CHALMERS | UNIVERSITY OF GOTHENBURG

PREPRINT 2011:12

Multi-Objective Design of a Combinatorial Structure

PETER LINDROTH
MICHAEL PATRIKSSON
ANN-BRITH STRÖMBERG

Department of Mathematical Sciences
Division of Mathematics
CHALMERS UNIVERSITY OF TECHNOLOGY
UNIVERSITY OF GOTHENBURG
Gothenburg Sweden 2011

Preprint 2011:12

Multi-Objective Design of a Combinatorial Structure

Peter Lindroth, Michael Patriksson,
Ann-Brith Strömberg

Department of Mathematical Sciences
Division of Mathematics
Chalmers University of Technology and University of Gothenburg
SE-412 96 Gothenburg, Sweden
Gothenburg, April 2011

Preprint 2011:12
ISSN 1652-9715

Matematiska vetenskaper
Göteborg 2011

Multi-Objective Design of a Combinatorial Structure

Peter Lindroth[*] Michael Patriksson[†] Ann-Brith Strömberg[†]

Abstract

Engineering design problems are often formulated as multi-objective optimization problems. We consider the problem of designing an optimal population of configurations, where the configurations are composed by common elements. Searching for a population of solutions that are good with respect to different combinations of the multiple objectives can be seen as a search for a clustering of the Pareto optimal set to the multi-objective optimization problem. Further, a natural wish is to use common parts to construct the population of design solutions. This paper proposes a (single-objective) optimization problem through which the clustering is performed in a way such that the resulting solutions approximate the Pareto optimal solution well, while at the same time the variables in the decision space are, by construction, required to be common. The procedure is applied to instances constructed from test functions from the literature with interesting results. The usefulness of applying the procedure to practical problems and what types of sensitivity analyses that can be performed are discussed and demonstrated. Suggestions are also made on how to adapt the developed methodology to simulation-based multi-objective optimization problems.

1 Introduction

A frequent wish in engineering design of mass-market products is to create a large variety of product configurations using just a few variants of each part. One example is trucks, another is kitchen cupboards.

In this paper we tackle the problem of deciding which variants to create by using a mathematical modeling approach with a strategy based on an underlying multi-objective optimization problem. We assume that the quality of a configuration is

[*]Corresponding author. Mathematical Sciences, Chalmers University of Technology, and Mathematical Sciences, University of Gothenburg, SE-412 96, Gothenburg, Sweden, and Volvo 3P, Chassis & Vehicle Dynamics, Chassis Strategies & Vehicle Analysis, SE-405 08, Gothenburg, Sweden. E-mail: peter.lindroth@volvo.com

[†]Mathematical Sciences, Chalmers University of Technology, and Mathematical Sciences, University of Gothenburg, SE-412 96, Gothenburg, Sweden. E-mail: {mipat,anstr}@chalmers.se

measured by a number of objective functions, each to be optimized, and that each configuration comprises a number of parts, each to be selected from a specific set of possible designs. The goal is to maximize the quality of the total product variety given the sets of possible parts. To our knowledge this approach is not apparent in the literature.

1.1 Motivation

A common property of engineering design problems is that they invoke a number of more or less conflicting criteria (or, objectives). Examples are *weight* ↔ *durability* and *cost* ↔ *feature level*. The objectives may be appreciated differently by different customers, for example depending on how and in which environment the product should be used and on the financial strength of the customer. This makes a multi-objective approach for solving such problems natural. Further, for cost and flexibility reasons, it is advantageous to design a small number of variants of each of the parts to be combined, forming a large number of possible configurations. Moreover, it is not enough to require that each configuration is good in itself; due to synergies of scale, the set of all produced configurations must be evaluated as a collective. We study the problem of how to systematically design the different variants such that the resulting collection of variants yields an, in a certain sense, optimal set of configurations.

We denote the technique to be introduced by *Implicit clustering*. Through traditional clustering or *Explicit clustering*, one partitions a set into groups, where objects belonging to the same group are similar, whereas objects belonging to different groups are dissimilar. An extensive overview of clustering techniques is found in [8]. Explicit clustering cannot be applied directly to our problem, since there is both a decision space and an objective space, both in which it is important where the resulting configurations are located. Clustering in the decision space only leads to no control of the distribution in the objective space, and clustering in the objective space only leads to a set of configurations without structure in the decision space. The technique presented in this paper resolves these problems by considering both spaces simultaneously, by the construction of a certain optimization problem.

1.2 Outline

In Section 2 we give a mathematical formulation of the design problem, discuss how to measure the quality of a set of configurations, and investigate the mathematical properties of the problem. We present a solution procedure in Section 3. In Section 4 we discuss the type of sensitivity analyses that can be performed for a practical problem and in Section 5 we solve some instances of the design problem using test functions from the literature. Then, in Section 6 we propose a procedure to be added to the solution process, which is reasonable if the objective functions are computationally intense. Finally, in Section 7 we conclude the paper and give some propositions for future work. These intend to make the solution strategy applicable to a larger class of design problems than that originally considered.

