definition 1.** Pareto Regret, Average Pareto Regret, Inequality in Pareto Regret Let PR_i denote the Pareto regret for each decision maker. Then the *average Pareto regret* of a solution x is defined as the average of the Pareto regrets of all DMs, i.e.,

$$\mathrm{APR}(x) = \frac{1}{d} \sum_{i=1}^{d} \mathrm{PR}_i(x)$$

and the *Inequality in Pareto Regret* (IPR) of a solution x is defined as the Gini index[2] of the Pareto regret.

$$\mathrm{IPR}(x) = \sum_{i=1}^{d} |\mathrm{PR}_i(x) - \mathrm{APR}(x)|$$

In the following, we will often denote the APR as *gain*, and the IPR as *fairness*.[3]

4.6.3 Matlab code

In order to illustrate our ideas, we wrote a small Matlab script. An example having two objectives is used which means that the objective space and the fairness-gain space have the same dimensionality. The reason for this choice is the ease of visualization and interpretation but the ideas work for problems with more objectives. Further, we have three DMs in our example but, again, the ideas work in case we have more.

The code produces plots in two different spaces: the original objective space and the fairness-gain space. We use a mesh grid to discretize the objective space and we then compute values for fairness and gain in all the points. In the objective space, we visualize the level curves of both the fairness and gain to have some idea of their behaviour with respect to each other. In addition, the reference points of all the DMs are shown.

[2] the average absolute deviation from the mean

[3] Mind that, following the convention in mathematical programming, we aim to minimise both objectives which strictly speaking would require using the more abstract terms IPR and APR. From that perspective fairness and gain are thus improved by means of minimisation.

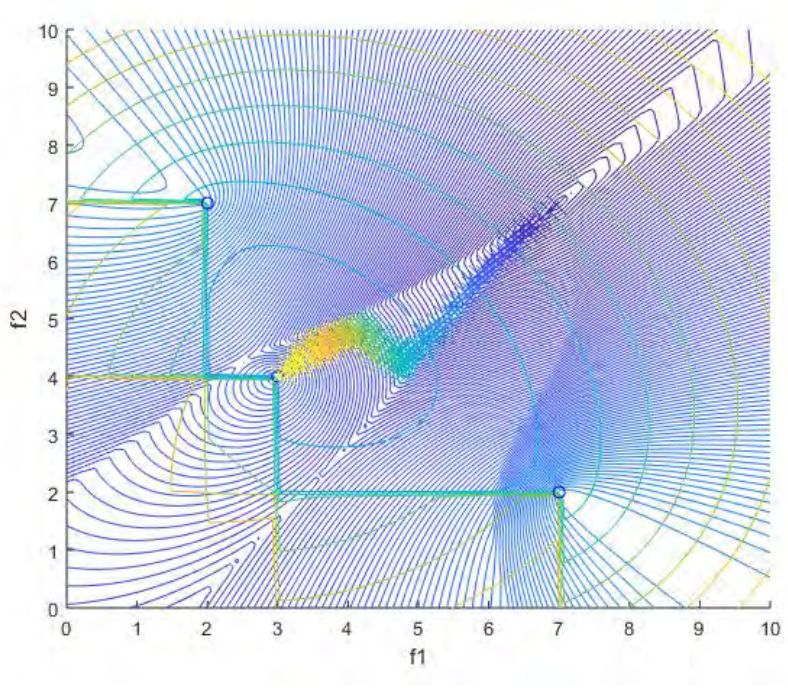


Figure 19 Reference points for DMs (blue circles) and level curves for fairness (denser) and gain (sparse) in the objective space.

Then, we do non-dominated sorting[1] in the fairness-gain space and obtain set of non-dominated points. We visualize these points in both the fairness-gain space and the original objective space. In order to be able to identify same non-dominated points in different plots, we color code them according to increasing values of fairness. An example of the visualizations in the objective space and the fairness-gain space are shown in Figures 19 and 20, respectively.

Figures 21 and 22 show the same trade-offs but this time for a particular example problem where there is a Pareto-optimal front that cannot be improved upon in the objective space (Figure 21. As can be seen, the solutions that belong to the Pareto front in the objective space form a rather peculiar line in the Gain/Fairness space and vice versa. We would be looking for solutions that perform well in both spaces as good compromise solutions.

4.6.4 NSGA-II for teams

It seems relatively straightforward to integrate the above ideas into the NSGA-II framework and focus the search towards solutions that are Pareto-optimal in the objective space, but also Pareto-optimal in the Fairness/Gain space. The proposed NSGA-II for teams algorithm first partitions a given set of solutions (population) into different layers of equal dominance ranks in the objective space. Then, for solutions of the same dominance rank, a non-dominated sorting procedure is executed w.r.t. the objectives fairness and gain.

This is the basic idea for the construction of the proposed algorithm termed *NSGA II for teams*. We leave the discussion of its performance, parameter, scalability studies to the future work and it shall also be remarked that NSGA-II can probably be replaced by other Pareto-based evolutionary multi-objective optimziation algorithms.

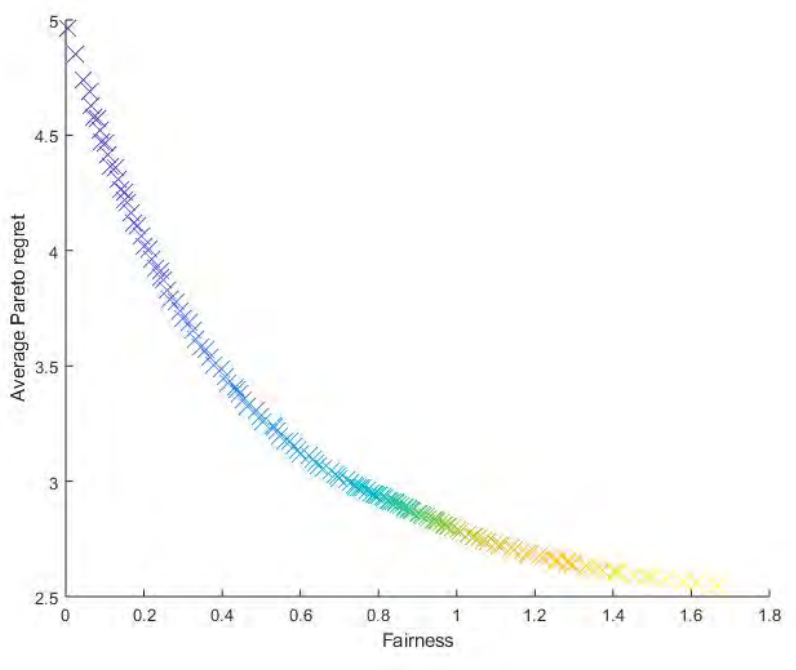


Figure 20 Non-dominated points in the fairness-gain space.

In summary our proposed idea of NSGA-II for teams proposes a straightforward way to integrate an important aspect of problems with many decision makers into EMOA frameworks, namely the trade-off between fairness and gain.

4.6.5 Relation to constraint satisfaction

This part of the report focuses on another aspect that distinguishes problems with many DMs those with a single DM which is the problem of constraint satisfaction in terms of conflicting views of the problem. It shall be noted that constraints and objective functions can coincide, if there is a minimum threshold for an objective function to be met for a solution to be feasible.

Let there be $d \in \mathbb{N}$ decision makers trying to arrive at a solution to an $m \in \mathbb{N}$ objective optimization problem. The (vector-valued) objective function $f(x) := (f_1(x), \ldots, f_m(x))$ is assumed to be a function from a non-empty set $X \subseteq \mathbb{R}^n$, where $n \in \mathbb{N}$ is the dimensionality of the search space. The function $f(x)$ is assumed to be the same for all decision makers (this assumption could, however, be relaxed). They may have their own set of constraints, or have a common set. We assume that decision-maker i has a constraint set $X_i \subseteq \mathbb{R}^n$, for all $i \in \{1, \ldots, d\}$. In the general form, the sets X_i's are can be assumed to be defined using k_i inequality and ℓ_i equality constrains in the following way:

$$X_i = \{x \in \mathbb{R}^n | g_1^i(x) \leq 0, \ldots, g_{k_i}^i(x) \leq 0, h_1^i(x) = 0, \ldots, h_{\ell_i}^i(x) = 0\}. \tag{2}$$

All the functions used in the description (2) above are assumed to be known.