2 A mathematical formulation of the problem

We begin this section by defining multi-objective optimization. We then formulate our design problem, which, since it utilizes an underlying multi-objective optimization problem in its objective function, is called the *Multi-Objective Combinatorial Design Problem* (MOCDP). The objective function to use is discussed and some mathematical properties of MOCDP are analyzed.

2.1 Multi-objective optimization

A multi-objective (non-linear) optimization problem (MONP) can mathematically be formulated by the standard notation

$$\min_{\mathbf{x} \in X} \{f_1(\mathbf{x}), \ldots, f_k(\mathbf{x})\}, \tag{1}$$

where $\mathbf{x} \in \Re^n$ is a vector of decision variables, $X \subseteq \Re^n$ denotes the decision space, and each function $f_i : X \to \Re,\ i = 1 \ldots, k$, is an objective function to be minimized. We adopt the convention of letting the minimization operator apply to vectors. If the objective functions are at least partially in conflict, i.e., there exists no $\mathbf{x} \in X$ that simultaneously minimizes all k objectives, then an optimal solution to (1) is not well-defined since there exists no natural complete ordering between vectors. However, there exists a set of decision vectors, in which the best solution by rational judgements, provided the mathematical formulation, must be contained regardless of the relative importance of each single objective. This is the *Pareto optimal set* (or, equivalently, the *efficient* or *non-dominated* set).

Definition 2.1 *Given a set X of feasible vectors and a set $\{f_1, \ldots, f_k\}$ of objective functions to minimize, a vector $\mathbf{x}^* \in X$ is defined as* Pareto optimal *if there exists no other vector $\mathbf{x} \in X$ such that $f_i(\mathbf{x}) \leq f_i(\mathbf{x}^*),\ i = 1, \ldots, k$, and $f_j(\mathbf{x}) < f_j(\mathbf{x}^*)$ for at least one $j \in \{1, \ldots, k\}$. An objective vector $\mathbf{z}^* = \mathbf{f}(\mathbf{x}^*)$ is called* Pareto optimal *if the corresponding vector $\mathbf{x}^*$ is Pareto optimal. The set of all Pareto optimal vectors is denoted $\mathcal{P} \subseteq X$.*

Definition 2.2 *Given a set X of feasible vectors and a set $\{f_1, \ldots, f_k\} \in \Omega = \{\mathbf{f} \mid \mathbf{f} : \Re^n \to \Re^k\}$ of objective functions to minimize, the* Pareto operator *$P : \Re^n \times \Omega \to \Re^n$ is defined by $P(X, \mathbf{f}) = \mathcal{P}$.*

2.2 The multi-objective combinatorial design problem

Initially, we assume that the decision variables are continuous and that $X \subseteq \Re^n$ is defined by box constraints: $X = \prod_{j=1}^{n} X^j$, where $X^j = [l_j, u_j],\ j = 1, \ldots, n$, with $-\infty < l_j < u_j < \infty$. The assumption that each configuration consists of a fixed number of parts that are combined then translates to that each variant x_j of part j is to be selected from the interval $X^j = [l_j, u_j]$, and that each configuration has the representation $\mathbf{x} = (x_1, \ldots, x_n),\ x_j \in X^j,\ j = 1, \ldots, n$.

Assume that part j may have m_j different variants, and let the variants selected be represented by the variables $x_{j\ell},\ \ell = 1, \ldots, m_j$. The available configurations are then defined by the product set $X_D = \prod_{j=1}^{n} \{x_{j1}, \ldots, x_{jm_j}\}$. Such a configuration set is illustrated in Figure 1 where $n = 2, m_1 = 3$, and $m_2 = 2$.

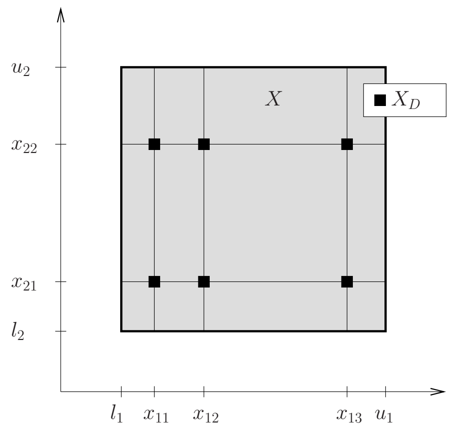


Figure 1: Illustration of a set of configurations $X_D \subset X$.

We wish to select the values of the variables $x_{j\ell},\ \ell = 1, \ldots, m_j,\ j = 1, \ldots, n$, such that the product set of configurations is, in a certain sense, optimal. We collect the decision variables in the vector

$$\mathbf{y} = (x_{11}, \ldots, x_{1m_1}, x_{21}, \ldots, x_{2m_2}, \ldots\ldots, x_{n1}, \ldots, x_{nm_n}) \in Y \subseteq \Re^{\sum_{j=1}^{n} m_j}, \qquad (2)$$

where

$$Y = \underbrace{X^1 \times \cdots \times X^1}_{m_1 \text{ factors}} \times \underbrace{X^2 \times \cdots \times X^2}_{m_2 \text{ factors}} \times \cdots\cdots \times \underbrace{X^n \times \cdots \times X^n}_{m_n \text{ factors}}, \qquad (3)$$

and denote the resulting set of available configurations as $X_D(\mathbf{y})$. The motivation behind the problem formulation is that with just $m := \sum_{j=1}^{n} m_j$ decision variables we decide on (the much larger number) $\prod_{j=1}^{n} m_j$ configurations. Further, let $\mathcal{Q}_{\mathcal{R}} : \Re^{\sum_{j=1}^{n} m_j} \to \Re$ be a function measuring the negative collective quality of a set of configurations (negative quality is utilized in order to obtain a minimization problem). The subscript $\mathcal{R}$ on the quality function represents a possible reference set for the configurations to be compared to.

We next introduce the (single-)objective optimization problem, the *Multi-Objective Combinatorial Design Problem* (MOCDP):

$$\mathcal{Q}^*_{\mathcal{R}}(m_1, \ldots, m_n) = \underset{\mathbf{y}}{\text{minimize}} \quad \mathcal{Q}_{\mathcal{R}}(\mathbf{y}) \tag{4a}$$

$$\text{subject to} \quad l_j \leq x_{j\ell} \leq x_{j,\ell+1} \leq u_j, \qquad \ell = 1, \ldots, m_j - 1, \quad j = 1, \ldots, n. \tag{4b}$$

The constraints (4b) ensure that the value of each decision variable is chosen from its feasible interval, i.e., $\mathbf{y} \in Y$ and thus $X_D \in X$; they also exclude solutions that are equivalent due to symmetry. For $m_j = 1$, the constraints (4b) should be replaced by $l_j \leq x_{j1} \leq u_j, \;\; j = 1, \ldots, n$.

The above formulation uses exactly m_j variants of part j (however not necessarily distinct). One could think that "at most" m_j variants would be more appropriate. These two formulations are, however, equivalent in the sense that their optimal $\mathcal{Q}_{\mathcal{R}}$-values are the same. The latter formulation is a relaxation of the former, and the optimal objective function $\mathcal{Q}^*_{\mathcal{R}}(m_1, \ldots, m_n)$ is monotonously decreasing with each $m_j, \; j = 1, \ldots, n$, for all reasonable quality functions $\mathcal{Q}_{\mathcal{R}}(\cdot)$.

Remark 2.3 *The formulation (4) can, with a suitable definition of $\mathcal{Q}_{\mathcal{R}}(\cdot)$, also be used for single-objective optimization design, with the aim of finding a combinatorial set of solutions which, as a collective, is robust with respect to variations or uncertainties in the underlying optimization problem.*

We next present an instance of MOCDP (without specifying the quality measure $\mathcal{Q}_{\mathcal{R}}(\cdot)$), which will be used for illustrative purposes in the paper.

Example 2.4 *Let the underlying MONP be defined by the decision space $X = [0, 1]^2$ and the objective functions $\mathbf{f}(\mathbf{x}) := \{f_1(\mathbf{x}), f_2(\mathbf{x})\}$, where*

$$f_1(\mathbf{x}) = \left(x_1 + \frac{1}{4}\right)^2 + \left(x_2 + \frac{1}{4}\right)^2, \tag{5a}$$

and

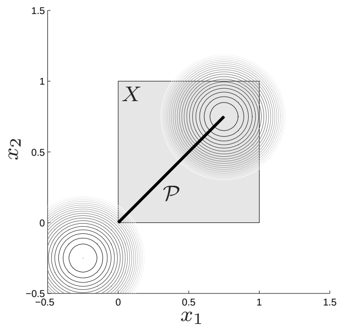
$$f_2(\mathbf{x}) = \left(x_1 - \frac{3}{4}\right)^2 + \left(x_2 - \frac{3}{4}\right)^2. \tag{5b}$$

The Pareto optimal set $\mathcal{P} = \left\{\mathbf{x} \in \Re^2 \mid 0 \leq x_1 = x_2 \leq \frac{3}{4}\right\}$. Let now $m_1 = m_2 = 2$, which leads to the vector $\mathbf{y} = (x_{11}, x_{12}, x_{21}, x_{22})$ of decision variables and the configuration set $X_D(\mathbf{y}) = \{(x_{11}, x_{21}), (x_{11}, x_{22}), (x_{12}, x_{21}), (x_{12}, x_{22})\}$. We get the MOCDP

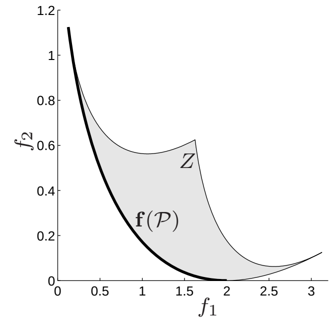
$$\text{minimize} \quad \mathcal{Q}_{\mathcal{R}}(\mathbf{y}), \tag{6a}$$

$$\text{subject to} \quad 0 \leq x_{j1} \leq x_{j2} \leq 1, \qquad j = 1, 2. \tag{6b}$$

Figure 2 illustrates (a) the decision space and (b) the objective space of the underlying MONP.