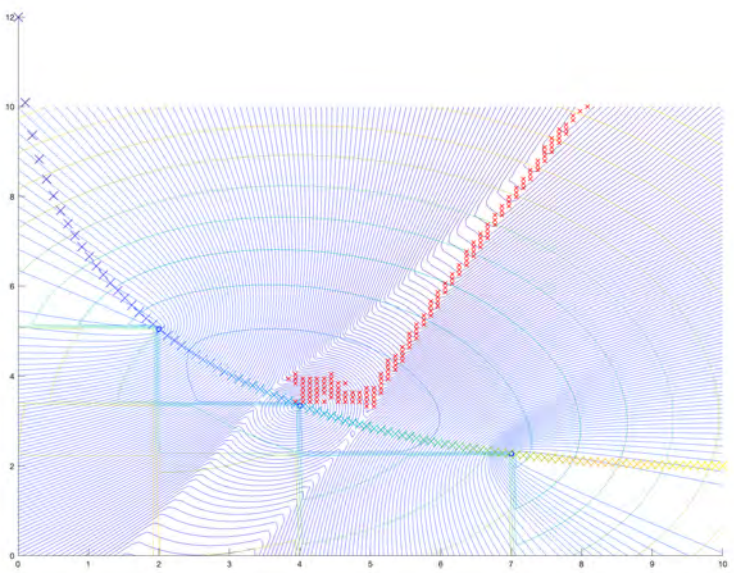


Figure 21 The coloured crosses are the Pareto front approximation in objective space, the red crosses are the solutions that form the Pareto front approximation in the Gain/Fairness space.

Detecting intra-inconsistencies within the constraint sets

The aim here is to detect whether the constraints X_i, defined by (2), are consistent or not. This would be done for all the decision-makers. To do this, for every decision-maker i, we consider the following optimization problem:

$$\begin{aligned} \max_{(u,x)\in\mathbb{B}^{k_i+\ell_i}\times X} & \sum_{i=1}^{k_i+\ell_i} u_i \\ \text{s.t.} \quad & u_j g_j^i(x) \leq 0,\ \forall j = 1,\dots,k_i, \\ & u_l h_j^l(x) = 0,\ \forall l = 1,\dots,\ell_i, \end{aligned} \tag{Opt(i)}$$

where $\mathbb{B}$ denotes the set $\{0,1\}$.

▶ **Proposition 2.** *The constraint set of Opt(i) is non-empty.*

Proof. We assumed that X is nonempty. Let $x \in X$ be a feasible point. It is easy to see that the point $(0,x) \in \mathbb{B}^{k_i+\ell_i} \times X$ is feasible to Opt(i). ◀

Let n_i denote the (global) optimal value of Opt(i).

▶ **Proposition 3.** *If n_i equals $k_i + \ell_i$, then X_i is consistent.*

Proof. If the optimal value n_i equal to $k_i+\ell_i$, then $u_{\tilde{j}} = 1$ for all $\tilde{j} \in \{1,2,\dots,k_i+\ell_i\}$. From the constraints in Opt(i), this implies that there is an $\tilde{x} \in X$ that satisfies the constrains in the description of (2). Therefore, $\tilde{x} \in X_i$ and $X_i \neq \emptyset$. ◀

▶ **Proposition 4.** *If $n_i < k_i + \ell_i$, then X_i is inconsistent. Furthermore, the set of constraints in Opt(i) that correspond to $u_{\tilde{j}} = 1$ for $\tilde{j} \in \{1,2,\dots,k_i+\ell_i\}$ form the largest subset of constraints in X_i that is consistent.*

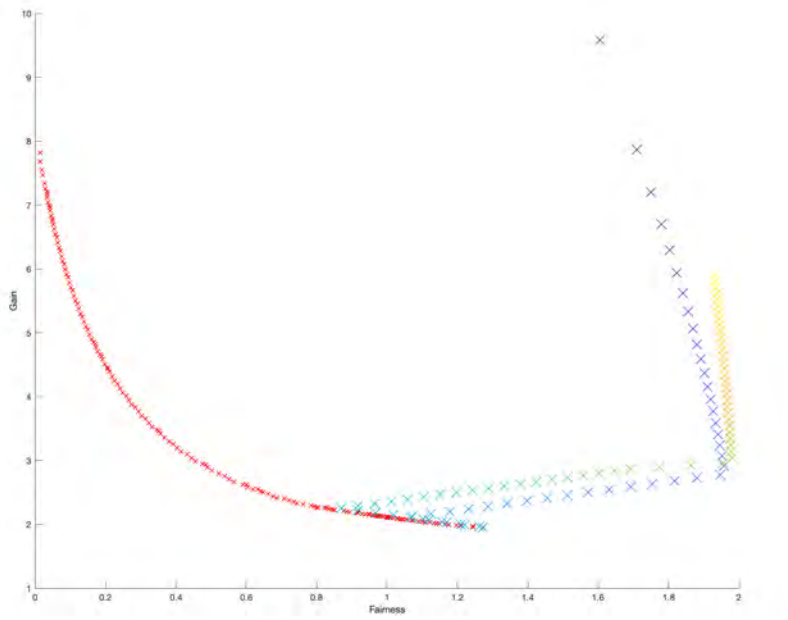


Figure 22 The red crosses are the Pareto front approximation in the Gain/Fairness space, the coloured crosses are the solutions that form the Pareto front approximation in the objective space.

Proof. If $n_i < k_i + \ell_i$, then $u_{\tilde{j}} = 0$ for exactly $k_i + \ell_i - n_i$ indices $\tilde{j}$'s. These are the constrains for which $u_{\tilde{j}} \neq 1$, meaning they cannot be satisfied. Those $\tilde{j}$'s for which $u_{\tilde{j}} = 1$ correspond to the constraint that can be simultaneously satisfied, and therefore the statement of the proposition follows. ◀

Solving Opt(i), therefore, is informative to the decision maker i. Depending on the solution to Opt(i), this either shows that the constraints formulated by him/her are consistent, or, in the inconsistent case, gives information about the constraints that are causing inconsistency. It is clear that all the X_i's need to be non-empty, before any optimization problem or preference information could be formulated.

▶ Remark. is to be noted that for checking whether a set is consistent or not can often be realized by minimizing a penalty function like $\sum_j \max\{0, g_j^i(x)\} + \sum_l |h_j^l(x)|$ or its differentiable version $\sum_j (\max\{0, g_j^i(x)\})^2 + \sum_l |h_j^l(x)|^2$. A minimal point x with minimal value zero is feasible, while a positive minimal value proves inconsistency. The latter can sometimes also be verified by a lower bounding procedure (like interval arithmetic) if this yields a positive lower bound for the penalty function. If only inequality constraints are present, a common feasibility test also is the minimization of some variable t subject to the constraints $g_j(x) \leq t$ for all j and $t \geq -1$ (to guarantee boundedness). A positive optimal value proves inconsistency, otherwise a feasible point x is generated. The advantage of our approach based on Opt(i) is that it explicitly gives the constraints that are causing inconsistency.

Detecting inter-inconsistencies across the constraint sets

In this section we assume that $X_i \neq \emptyset$ for all $i \in \{1, \ldots, d\}$. The individual constraint sets of the decision-makers are, therefore, non-empty (i.e., the intra-inconsistencies, if any, are assumed to be removed). The aim now is to check if $\bigcap_{i=1}^{d} X_i \neq \emptyset$. Obviously, $\bigcap_{i=1}^{d} X_i = \emptyset$ would mean that there exist at least two decision-makers i and j such that $X_i \cap X_j = \emptyset$, meaning that these decision makers *cannot* be simultaneously satisfied. To check this, we consider the following optimization problem:

$$\begin{aligned} \max_{(v,x)\in\mathbb{B}^d\times X} & \sum_{i=1}^{d} v_i \\ \text{s.t.} \quad & v_i g_j^i(x) \leq 0,\ \forall j = 1, \ldots, k_i, \forall i = 1, \ldots, d \\ & v_i h_l^i(x) = 0,\ \forall l = 1, \ldots, \ell_i, \forall i = 1, \ldots, d. \end{aligned} \tag{Opt}$$

▶ **Proposition 5.** *The constraint set of Opt is non-empty.*

Proof. We assumed that X is nonempty. Let $x \in X$ be a feasible point. It is easy to see that the point $(0, x) \in \mathbb{B}^d \times X$ is feasible to Opt(i). ◀

Let $\tilde{n}$ denote the (global) optimal value of Opt.

▶ **Proposition 6.** *Let $X_i \neq \emptyset$ for all i. If $\tilde{n}$ equals d, then $\bigcap_{i=1}^{d} X_i \neq \emptyset$.*

Proof. As $X_i \neq \emptyset$ for all $i \in \{1, \ldots, d\}$ we obtain from Proposition 6 that all the components of u in Opt(i) equal 1, for every i. This means that a common value v_i can be used for all the inequalities and equalities that define X_j. Therefore, for every $i \in \{1, \ldots, d\}$, it is sufficient to use just one boolean variable v_i to represent all the k_i inequality and ℓ_i equality constrainsts.