(a) The decision space, X, level curves for the two objective functions, and the Pareto optimal set $\mathcal{P} \subseteq X$.



(b) The objective space, $Z = \mathbf{f}(X)$, and the image, $\mathbf{f}(\mathcal{P})$, of the Pareto optimal set.

Figure 2: Illustration of the design and objective spaces of the MONP in Example 2.4. The Pareto optimal set $\mathcal{P} \subseteq X$, and the image of the Pareto optimal set $\mathbf{f}(\mathcal{P}) \subseteq Z = \mathbf{f}(X)$, are marked in black in the respective figures.

2.3 Measuring the quality of the set of configurations

It is not obvious how to define the quality function $\mathcal{Q}_{\mathcal{R}}(\cdot)$, but it is clear that $\mathbf{f}(\mathcal{P})$ in some suitable sense should be approximated by $\mathbf{f}(X_D)$. As noted e.g. in [16, 18, 3, 21] there is no standard technique in the literature for measuring the quality of approximate Pareto sets, and for many of the intuitive measures one can easily construct examples that shows good results for obviously bad approximations and vice versa.

Two measures that have been designed for evaluation of metaheuristics for multi-objective optimization problems, and which make sense also in our application, are $Dist1_{\mathcal{R}}$ and $Dist2_{\mathcal{R}}$, proposed in [4] and also used in e.g. [19]. We give the definitions of $Dist1_{\mathcal{R}}$ and $Dist2_{\mathcal{R}}$ below. We will replace $\mathcal{Q}_{\mathcal{R}}(\cdot)$ by $Dist1_{\mathcal{R}}$, $Dist2_{\mathcal{R}}$, or with a combination of these. These two metrics reward approximate sets (in our case X_D) that comprise points that are near-Pareto optimal while being evenly distributed over the Pareto set. For the evaluation of the approximate set X_D, both $Dist1_{\mathcal{R}}$ and $Dist2_{\mathcal{R}}$ require a reference set $\mathcal{R} \subset X$, which should be a discrete approximation of the true Pareto optimal set $\mathcal{P}$. If $\mathcal{P}$ is known, $\mathcal{R}$ can be an evenly spread discrete subset of $\mathcal{P}$, which is the ideal situation. If $\mathcal{P}$ is not known, $\mathcal{R}$ may consist of (a subset of) the non-dominated points found using any solution method for multi-objective optimization.

A high quality of a set X_D means that to each vector $\mathbf{x}^r \in \mathcal{R}$ there is a vector $\mathbf{x}^d \in X_D$ *close* to $\mathbf{x}^r$. The closeness, $c_{\mathbf{w}}(\mathbf{x}^r, \mathbf{x}^d)$, of the vectors $\mathbf{x}^r \in \mathcal{R}$ and $\mathbf{x}^d \in X_D$ is a non-symmetric measure defined as

$$c_{\mathbf{w}}(\mathbf{x}^r, \mathbf{x}^d) = \max_{i \in \{1,\ldots,k\}} \left\{ \max \left\{ 0, w_i \big(f_i(\mathbf{x}^d) - f_i(\mathbf{x}^r) \big) \right\} \right\}, \quad \mathbf{x}^r \in \mathcal{R},\ \mathbf{x}^d \in X_D, \tag{7}$$

where $w_i \geq 0$ is the weight assigned to objective i, $i = 1, \ldots, k$, and $\mathbf{w} = \{w_1, \ldots, w_k\}$. In contrast to the original definition in [4] we will from now on replace X_D in the definition (7) of closeness by $\mathcal{P}_D = P(X_D, \mathbf{f})$, since a dominated solution is never preferred to a non-dominated solution by a rational decision maker. Thus, the closeness of a point $\mathbf{x}^r \in \mathcal{R}$ to a point $\mathbf{x}^d \in X_D$, where $\mathbf{x}^r$ is (weakly) dominated by $\mathbf{x}^d$, (i.e., $\mathbf{f}(\mathbf{x}^d) \leq \mathbf{f}(\mathbf{x}^r)$), is defined to be zero. Otherwise, the closeness is given by the maximum weighted deterioration of an objective value over the set of objective functions. The weights in the expression (7) are set to

$$w_i = \frac{1}{\max\limits_{\mathbf{x}\in\mathcal{R}} f_i(\mathbf{x}) - \min\limits_{\mathbf{x}\in\mathcal{R}} f_i(\mathbf{x})}, \quad i = 1, \ldots, k, \tag{8}$$

i.e., inversely proportional to the range of f_i over $\mathcal{R}$ [1]. An illustration of the closeness between two points $\mathbf{x}^r$ and $\mathbf{x}^d$ is given in Figure 3.

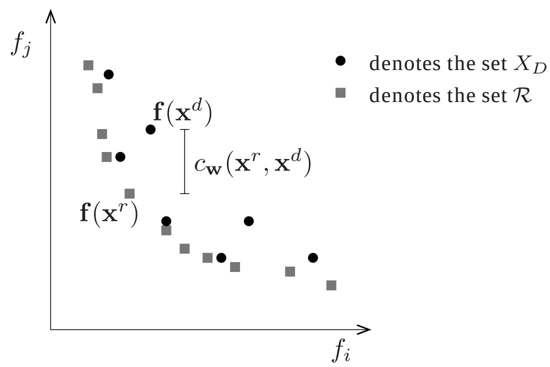


Figure 3: An illustration of the closeness between two points $\mathbf{x}^r$ and $\mathbf{x}^d$ according to the definition (7). Here, $w_i = w_j = 1$.

The $Dist1_{\mathcal{R}}$ measure yields information on the average distance from a point $\mathbf{x}^r \in \mathcal{R}$ to its closest point in X_D, and is defined as

$$Dist1_{\mathcal{R}}(\mathbf{y}) = \frac{1}{|\mathcal{R}|} \sum_{\mathbf{x}^r \in R} \left(\min_{\mathbf{x}^d \in X_D(\mathbf{y})} c_{\mathbf{w}}(\mathbf{x}^r, \mathbf{x}^d) \right). \tag{9}$$

Correspondingly, $Dist2_{\mathcal{R}}$ yields information on the maximum distance and is defined as

$$Dist2_{\mathcal{R}}(\mathbf{y}) = \max_{\mathbf{x}^r \in \mathcal{R}} \left\{ \min_{\mathbf{x}^d \in X_D(\mathbf{y})} c_{\mathbf{w}}(\mathbf{x}^r, \mathbf{x}^d) \right\}. \tag{10}$$

Note that if the points in $\mathcal{R}$ are more dense in some region of X, $Dist1_{\mathcal{R}}$ will lead to a biased result, since the denser part of the approximation will possess a larger weight in the sum.

[1] Assumed is that the range is non-zero which is a reasonable assumption for practical problems. If this assumption is not valid, then a positive constant could be added to the denominator in (7) or an estimation could be made of the "scale" of each objective over the interesting region.

Remark 2.5 *The $Dist1_{\mathcal{R}}$ and $Dist2_{\mathcal{R}}$ measures are adopted from the evaluation of meta-heuristics for applications for which $\mathcal{P}$ is not known. If there exists an explicit expression for $\mathcal{P}$ one can choose $\mathcal{R} = \mathcal{P}$ and replace the sum in $Dist1_{\mathcal{R}}$ with an integral. This is also possible if there exists a function describing $\mathcal{R}$, e.g. by interpolating the non-dominated points found by some (approximate) solution method.*

2.4 Some mathematical properties of MOCDP

We are interested in MOCDP applied to practical problems. The purpose of this section is to analyze enough mathematical properties of MOCDP such that a suitable solution method can be proposed for such problems.

Proposition 2.6 *MOCDP with the quality function $\mathcal{Q}_{\mathcal{R}}(\cdot)$ being either $Dist1_{\mathcal{R}}$ or $Dist2_{\mathcal{R}}$ is continuous if the underlying MONP is continuous.*

Proof. All f_i's are continuous since MONP is continuous and the $\max$- and $\min$-operators in $Dist1_{\mathcal{R}}$, $Dist2_{\mathcal{R}}$ are continuous. A composition of continuous function is continuous. The feasible set of MOCDP is continuous and, hence, so is the problem. ■

Since the closeness function defined in (7) is non-differentiable for $\mathbf{y} \in Y$ such that $c_{\mathbf{w}}(\mathbf{x}^r, \mathbf{x}^d) = c_{\mathbf{w}}(\mathbf{x}^s, \mathbf{x}^d),\ r, s \in \mathcal{R}$ for some $\mathbf{x}^d \in X_D(\mathbf{y})$ (i.e. when $\mathbf{x}^d$ changes its nearest point in $\mathcal{R}$), we have the following result:

Proposition 2.7 *If $\mathcal{R}$ a discrete set of points then MOCDP is non-differentiable.*

We continue with investigating convexity properties of MOCDP. If MOCDP is convex, then a local optimum is a global optimum and the problem can be solved to global optimality using a local optimization algorithm. Unfortunately, as shown below, this is not the case, even under very strong assumptions on the underlying MONP.