If the optimal value $\tilde{n}$ equal to d, then $v_i = 1$ for all $i \in \{1, \ldots, d\}$. From the constraints in Opt, this implies that there is an $\tilde{x} \in X$ that satisfies the constrains in the description of (2) for every index i. Therefore, $\bigcap_{i=1}^{d} X_i \neq \emptyset$. ◀

▶ **Proposition 7.** *If $\tilde{n} < d$, then there are at least two decision makers having inconsistent constraint sets. Furthermore, the set of constraints in Opt that correspond to $v_i = 1$ for $i \in \{1, 2, \ldots, d\}$ form the largest number of decision makers that have common feasible sets.*

Proof. If $\tilde{n} < d$, then $v_i = 0$ for exactly $d - \tilde{n}$ indices i's. These are the constraint sets of decision makers for which $v_i \neq 1$, meaning they cannot be satisfied simultaneously. Those i's for which $v_i = 1$ correspond to the constraint sets (or decision makers) that can be simultaneously satisfied, and therefore the statement of the proposition follows. ◀

Solving Opt, therefore, is informative to all the decision makers. Depending on the solution to Opt, this either shows that the constraints formulated by the decision makers are consistent globally, or, in the inconsistent case, gives information about the decision makers that are causing inconsistency.

4.6.6 Conclusions and future directions

This report discussed in general problems that arise when moving from problems with a single DM to problems with multiple or many DMs. We discussed the general process of solving such problems, and then focused on two more specific aspects, namely:

1. integrating fairness and gain in problems with conflicting preferences among the decision makers. For this we identified NSGA-II for teams and introduced the notion of the fairness-gain space in addition to the objective and decision space for problems with many DMs.
2. solving and discussing problems with conflicting constraints among different decision makers using a formal framework.

Future work will have to be done to further refine these ideas and test them on benchmarks or real world problem solving scenarios. Moreover, it might become necessary to develop additional techniques in case the number of DMs is large (in accordance to the term 'many objective optimization one might term this 'many decision makers' scenario). In this context the following idea might become very relevant:

Apart from including all DM preferences individually into the optimization and decision making process, an alternative and promising perspective for future research lies in reducing complexity of decision preferences prior to optimization. Thus, e.g. by means of clustering techniques, reference points which are representative for subgroups of decision makers could be generated which in combination then reduce the amount of different decision maker preferences and potentially reduces the complexity of agreeing on final compromise solutions. Ideally, cluster centers reflecting preference compromises of the subgroups should be presented to the decision makers prior to optimization.

In general such frameworks should also be integrated further with methodologies used in the MCDM community that have been developed for team decision making and consensus finding given only a small number of alternative solutions. There is a rich literature to be explored here and it would extend the scope of this report to provide a comprehensive overview.

Source code

Matlab-code is available on: http://moda.liacs.nl.

Acknowledgements

We are grateful to Prof. José Figueira about his input on the definitions of fairness/equity, Prof. Hisao Ishibuchi for contributions in the initial discussions, and Prof. Oliver Stein for his remarks on constraint feasibility.

References

1 Kalyanmoy Deb. Multi-objective optimisation using evolutionary algorithms: an introduction. In *Multi-objective evolutionary optimisation for product design and manufacturing*, pages 3–34. Springer, 2011.

2 Michael Emmerich, André Deutz, Longmei Li, Asep Maulana, and Iryna Yevseyeva. Maximizing consensus in portfolio selection in multicriteria group decision making. *Procedia Computer Science*, 100:848–855, 2016.

3 Ralph L. Keeney. Value-focused thinking: Identifying decision opportunities and creating alternatives. *European Journal of Operational Research*, 92(3):537–549, 1996.

4 Wouter M Koolen. The pareto regret frontier. In C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 26*, pages 863–871. Curran Associates, Inc., 2013.

5 Kaisa Miettinen and Marko M Mäkelä. On scalarizing functions in multiobjective optimization. *OR spectrum*, 24(2):193–213, 2002.

4.7 Turning Objective Functions into Constraints? Or Vice Versa?

Georges Fadel (Clemson University – Clemson, US), Karl-Heinz Küfer (Fraunhofer ITWM – Kaiserslautern, DE), Manuel López-Ibáñez (University of Manchester, GB), Luís Paquete (University of Coimbra, PT), Stefan Ruzika (TU Kaiserslautern, DE), and Anita Schöbel (Fraunhofer ITWM – Kaiserslautern, DE)

4.7.1 Introduction and Motivation

Multiobjective mathematical programming problems are characterized by the presence of several, typically incommensurable objective functions. This circumstance is a fundamental difference to single objective optimization. As a consequence, the notion of optimality has to be revised. One common way to define optimality in the presence of multiple objective functions is the notion of Pareto-optimality. An efficient (or Pareto-optimal) solution to a mathematical programming problem is characterized by the fact that there does not exist another solution that is at least as good in all objectives and strictly better in at least one objective. The images of these efficient solutions are referred to as nondominated points. The goal of multiobjective optimization is to compute the set of nondominated points and at least one efficient solution for each nondominated point. For a rigorous introduction to multiobjective programming, we refer to the book [3].

It is well-known that there is an increase in difficulty when considering an arbitrary number of objective functions compared to the biobjective case [5]. This can be explained by the kind of trade-off which stems from the notion of Pareto optimality: In biobjective problems, two different nondominated points are characterized by the fact that one solution is better in one objective and worse in the other. As a consequence, there is a monotonicity among the nondominated points, i.e., they can be sorted by increasing values of the first objective and, as a consequence, the values of the second objective will turn out to be decreasingly sorted. This is not the case when having three or more objectives.

A research direction that has lately attracted quite some interest, is multiobjective problems with *many* objective. In this context, the term 'many' is not precisely defined. It is generally understood as pointing towards the fact that the mere number of objectives imposes some difficulty, e.g., in the theoretical analysis of structural properties, of the numerical solution of the problem, or in the decision-making process.

This situation is, in principle, not new to multiobjective optimization, since the presence of more than one objective inherently already imposes an increase in difficulty compared to the single-objective situation. One way of copying with this fact is as old as multiobjective optimization and widely spread: the idea of reducing the number of objective functions. This technique is called *scalarization* and there is a huge body of literature on scalarization [3, 4].

In this article, we address the question of whether it is beneficial to reformulate some given multiobjective optimization problem by converting one or more objectives to constraints. The resulting problem would have fewer objectives and should be easier to solve. However, only a subset of Pareto-optimal solutions is found and, therefore, this process of formulating and solving a smaller problem has to be iteratively repeated to guarantee finding all or a good representation of the Pareto optimal solutions. In a sense, this technique can be referred to as *partial scalarization*.

4.7.2 A First Theoretical Framework

We consider a multiobjective optimization problem with k objectives and m constraints. On top of these constraints, we denote additional constraints by X. This set X subsumes other restrictions, e.g., sign constraints, bounds on variables, or other side constraints that are not in the focus of the following consideration. We denote the encoding length of X by n. A multiobjective optimization problem can thus be referred to by $MO(k, m)$ and concisely stated as

$$\begin{aligned} \min \quad & \begin{pmatrix} f_1(x) \\ \vdots \\ f_{k-1}(x) \\ f_k(x) \end{pmatrix} \\ \text{s.t.} \quad & g_1(x) \leq 0 \\ & \vdots \\ & g_m(x) \leq 0 \end{aligned} \tag{3}$$

One way of studying and balancing the trade-off between objectives and constraints is by means of the so-called *ε-constraint method* [7]. In this scalarization technique, one of the objective functions is minimized while all other are reformulated as constraints. In the following, we slightly adapt this idea: we choose one objective and reformulate it as a constraint while keeping all remaining objectives. In view of this partial scalarization, the above $MO(k, m)$ problem (Eq. 3) would be reformulated as the following (multiobjective) mathematical programming problem:

$$\begin{aligned} \min \quad & \begin{pmatrix} f_1(x) \\ \vdots \\ f_{k-1}(x) \end{pmatrix} \\ \text{s.t.} \quad & g_1(x) \leq 0 \\ & \vdots \\ & g_m(x) \leq 0 \\ & f_k(x) \leq \varepsilon \end{aligned} \tag{4}$$

For some given ε, we refer to the above problem as $MO(k-1, m+1)$. Comparing the two problems $MO(k, m)$ and $MO(k-1, m+1)$, the sets of efficient solutions and nondominated points obviously differ (see [2, 6] on the effect of adding/deleting objective functions). The step of reducing the objectives by one and adding one more constraint can of course be iteratively applied until, at the end, the resulting problem is equivalent to the ε-constraint problem. This technique of reducing the number of objectives and increasing the number of constraints is key to our analysis.

To find all nondominated points of $MO(k, m)$ by means of solving problems of the kind $MO(k-1, m+1)$, one can use an iterative algorithm that varies the chosen bound ε and solves these smaller problems. Let I denote the number of relevant values for the bound ε. We refer to such an iterative algorithm by $\varepsilon ALG(k, m)$. In other words, $\varepsilon ALG(k, m)$ is an iterative algorithm based on partial ε-constraint scalarization calling I-times an algorithm $ALG(k-1, m+1)$ that solves some specific problem with k objectives and m constraints. Figure 23 illustrates this idea.

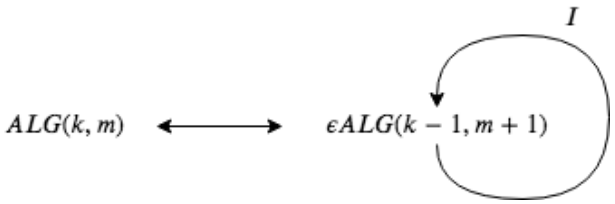


Figure 23 Illustration of partial ε-constraint scalarization used in an iterative manner.

The following result establishes an obvious connection between $\varepsilon ALG(k,m)$ and $ALG(k,m)$.

▶ **Theorem 1.** *$ALG(k,m)$ is faster than $\varepsilon ALG(k,m)$ if*

$$time[ALG(k,m)] < I \cdot time[ALG(k-1,m+1)]$$

Therefore, the algorithm $ALG(k,m)$ is computationally preferable if $I > \frac{time[ALG(k,m)]}{time[ALG(k-1,m+1)]}$. Note that I is determined by the number of nondominated solutions of $MO(k,m)$.

4.7.3 Some Examples

The Multiobjective Unconstrained (Integer) Quadratic Minimization Problem

The problem

$$\min(x-a)^2$$

for a given parameter $a \in \mathbb{R}^n$ and a (maybe integer) variable $x \in \mathbb{R}^n$ is a nice problem for visualization for $n = 2$. For example, Given $a_1, \ldots, a_k$ points in the plane, we may formulate k objective functions, namely $f_k(x) = (x - a_k)^2$ describing the distance from x to a_k. One can visualize what reducing the number of objective functions means: Instead of minimizing f_k we restrict the solution space to a ball with radius ε centered at a_k. It can be easily seen that due to the nonlinear constraint, the problem becomes much more complicated. While it can be used for visualization, it seems not to be suited for an analytical analysis.

Multiobjective Linear Programming

Multiobjective linear programming seems to be a suitable problem since
a) the multiobjective simplex algorithm is available for experiments;
b) the ε-constraint problem is again a multiobjective linear program that can be solved by the multiobjective simplex algorithm.

It remains open to use this setting for numerical experiments.

4.7.4 The Multiobjective Shortest Path Problem

In the following, we illustrate the application of these concepts for a particular case of the shortest path problem with k objectives in a directed acyclic graph. We consider an extension of the *pulling* algorithm [1], which is the fastest approach for finding the single-source single-sink shortest path in acyclic and topologically ordered networks. This extended pulling algorithm processes the nodes in the topological order. At each iteration i $(i = 1, \ldots, n)$, it

Table 4 Computational results.

k	objectives	constraints	CPU-time
3	3	0	0.00
	2	1	0.35
	1	2	1.45
4	4	0	0.00
	3	1	1.77
	2	2	6.84
	1	3	28.47

calculates the nondominated shortest paths from node 1 to node v_i by considering only the distances from each node v_j that is incident to j, $j < i$. Since the network is topologically ordered, the nondominated shortest paths to each node v_j were already computed.

The following pseudo-code implements the algorithm above, where $ND_{1,\ldots,k}(\cdot)$ is a procedure that extracts the nondominated paths with respect to the k objectives and L_i is a list that stores the nondominated paths at each node i.

Input: An acyclic graph $G = (V, A)$ with $n = |V|$ nodes, $V = \{v_1, \ldots, v_n\}$, $p = |A|$ arcs and k costs at each arc
1: **Initialization**: set $L_1 = \{(0, \ldots, 0)\}$ and $L_i = \{\}$, for $i = 2, \ldots, n$
2: **for** $i = 1, \ldots, n-1$ **do**
3: **for all** v_j, $(v_i, v_j) \in A$ **do**
4: $L_j = ND_{1,\ldots,k}\left(L_j \cup \{\ell + c_{i,j} \mid \ell \in L_i\}\right)$
5: **Return** L_n

In the following, we present the pseudo-code of an algorithm that solves the mutiobjective shortest path problem with $k-1$ objectives and a resource constraint, which corresponds to one step of the ϵ-constraint for this problem. This algorithm is similar to the previous one, except that only nondominated paths with respect to the $k-1$ objectives and the cost, and that satisfy the resource constraint, are stored on each list L_i, $i = 1, \ldots, n$. Note that is easy to adapt it to handle more resource constraints.

Input: An acyclic graph $G = (V, A)$ with $n = |V|$ nodes, $V = \{v_1, \ldots, v_n\}$, $p = |A|$ arcs, k costs at each arc and a resource constraint ϵ on the k-th cost
1: **Initialization**: set $L_1 = \{(0, \ldots, 0)\}$ and $L_i = \{\}$, for $i = 2, \ldots, n$
2: **for** $i = 1, \ldots, n-1$ **do**
3: **for all** v_j, $(v_i, v_j) \in A$ **do**
4: $L_j = ND_{1,\ldots,k}\left(L_j \cup \{\ell + c_{i,j} \mid \ell^k + c^k_{i,j} \leq \epsilon, \ell \in L_i\}\right)$
5: **Return** L_n

We performed an experimental analysis of these algorithms on several randomly generated graphs for size 50 with 3 and 4 objectives. Table 4 presents the running time in seconds for each combination on the number of objectives and number of resource constraints. The results clearly indicate that the CPU-time increases quite strongly with the increase on the number of constraints.

4.7.5 Summary and Open Questions

Our claim is that, despite solving a sequence of supposedly easier problems with fewer objectives, $\varepsilon ALG(k, m)$ is (typically) inferior to $ALG(k, m)$ under reasonable assumptions. However, more precise analysis of run time necessary to compare the two algorithmic approaches.

Furthermore, if a specific algorithm is available for the many-objective problem, it may be even more efficient to solve it than equally structured fewer objective problem. Also, the many-objective problem may yield more information potentially relevant for the decision-maker.

References

1 Ahuja, Ravindra K., Magnanti, Thomas, Orlin, James B.: Network Flows: Theory, Algorithms and Applications. Prentice-Hall (1993)
2 Dempe, Stephan, Eichfelder, Gabriele, Fliege, Joerg: On the effects of combining objectives in multi-objective optimization. Mathematical Methods of Operations Research **82**(1), 1–18 (2015)
3 Ehrgott, Matthias: Multicriteria Optimization. Springer, Berlin, Germany, 2nd edn. (2005). 10.1007/3-540-27659-9
4 Ehrgott, Matthias: A discussion of scalarization techniques for multiple objective integer programming. Annals of Operations Research **147**(1), 343–360 (2006)
5 Figueira, Jose R., Fonseca, Carlos M., Halffmann, Pascal, Klamroth, Kathrin, Paquete, Luis, Ruzika, Stefan, Schulze, Britta, Stiglmayr, Michael, Willems, David: Easy to say they are hard, but hard to see they are easy—towards a categorization of tractable multiobjective combinatorial optimization problems. Journal of Multi-Criteria Decision Analysis **24**(1-2), 82–98 (2017). 10.1002/mcda.1574
6 Fliege, Joerg: The effects of adding objectives to an optimisation problem on the solution set. Operations Research Letters **35**(6), 782–790 (2007)
7 Haimes, Yacov, Lasdon, Leon, Da Wismer, Da: On a bicriterion formation of the problems of integrated system identification and system optimization. IEEE Transactions on Systems, Man, and Cybernetics **1**(3), 296–297 (1971). 10.1109/TSMC.1971.4308298

4.8 Data and Preference Driven Objective Space Reduction in Multiobjective Optimization

Serpil Sayin (Koc University – Istanbul, TR), Mickaël Binois (INRIA – Valbonne, FR), and Margaret M. Wiecek (Clemson University, US)

The complexity of multiobjective optimization problems (MOPs) is addressed from the perspective of a large number of objective functions. Two approaches to reduce the number of objectives are proposed. In contrast to prior studies that have concentrated on mathematical features of the MOP allowing for decrease of the number of objectives, the developed approaches exploit the real-life context in which the MOP is being solved. The first approach relies on the classical Principal Component Analysis and is based on the data that is carried by an instance of the MOP, while the other is the Active Subspace Approach, additionally making use of the decision maker's desirability function. Numerical examples are included.