Example 2.8 *Recall Example 2.4. The underlying MONP is convex since both objective functions are convex and the feasible decision space X is a convex set. Let the reference set be $\mathcal{R} = \{(\frac{1}{10}, \frac{1}{10}), (\frac{6}{10}, \frac{6}{10})\}$. Then $\mathcal{R} \subset \mathcal{P} = \big\{(x_1, x_2) \in \Re^2 |\ 0 \le x_1 = x_2 \le \frac{3}{4}\big\}$. A globally optimal solution to MOCDP is then $\mathbf{y}^* = (x^*_{11}, x^*_{12}, x^*_{21}, x^*_{22}) = (\frac{1}{10}, \frac{6}{10}, \frac{1}{10}, \frac{6}{10})$ with $\mathcal{Q}_{\mathcal{R}}(\mathbf{y}^*) = 0$ since the number of decision variables is enough to meet all elements in $\mathcal{R}$*[2]*. However, experimenting with a local optimizer shows that it is possible to end up in a local minimum with a positive quality measure. The instance of MOCDP is clearly non-convex which is exemplified below. Let*

$$
\begin{aligned}
\mathbf{y}^1 &= (0, 0.50, 0.35, 0.70)^{\mathsf{T}},\\
\mathbf{y}^2 &= (0, 0.70, 0.35, 0.70)^{\mathsf{T}},\\
\lambda &= 0.5.
\end{aligned}
$$

[2] This is always true when each $m_j,\ j = 1, \ldots, n$, is larger than the number of distinct values in $\mathcal{R}$ in the corresponding dimension.

A necessary condition for a function to be convex is that a linear interpolation of two function values never is lower than the function itself at the corresponding interpolation between the decision variables. We have a counterexample for convexity of MOCDP for both $Dist1_{\mathcal{R}}$ and $Dist2_{\mathcal{R}}$, since

$$0.149 \approx \mathcal{Q}_{\mathcal{R}}(\lambda \mathbf{y}^1 + (1-\lambda)\mathbf{y}^2) \nleq \lambda \mathcal{Q}_{\mathcal{R}}(\mathbf{y}^1) + (1-\lambda)\mathcal{Q}_{\mathcal{R}}(\mathbf{y}^2) \approx 0.117$$

for $\mathcal{Q}_{\mathcal{R}}(\cdot) = Dist1_{\mathcal{R}}$ and

$$0.150 \approx \mathcal{Q}_{\mathcal{R}}(\lambda \mathbf{y}^1 + (1-\lambda)\mathbf{y}^2) \nleq \lambda \mathcal{Q}_{\mathcal{R}}(\mathbf{y}^1) + (1-\lambda)\mathcal{Q}_{\mathcal{R}}(\mathbf{y}^2) \approx 0.148 \tag{11}$$

for $\mathcal{Q}_{\mathcal{R}}(\cdot) = Dist2_{\mathcal{R}}$.

The disappointing non-convexity result is obviously true also when the quality function $\mathcal{Q}_{\mathcal{R}}$ is a convex combination of $Dist1_{\mathcal{R}}$ and $Dist2_{\mathcal{R}}$. As far as we know there exists no reasonable quality function that preserves the convexity property.

Assuming that the underlying MONP is continuous, we conclude that MOCDP is in general continuous, non-differentiable, and non-convex; we conclude that some suitable global optimizer is needed to solve it.

Our interest in the following result is motivated by the fact that we intend to use penalty-based methods in the solution procedure to handle constraints. The result requires some weak assumptions that are quite vaguely formulated. The important point is that it is likely to hold for practical problems, which are of our interest. Figure 4 helps to understand the result.

Proposition 2.9 *Let $X = \prod_{j=1}^{n}[l_j, u_j]$ be a box-constrained decision space to the underlying MONP to a MOCDP. Assume that MOCDP has a reference set $\mathcal{R}$ which is sufficiently large compared to the cardinality of the configuration set X_D. Assume further that $\mathcal{R}$ is sufficiently spread in X, that $\mathbf{f}(\mathcal{R})$ is sufficiently spread in $\mathbf{f}(X)$, and that the objectives $f_1, \ldots, f_k$, are sufficiently well-behaved. Then to MOCDP, there exists optimal solutions $\mathbf{y}^* \in \text{int}(Y)$ or, equivalently, $X_D(\mathbf{y}^*) \subset \text{int}(X)$.*

Proof. If $\mathcal{R}$ is sufficiently large and spread, then at an optimal solution $\mathbf{y}^* \in Y$, many $\mathbf{x}^r \in \mathcal{R}$ will share the same $\mathbf{x}^d \in X_D(\mathbf{y}^*)$ as their nearest point in X_D. In particular, each $\mathbf{x}^r \in \partial X$ has other vectors $\hat{\mathbf{x}}^r \in \text{int}(X)$ with the same nearest point $\mathbf{x}^d \in X_D$. Then, if $f_1, \ldots, f_k$, are sufficiently well-behaved there will be an optimal solution $\mathbf{y}^*$ where each $\mathbf{x}^d \notin \partial X$, or, equivalently, $\mathbf{y} \in \text{int}(Y)$. This because a small movement of a $\mathbf{x}^d \in \partial X$ out from the boundary will not decrease the maximum closeness between each $\mathbf{x}^r \in \mathcal{R}$ to its nearest $\mathbf{x}^d \in X_D$. ∎