4.8.1 Introduction

This paper contains a report of the work performed by the Working Group "Objective reduction for many-objective problems" at the Dagstuhl Seminar 20031 "Scalability in Multiobjective Optimization" that took place in Schloss Dagstuhl – Leibniz Center for Informatics – on January 12-17, 2020.

Solving multiobjective optimization problems (MOPs) has been well studied and is known to be difficult for some classes of problems (see [6] and many others). The computational difficulty grows with an increase in the number of objective functions. However, in the presence of many objectives, not all functions may be of interest to the decision maker (DM) or not all objectives may be in conflict with each other. It is of interest to make the original MOP simpler by removing unnecessary objective functions, or by building combinations of objectives, while the solution set remains unchanged.

The concept of redundant (or, also called later, *nonessential*) objective functions is first introduced in [7]. An objective function is said to be *redundant* if the efficient set is unchanged when that function is removed. For multiobjective linear programs (MOLPs), sufficient conditions are given for determining the redundancy of an objective function in [7] and later extended to necessary and sufficient conditions in [13, 11]. Similar results for MOPs with quasiconvex objective functions under the assumptions of a compact and convex feasible set or injective objective functions are developed in [12]. The weakness of this approach is that only one objective at a time may be tested for redundancy and that this testing may have a high computational cost if there is a large or infinite number of objectives. The concept of representative objective functions for MOLPs is introduced in [21] and algebraic properties of objectives are used to reduce their number but maintain the same or an equivalent efficient set. A collection of objectives is called *representative* provided all objectives not in the collection can be represented as a conical combination of the criteria in the collection. It is shown that the efficient set to the original MOLP is equal to the efficient set of the problem reduced to a representative family of objectives.

Another way of reducing a large number of objective functions is to construct subproblems with a smaller number of criteria. In [14] it is shown that the weakly efficient set of the convex MOP with n variables is equal to the union of the efficient sets of subproblems obtained from the original MOP by selecting at most $n + 1$ criteria [14]. In this context, the MOP is said to be *Pareto reducible* if its weakly efficient solutions are the efficient solutions for this MOP itself and also for a subproblem obtained from it by selecting certain objectives [17, 18].

The aim of this paper is to further study MOPs with a large number of objectives, possibly non-linear, and develop other rational and practical possibilities for reducing the number of objectives. In contrast to prior studies as above that have concentrated on mathematical features of the MOP allowing for decrease of the number of objectives, we propose two reduction approaches exploiting the real-life context in which the MOP is being solved. The first approach is Principal Component Analysis (PCA), which is based on the data that is carried by an instance of the MOP. PCA has been incorporated into Evolutionary Multiobjective Optimization (EMO) methods that solve problems with *many* objectives. In [5], PCA is incorporated into the EMO algorithm NSGA-II to reduce the number of objectives throughout the solution process as populations evolve. This has been reported to improve performance of EMO algorithms. Other PCA variations, including kernel PCA, have been implemented in different settings such as in [19]. In addition, PCA has been used to improve computational performance of the hypervolume based approaches [2] or to provide visualizations as in [4]. The analysis presented in this report differs from those studies mainly in its focus on identifying a mapping that would support the user's understanding of the set

of Pareto solutions rather than solely using those methods as tools to improve computational efficiency of existing algorithms. The second approach relies on the cost (utility) function used by the DM who is involved in a decision-making process of choosing the most preferred solution from among all Pareto solutions of the MOP. This approach is referred to as the Active Subspace Approach (ASA) since it builds on the active subspace methodology for dimension reduction in parameter studies [3].

We envision an environment where some efficient solutions and thus Pareto points have been obtained by utilizing an optimization algorithm. The available Pareto points are given as input data that is used to explore the relationships among the objective functions using the two techniques, PCA and ASA. If the techniques identify a reduction of objectives, we anticipate that the reduced MOP would be solved instead of the original more complex one.

In this preliminary work, we study the two reduction approaches in more detail to gain insight into how they might contribute to the effort of MOP complexity reduction in terms of reducing the number of objective functions. The report is structured as follows. We provide a mathematical definition of the problem under study and introduce two problem instances of an MOP we experiment with. In Sections 4.8.2 and 4.8.3 we present the PCA and ASA approaches respectively, while in Section 4.8.4 we compare the obtained results and discuss future research directions.

Problem Statement

The MOP of interest is defined as

$$\begin{array}{lll} (\mathbf{MOP}) & \min & f(\mathbf{x}) = (f_1(\mathbf{x}), \ldots, f_p(\mathbf{x})) \\ & \text{s.t.} & \mathbf{x} \in \mathcal{X} \end{array}$$

where $\mathcal{X} \subseteq \mathbb{R}^n$ denotes the feasible set and $f_i : \mathbb{R}^n \to \mathbb{R},\ i = 1, \ldots, p$ are the objective functions.

We assume the MOP is difficult to solve due to the large number of objectives it contains. Therefore a goal could be to eliminate some objectives from the MOP and be left with the same MOP but with a smaller number of objectives, which would call for identifying redundant objectives, if the MOP contains any. Alternatively, a goal could be to build another MOP, which is equivalent to the original one in some sense, but having a smaller number of objectives that are not all a subset of the original ones but perhaps a combination of them.

Examples

We demonstrate the ideas and work on two instances of the multiobjective knapsack problem (MOKP). In the MOKP, there are n objects each of which has a positive integer weight w_r and p non-negative integer profits v_r. The decision variable x_r denotes whether object r is selected for the knapsack or not. The total weight of selected objects should be within an integer capacity $W > 0$.

$$\begin{array}{lll} (\mathbf{MOKP}) & \max & \displaystyle\sum_{r=1}^{n} v_r^j x_r \quad j = \{1, \ldots, p\} \qquad (5) \\ & \text{s.t.} & \displaystyle\sum_{r=1}^{n} w_r x_r \leq W \qquad (6) \\ & & x_r \in \{0, 1\} \quad r = 1, \ldots, n \qquad (7) \end{array}$$

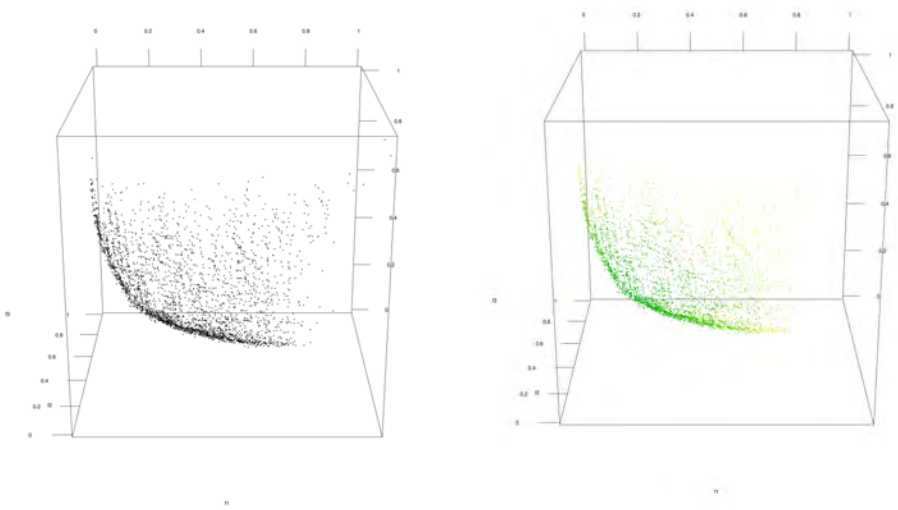


Figure 24 Left: Pareto points for the 3D case. Right: Pareto points for the 3D case with desirability values in color scale (yellow is low, green is high).

The vector of p objective functions where each objective function denotes the total profit of chosen objects is given in (5). Inequality (6) models the capacity constraint meaning that the total weight of selected objects has to be less than or equal to the knapsack's capacity. The binary constraints complete the model in (7) .

Two instances of the MOKP used in the computational experiments in [10] are selected. The first instance has $p = 3,\ n = 100$ and the second instance has $p = 5,\ n = 20$. These are instances $KP\_p-3\_n-100\_ins-10$ and $KP\_p-5\_n-20\_ins-10$ in [9] and from here on they are referred to as the 3D case and the 5D case, respectively. The Pareto set of the 3D case contains about 3200 points and of the 5D case about 400 points.

The PCA-based approach does not require any additional information from the DM and relies on problem data. The ASA can be used when some valuations emulating decision maker's preferences are available. In the two cases, this preference/desirability function is taken as $F : \mathbf{x} \in [0,1]^p \to \sum_{i=1}^{p}(x_i - -1)^2$, operating on normalized data. The Pareto points of the 3D case with and without desirability information are depicted in Figure 24.

4.8.2 Principal Component Analysis

In this section the PCA is reviewed, and the resulting approach is presented and applied to the two cases of the MOKP.

Data Driven Reduction of the Objective Space

PCA [16] is a statistical procedure that is designed to reduce the number of *attributes* in a data set without losing its essence. A reduction is achieved by building new *components* that are linear combinations of the original attributes. Therefore the main idea is to build a transformation of the original data set that can be used for exploratory or predictive analysis. In the Pearson sense, this transformation is built by seeking the sum of least squares that optimizes the projection of the original data set onto a lower dimensional subspace [20]. In the Hotelling version, components are built iteratively in a way to carry the maximum

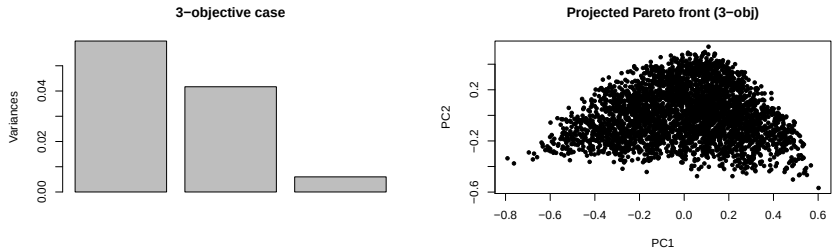


Figure 25 PCA results for the 3D case. Left: variances. Right: projection of the Pareto set on the first two principal components.

variance and remain orthonormal to the previously built ones [1]. There are various methods with different computational aspects to implement PCA. In general, PCA requires computing the covariance matrix and then finding its eigenvalues and eigenvectors. The magnitude of the eigenvalues are indicators of the importance, and the associated eigenvectors indicate the components. Truncation is generally performed, relying on an arbitrary threshold on the variance explained or on probabilistic interpretations, see, e.g., [15]. PCA is used extensively in engineering and machine learning for dimensionality reduction.

In Algorithm 1 we give a pseudo code of the procedure we apply to the test cases.

Algorithm 1 Pseudo-code for the PCA-based procedure.

Require: A set of uniformly spread Pareto points $\mathbf{y}^1, \ldots, \mathbf{y}^K$ in the objective space, represented by the $K \times p$ matrix Y.

1: Normalize Y. Compute the covariance matrix of Y, denoted as C.
2: Obtain the eigen-decomposition of C.
3: Sort the eigenvectors in the decreasing order of eigenvalues and determine the number of principal components, k, that suffice to represent Y.
4: Map Y using the first k eigenvectors.

The 3D Example

Algorithm 1 applied to the 3D case produces the matrix of eigenvectors given below, where every column is referred to as a principal component.

$$\begin{bmatrix} PC1 & PC2 & PC3 \\ -0.199 & -0.891 & -0.408 \\ 0.710 & 0.156 & -0.687 \\ -0.675 & 0.426 & -0.602 \end{bmatrix}$$

The results given in Figure 25 show that the first two principal components (objectives) carry most of the variance, and that the projection on the first two components keeps most of the shape of the 3D Pareto set. This result suggests that the 3D problem can be reformulated as an MOP with two objective functions using the first two component vectors as weights that combine the three objective functions with minimal loss of information.

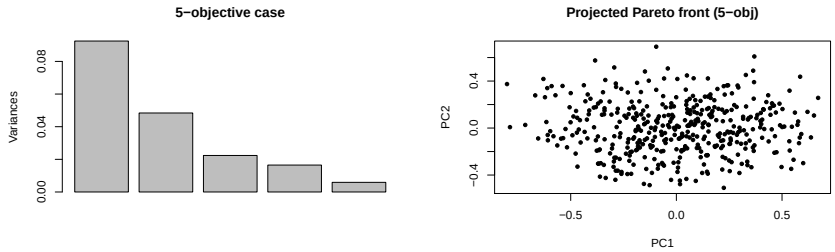


Figure 26 PCA results for the 5D case. Left: variances. Right: projection of the Pareto set on the first two principal components.

The 5D Example

When we apply Algorithm 1 to the 5D example, we obtain the following matrix of five principal components:

$$\begin{bmatrix} PC1 & PC2 & PC3 & PC4 & PC5 \\ -0.140 & 0.753 & -0.077 & 0.547 & -0.330 \\ 0.498 & -0.279 & 0.315 & 0.102 & -0.7511 \\ -0.554 & 0.082 & 0.800 & -0.198 & -0.089 \\ 0.413 & 0.582 & 0.052 & -0.698 & -0.014 \\ -0.505 & -0.098 & -0.503 & -0.406 & -0.563 \end{bmatrix}$$

The results depicted in Figure 26 indicate that the fifth component carries very little variance and therefore can be dropped from consideration in the transformed MOP. In the same figure we also observe the projection of the points onto two-dimensional subspace determined by the first two principal components. We note that it may be possible to drop the fourth and even the third principal component and still not lose the essence of the Pareto set of the original problem. However, the implications of all these reformulations must be studied and understood carefully.

4.8.3 Active Subspace Approach

In this section the ASA is reviewed, and the resulting algorithm is presented and applied to the two cases of the MOKP.

Preference Driven Reduction of the Objective Space

When the decision maker is able to provide a desirability or cost function, which can be only approximate or estimated by another method, then this additional information can be taken into account and used in the ASA. We study the transposition of the active subspace methodology, see [3], to the context of interest in this report.

In parameter studies with a function $F : \mathbb{R}^p \to \mathbb{R}$, ASA is applied to reduce the dimension on the parameter space $\mathbb{R}^p$ to avoid the curse of dimensionality. The principle is to estimate the $p \times p$ matrix $\mathbf{C} = \int\limits_{\mathbf{y} \in D \subset \mathbb{R}^p} \nabla F(\mathbf{y}) \nabla F(\mathbf{y})^\top \mu(d\mathbf{y})$, where D is the domain of F and μ is an appropriate measure on D, usually uniform for bounded domains and Gaussian for unbounded ones. The eigen-decomposition of $\mathbf{C}$ is then computed in the form $\mathbf{C} = \mathbf{W}\mathbf{\Lambda}\mathbf{W}^\top$, where $\mathbf{\Lambda}$ is the diagonal matrix of eigenvalues with its columns sorted in the decreasing order of the eigenvalues, and $\mathbf{W}$ is the matrix of eigenvectors.

The first eigenvector is then the direction along which most of the variations of F occur on average. The last eigenvector determines the direction along which least variations of F

occur on average. The active subspaces are identified by gaps in the eigenvalues: the larger the gap between two subsequent eigenvalues, the more important is the subspace defined by the eigenvectors preceding the gap. The eigenvectors define a rotation of the original space $\mathbb{R}^p$ and consequently the domain of F. The $\mathbb{R}^p$ space is reduced by dropping the directions associated with the smallest eigenvalues [3, 8]. A multi-objective version of the method is proposed by [22] and could be further explored.

In the many objective context we consider, the Pareto points in $\mathbb{R}^p$ are the designs, the desirability function is F, and a natural choice for μ is the uniform measure on the Pareto set. In practice, the estimation of $\mathbf{C}$ is detailed in Algorithm 2.

Algorithm 2 Pseudo-code for the ASA procedure.

Require: A set of uniformly spread Pareto points $\mathbf{y}^1, \ldots, \mathbf{y}^K$ in the objective space, and a cost/desirability function $F : \mathbb{R}^p \to \mathbb{R}$

1: Compute the gradient of F at each point of the sample: $\nabla F(\mathbf{y}^1), \ldots, \nabla F(\mathbf{y}^K)$

2: Compute $\hat{\mathbf{C}} = \frac{1}{K} \sum_{i=1}^{K} \nabla F(\mathbf{y}^i) \nabla F(\mathbf{y}^i)^\top$

3: Obtain the eigen-decomposition of $\hat{\mathbf{C}}$

4: Look for gaps in the eigenvalues of $\hat{\mathbf{C}}$

We identify two potential applications of the ASA approach. First, when the desirability function is not known precisely, and second, to drive an optimization algorithm. These circumstances are further discussed in the following examples.