Remark 2.10 *A similar result as in Proposition* (2.9) *with an analogous proof can be formulated for the symmetry-breaking constraints $x_{j\ell} \leq x_{j,\ell+1}$ in* (4b). *In an optimal solution to MOCDP, there are (likely to exist) optimal solutions where the constraints are not active.*

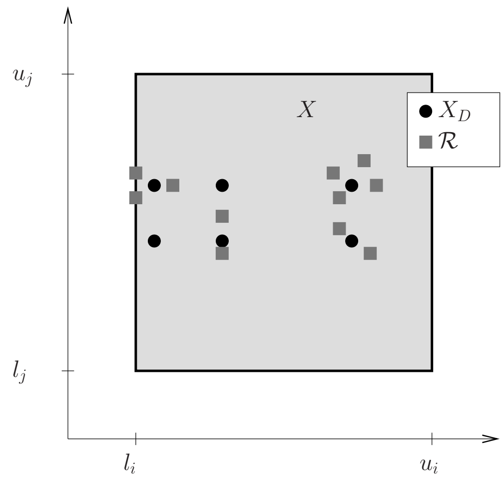


Figure 4: A rough illustration of the fact that the optimal configuration likely will lie in the interior of the decision space X to the underlying MONP for most practical problems. For sufficiently well-behaved objective functions, the closeness between the upper-left-most $\mathbf{x}^d \in X_D$ and its neighbor $\mathbf{x}^r \in \text{int}(X)$ to the right, will not be minimal if $\mathbf{x}^d$ is moved to the left onto the boundary.

3 A solution procedure

First, observe that for a real application the numbers m_j, $j = 1, \dots, n$, of variants may often be decision variables. We suggest to treat them as input parameters, and to solve the problem for different values of m_j to study the sensitivities, i.e., how the optimal solution to MOCDP varies with changes in m_j (cf. Section 4).

We propose a two-step method for solving MOCDP. If there is no problem-specific distance measure known for the evaluation of an approximate Pareto optimal set, we suggest using either of the functions $Dist1_{\mathcal{R}}$, $Dist2_{\mathcal{R}}$ or a combination of these. In the first step of the procedure, a representation of the Pareto optimal set of the underlying MONP should be found. The method for this is arbitrary, and should be chosen with respect to the actual MONP. If this is a non-convex problem with unknown problem characteristics, some evolutionary method [5] might be a reasonable choice.

In the second step of the procedure some global optimization method should be used to find the optimal decision variables $\mathbf{y}^* \in Y$ given the reference set found in step 1. Figure 5 illustrates the two steps of the solution process.

To handle the box and symmetry-breaking constraints (4b) we have used a modified barrier method where linear / logarithmic penalties are added to the objective instead of using constraints. An illustration of modified linear / logarithmic penalty functions is given in Figure 6. By "modified" we mean here that the logarithmic

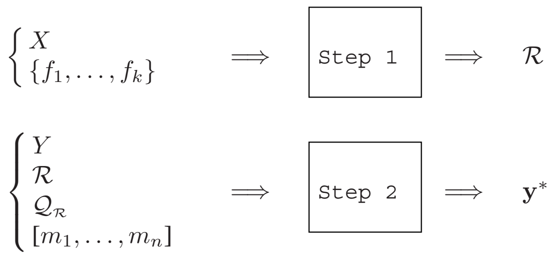


Figure 5: The solution process is divided into two steps. In step 1 , the underlying MONP is solved using some multi-objective optimization solver and $\mathcal{R}$ is defined. In step 2 a global optimizer is used to minimize $\mathcal{Q}_{\mathcal{R}}$ over Y given the reference set $\mathcal{R}$ and the number of allowed variants m_j in each dimension $j,\ j = 1, \ldots, n$.

penalties which lead to the objective function being undefined in parts of the domain, are replaced by linear functions near and outside the boundaries. For example, the optimization problem

$$\begin{aligned} \text{minimize} \quad & f(x), \\ \text{subject to} \quad & x \leq u, \\ & x \in \Re, \end{aligned} \tag{12}$$

is replaced by

$$\begin{aligned} \text{minimize} \quad & f(x) - \nu \left(1_{(-\infty,u-\epsilon\}}(x) \log(u-x) + 1_{(u-\epsilon,+\infty)}(x) \left(\frac{x-u}{\epsilon} + 1 - \log \epsilon \right) \right), \\ \text{subject to} \quad & x \in \Re, \end{aligned} \tag{13}$$

where $1_S(x)$ is an indicator function, i.e., equal to one if $x \in S$ and zero otherwise, and ϵ is the distance from the boundary where the logarithmic function is replaced by a linear function. ν is a penalty parameter. For a sufficiently well-behaving function f a globally optimal solution to (13) converges towards a globally optimal solution to (12).

Due to the result in Proposition 2.9 we have good reasons to believe that an optimal solution $\mathbf{y}^*$ lies in the interior of Y and where the symmetry-breaking constraints are non-active. Here, the added penalty does not affect the objective function that much even for a penalty parameter with a positive value of significant size.