The 3D Example

For comparison with the components produced by PCA, we give $\hat{\mathbf{C}}$, the matrix of eigenvectors computed by Algorithm 2.

$$\begin{bmatrix} AS1 & AS2 & AS3 \\ 0.576 & 0.132 & 0.806 \\ 0.588 & -0.752 & -0.297 \\ 0.567 & 0.645 & -0.511 \end{bmatrix}$$

Similar to the PCA example, in the top of Figure 27 we give the weights of each component in the form of the (log) eigenvalues (left) and the projection of the Pareto points along the first two directions (right). In this case the first eigenvalue is much larger than the other two, hence a gap is formed. The projection on the first two components also resembles the shape of the Pareto set of the original problem. With the additional information carried in F, with ASA it is also possible to show the variability of F with respect to the active directions. In Figure 27 bottom left we observe that the F values almost linearly increase along the first eigenvector. On the contrary, as depicted in Figure 27 bottom right, representing the F values along the last eigenvector is much less informative, since they vary significantly when fixing the abscissa. In any case, this analysis provides information on the directions along which the DM's desirability varies less, which may be helpful when DM explores solutions of equivalent desirability when the desirability is not well known.

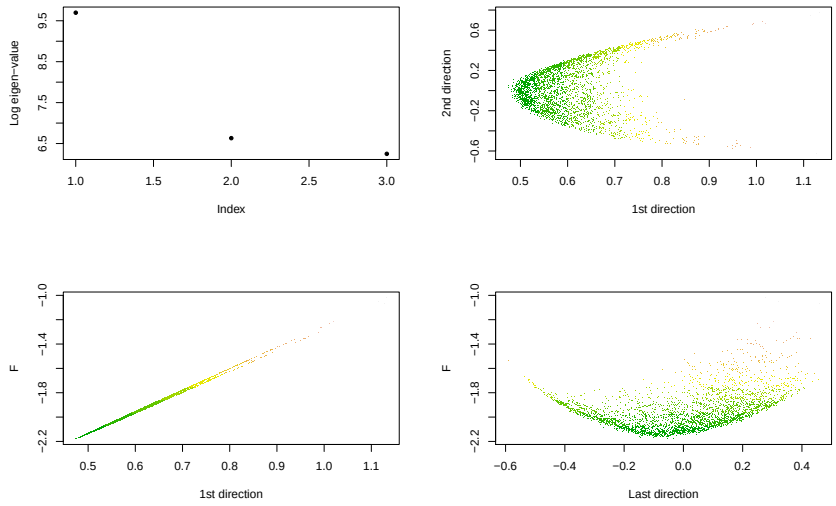


Figure 27 ASA results for the 3D case. Top left: log eigenvalues. Top right: projection of the Pareto set on the first two active directions. Bottom left: values of F along the first active direction. Bottom right: values of F along the second active direction. The color scale is the same as in Figure 24 (right).

The 5D Example

Again, for comparison with the components produced by PCA, we give $\hat{\mathbf{C}}$, the matrix of eigenvectors computed by Algorithm 2.

$$\begin{bmatrix} AS1 & AS2 & AS3 & AS4 & AS5 \\ -0.423 & 0.070 & 0.678 & -0.079 & 0.591 \\ -0.485 & -0.540 & -0.576 & -0.093 & 0.365 \\ -0.469 & 0.499 & -0.159 & -0.644 & -0.300 \\ -0.423 & -0.485 & 0.372 & 0.145 & -0.653 \\ -0.432 & 0.469 & -0.211 & 0.741 & -0.024 \end{bmatrix}$$

From the eigenvalues given in Figure 28 top left, there is a gap between the first and second eigenvalues, indicating that the first eigenvector is mostly sufficient to represent the F values, as shown in the bottom left panel in this figure. The projection on the last eigenvector is again not informative, since the F values vary greatly for the same abscissa. This is probably related to the simple structure of the F function used in this example, as its values depend on the radius of the sphere, which is recovered by ASA. Also the projection on the first two eigenvectors (in Figure 28 top right) is this time different from the one obtained in Figure 26 top right, which has a V shape. The reasons for this difference and the implications in terms of visualisation need further research.

4.8.4 Conclusion

Both PCA and ASA are descriptive tools giving insight into the problem at hand. Because PCA uses the information embedded in the data of the problem, while ASA additionally uses information external to the problem, the two methods provide different results.

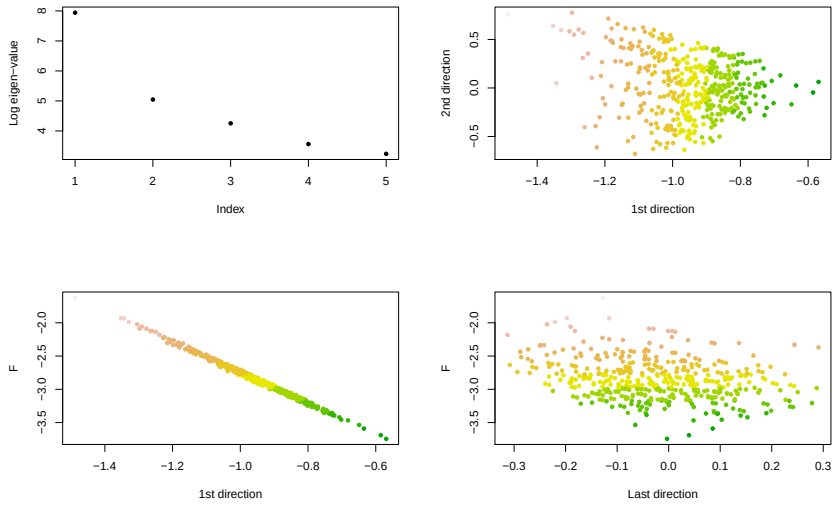


Figure 28 ASA results for the 5D case. Top left: log eigenvalues. Top right: projection of the Pareto set on the first two active directions. Bottom left: values of F along the first active direction. Bottom right: values of F along the second active direction. The color scale is the same as in Figure 24 (right).

Based on this preliminary study, we pose the following tasks for further research:

- Investigate the sensitivity of the PCA approach with respect to the number and/or distribution of the initial Pareto points supplied;
- Investigate the sensitivity of ASA with respect to the number and/or distribution of the initial Pareto points supplied **and** their costs/desirability;
- Establish the relationship between the Pareto set of the reduced problem and the Pareto set of the original problem using numerical instances;
- Study the effect of changing the Pareto cone into another polyhedral cone;
- Improve visualization of the projection of the basis directions of the original space as it is done for PCA with biplots, see e.g., [4].

We believe these investigations would be helpful in defining *reducibility* of MOPs and assessing the applicability of the two approaches presented in this report.

References

1 Rasmus Bro and Age K Smilde. Principal component analysis. *Analytical Methods*, 6(9):2812–2831, 2014.

2 Dimo Brockhoff and Eckart Zitzler. Improving hypervolume-based multiobjective evolutionary algorithms by using objective reduction methods. In *2007 IEEE congress on evolutionary computation*, pages 2086–2093. IEEE, 2007.

3 Paul G Constantine. *Active subspaces: Emerging ideas for dimension reduction in parameter studies*, volume 2. SIAM, 2015.

4 Lino Costa and Pedro Oliveira. Biplots in offline multiobjective reduction. In *IEEE Congress on Evolutionary Computation*, pages 1–8. IEEE, 2010.

5 Kalyanmoy Deb and D Saxena. Searching for Pareto-optimal solutions through dimensionality reduction for certain large-dimensional multi-objective optimization problems. In *Proceedings of the World Congress on Computational Intelligence (WCCI-2006)*, pages 3352–3360, 2006.

6 Matthias Ehrgott. *Multicriteria Optimization*. Springer, New York, 2005.

7 Tomas Gal and Heiner Leberling. Redundant objective functions in linear vector maximum problems and their determination. *European Journal of Operational Research*, 1(3):176–184, 5 1977.

8 John T Holodnak, Ilse CF Ipsen, and Ralph C Smith. A probabilistic subspace bound with application to active subspaces. *SIAM Journal on Matrix Analysis and Applications*, 39(3):1208–1220, 2018.

9 Gokhan Kirlik. Test instances for multiobjective discrete optimization problems, 2014.

10 Gokhan Kirlik and Serpil Sayın. A new algorithm for generating all nondominated solutions of multiobjective discrete optimization problems. *European Journal of Operational Research*, 232(3):479–488, 2014.

11 Agnieszka B Malinowska and Delfim FM Torres. Computational approach to essential and nonessential objective functions in linear multicriteria optimization. *Journal of Optimization Theory and Applications*, 139(3):577–590, 2008.

12 Agnieszka Barbara Malinowska. Changes of the set of efficient solutions by extending the number of objectives and its evaluation. *Control and Cybernetics*, 31(4):965–974, 2002.

13 Agnieszka Barbara Malinowska. Nonessential objective functions in linear multiobjective optimization problems. *Control and Cybernetics*, 35(4):873–880, 2006.

14 C Malivert and N Boissard. Structure of efficient sets for strictly quasi convex objectives. *Journal of Convex Analysis*, 1(2):143–150, 1994.

15 Thomas P Minka. Automatic choice of dimensionality for PCA. In *Advances in Neural Information Processing Systems*, pages 598–604, 2001.

16 Karl Pearson. Principal components analysis. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 6(2):559, 1901.

17 Nicolae Popovici. Pareto reducible multicriteria optimization problems. *Optimization*, 54(3):253–263, 2005.

18 Nicolae Popovici. Involving the helly number in Pareto reducibility. *Operations Research Letters*, 36(2):173–176, 2008.

19 Dhish Kumar Saxena and Kalyanmoy Deb. Non-linear dimensionality reduction procedures for certain large-dimensional multi-objective optimization problems: Employing correntropy and a novel maximum variance unfolding. In *International Conference on Evolutionary Multi-Criterion Optimization*, pages 772–787. Springer, 2007.

20 Jos MF Ten Berge and Henk AL Kiers. Optimality criteria for principal component analysis and generalizations. *British Journal of Mathematical and Statistical Psychology*, 49(2):335–345, 1996.

21 Nguyen V Thoai. Criteria and dimension reduction of linear multiple criteria optimization problems. *Journal of Global Optimization*, 52(3):499–508, 2012.

22 Olivier Zahm, Paul Constantine, Clementine Prieur, and Youssef Marzouk. Gradient-based dimension reduction of multivariate vector-valued functions. *arXiv preprint arXiv:1801.07922*, 2018.

5 Seminar schedule

Monday, January 13, 2020

09:00 – 10:30: Welcome Session

- Welcome and Introduction
- Short presentation of all participants (2 minutes each!)

Coffee Break

11:00 – 12:00: Challenges in Models

- Dimo Brockhoff, Michael Emmerich, Boris Naujoks & Robin Purshouse: MACODA – A Lorentz Center Workshop on "MAny Criteria Optimization and Decision Analysis"
- Arnaud Liefooghe: On the Difficulty of Multiobjective Combinatorial Optimization Problems

Lunch

14:00 – 15:00: Challenges in Methodology

- Mickael Binois: Scaling up Multi-Objective Bayesian Optimization
- Sanaz Mostaghim: Recent Advances in Multi-Objective Large Scale Optimisation

15:00 – 15:30: Group Discussion

Coffee Break

16:00 – 16:30: Working Group Formation

16:30 – 18:00: Working Groups

Dinner

19:30: Opening of the art exhibit "Das Loch das von der anderen Seite kam" by the German artist Lola Sprenger

Tuesday, January 14, 2020

09:00 – 09:30: Many Objectives in Stochastic Settings Chair: Heike Trautmann

- Susan Hunter: Multi-Objective Simulation Optimization: Theory and Practice

09:30 – 10:30: Reporting from Working Groups and Splitting into Smaller Groups

Coffee Break

11:00 – 12:00: Many Objectives in Stochastic Settings Chair: Christiane Tammer

- Anita Schoebel: Robust Multiobjective Optimization Problems – An Approach with Very Many Objective Functions
- Gabriele Eichfelder: A Multiobjective Trust Region Method for Expensive and Cheap Functions

Lunch

14:00 – 15:30: Small Working Groups

Coffee Break

16:00 – 17:00: Small Working Groups

17:00 – 18:00: Reporting from Small Working Groups
- General discussion and working group adaptations

Wednesday, January 15, 2020

09:00 – 09:30: Problems with Many Variables Chair: Patrick M. Reed
- Georges Fadel: Multi-Objective Topology Design of Functionally Graded Components

09:30 – 10:30: Small Working Groups

Coffee Break

11:00 – 12:00: Small Working Groups

Lunch

14:00: Group Foto (Outside)

14:05 – 15:30: Hiking Trip

Coffee Break

16:00 – 17:00: Small Working Groups

17:00 – 18:00: Participant Announcements

Thursday, January 16, 2020

09:00 – 09:30: Future Directions Chair: Juergen Branke
- Fritz Boekler: Complexity in Multiobjective Optimization
- Dimo Brockhoff: On Set-Indicator-Based Search: Using Single-Objective Solvers for Multiobjective Problems
- Pascal Kerschke: Chances and Challenges of Multimodality in Multi-Objective Continuous Optimization Problems

10:30 – 11:00: Coffee Break

11:00 – 12:00: State of Play

Lunch

Time for Individual Discussions (e.g. Offers-and-Needs Market)

15:30 – 16:00: Coffee Break

16:00 – 18:00: Small Working Groups

Dinner

20:00: Wine-and-Cheese Party (Music room)

Friday, January 17, 2020

09:00 – 10:30: Final Reporting from Working Groups

10:30 – 11:00: Coffee Break

11:00 – 12:00: Closing Session

Lunch

6 Topics of interest for participants for next Dagstuhl seminar

It has evolved as a tradition to jointly discuss future challenges and topics of particular interest for the EMO and MCDM community during the closing session on Friday. During this discussion the participants identified the following prevalent topics: Machine learning and data science (ML & DS) for multiobjective optimization and multiobjective optimization for ML & DS, neuro sciences, dynamic and adaptive systems, mixed models, ethical issues, social choice theory, and communicating multiobjective optimization. The organizers will use these suggestions as the basis for their discussion about possible topics for the next edition of this seminar series and for the preparation of a proposal for a continuation of the series.

7 Changes in the seminar organization body

7.1 Kathrin Klamroth and Günter Rudolph step down as co-organizers

As part of a continuing effort to renew the organizing board of this series of Dagstuhl seminars, Kathrin Klamroth and Günter Rudolph step down from the team of organizers, a role that they have held for three terms of office.

On behalf of all the participants of the seminar, Carlos Fonseca and Margaret Wiecek would like to express appreciation to Kathrin and Günter for their contributions and leadership that have been fundamental for the series success.

7.2 Welcome to Richard Allmendinger and Serpil Sayin

We are pleased to announce that our esteemed colleagues, Richard Allmendinger and Serpil Sayin, have agreed to serve as co-organizers for future editions of this Dagstuhl seminar series on Multiobjective Optimization. We look forward to collaborating with them in the near future.

Participants

- Richard Allmendinger
University of Manchester, GB
- Mickaël Binois
INRIA – Valbonne, FR
- Fritz Bökler
Universität Osnabrück, DE
- Jürgen Branke
University of Warwick, GB
- Dimo Brockhoff
INRIA Saclay – Palaiseau, FR
- Carlos A. Coello Coello
CINVESTAV – Mexico, MX
- Kerstin Dächert
Fraunhofer ITWM – Kaiserslautern, DE
- Matthias Ehrgott
Lancaster University, GB
- Gabriele Eichfelder
TU Ilmenau, DE
- Michael Emmerich
Leiden University, NL
- Georges Fadel
Clemson University – Clemson, US
- José Rui Figueira
IST – Lisbon, PT
- Carlos M. Fonseca
University of Coimbra, PT
- Andreia P. Guerreiro
IST – Lisbon, PT
- Jussi Hakanen
University of Jyväskylä, FI
- Susan R. Hunter
Purdue University, US
- Hisao Ishibuchi
Southern Univ. of Science and Technology – Shenzen, CN
- Andrzej Jaszkiewicz
Poznan University of Technology, PL
- Pascal Kerschke
Universität Münster, DE
- Kathrin Klamroth
Universität Wuppertal, DE
- Karl Heinz Küfer
Fraunhofer ITWM – Kaiserslautern, DE
- Arnaud Liefooghe
University of Lille, FR
- Manuel López-Ibánez
University of Manchester, GB
- Kaisa Miettinen
University of Jyväskylä, FI
- Sanaz Mostaghim
Universität Magdeburg, DE
- Boris Naujoks
TH Köln, DE
- Luís Paquete
University of Coimbra, PT
- Patrick M. Reed
Cornell University – Ithaca, US
- Enrico Rigoni
ESTECO SpA – Trieste, IT
- Günter Rudolph
TU Dortmund, DE
- Stefan Ruzika
TU Kaiserslautern, DE
- Serpil Sayin
Koc University – Istanbul, TR
- Anita Schöbel
Fraunhofer ITWM – Kaiserslautern, DE
- Britta Schulze
Universität Wuppertal, DE
- Pradyumn Kumar Shukla
KIT – Karlsruher Institut für Technologie, DE
- Ralph E. Steuer
University of Georgia, US
- Michael Stiglmayr
Universität Wuppertal, DE
- Christiane Tammer
Martin-Luther-Universität Halle-Wittenberg, DE
- Heike Trautmann
Universität Münster, DE
- Tea Tusar
Jozef Stefan Institute – Ljubljana, SI
- Daniel Vanderpooten
University Paris-Dauphine, FR
- Margaret M. Wiecek
Clemson University, US