Since neither step 1 nor step 2 in general will reach a point, where it is not possible to improve anymore, the maximum allowed computational time of both steps must be set. In step 1, the longer time the algorithm is permitted to work, the more accurate representation of $\mathcal{P}$ is generally obtained. In step 2, the longer the global algorithm is applied, the higher the probability of finding a good solution. However, for sensitivity studies (cf. Section 4) step 1 only has to be performed once.

The fact that the solution algorithm is partitioned into two steps can be taken advantage of for problems with expensive function evaluations, e.g., given by com-

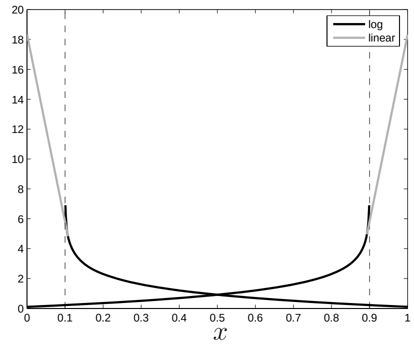


Figure 6: Example of a penalty function using a modified barrier method added to a problem with box constraints, $x \in [0.1, 0.9]$. The logarithmic function transcends smoothly into a linear function at a distance ϵ from the borders. Here $\epsilon = 0.008$.

putationally intensive simulations. The evaluations of the objective function made in step 1 can then be used for the computation of explicit response surfaces to be used in step 2, cf. Section 6 for more details.

So far we have not specified which quality function to use. The two possibilities $Dist1_{\mathcal{R}}$ and $Dist2_{\mathcal{R}}$ seem reasonable for general MOCDP's. However, it is possible that the MOCDP concerns some special application for which there is some other better measure, i.e., leads to solutions that are more attractive from a practical standpoint.

The main disadvantage of $Dist1_{\mathcal{R}}$ is its sensitivity for the distribution of the points in $\mathcal{R}$. It is a well-known fact that evolutionary algorithms often output solution sets $\hat{\mathcal{P}}$ (approximate Pareto optimal sets) whose "density" varies heavily and possesses an a priori unknown distribution. Hence, $Dist1_{\mathcal{R}}$ might not be a good choice for the quality measure. The main disadvantage of $Dist2_{\mathcal{R}}$ is that it is a worst-case measure, only considering the point in $\mathcal{R}$ with the largest distance (closeness) to a point in $\mathcal{P}_D$. As a special case, it assigns the same quality to a set $\mathcal{P}_D$ from which every point in $\mathcal{R}$ lies at distance d as to a set where $\mathcal{P}_D$ that coincides with $\mathcal{R}$ but for a single point in $\mathcal{R}$ at distance d from $\mathcal{P}_D$.

The two proposed quality functions $Dist1_{\mathcal{R}}$ and $Dist2_{\mathcal{R}}$ are obviously correlated; their characteristics are, however, different. By using a convex combination of the quality functions, their disadvantages can be diminished.

4 Sensitivity analysis

When modeling and solving a practical problem as a MOCDP, an important and interesting analysis is to study how the number m_j of variants, $j = 1, \ldots, n$, in the n decision dimensions of the MONP affects the optimal solution. For example, if there is a limitation on the total number of variants allowed, then it is critical to investigate how sensitive the resulting objective value is to the distribution of the variants in the respective dimensions. For the example in Figure 1, could it be favourable to use three variants in the x_2 dimension and just two in the x_1 dimension? Or perhaps it might be best to use only one variant in the x_1 dimension and four in the x_2 dimension. Further, if the costs of adding a new variant, or the savings of removing one, in a certain dimension, is known, then this information could be used when designing a good set of configurations.

Assume that the set of variants at some point in time is given by the vector $\hat{\mathbf{y}}$ and that the cost of adding a variant in dimension j is δ_j. Observe that the optimal objective value to MOCDP, $\mathcal{Q}^*_{\mathcal{R}}$, is a function of the underlying MONP (defined by $\mathbf{f}$ and X) together with the number of allowed variants $m_j,\ j = 1, \ldots, n$:

$$\mathcal{Q}^*_{\mathcal{R}} = \mathcal{Q}^*_{\mathcal{R}}(\mathbf{f}, X, m_1, \ldots, m_j, \ldots, m_n).$$

The decision to make is whether the quality increase is worth the extra cost, i.e., if the profits gained by reducing the quality measure with

$$\mathcal{Q}^*_{\mathcal{R}}(\mathbf{f}, X, m_1, \ldots, m_j + 1, \ldots, m_n) - \mathcal{Q}^*_{\mathcal{R}}(\mathbf{f}, X, m_1, \ldots, m_j, \ldots, m_n)$$

is larger than the cost δ_j. An analogous study can be made for a possible removal of variants by comparing the savings for removing the variants with the difference of the quality measure when decreasing m_j.

An assumption made above, which may not be valid in many real applications, is that the cost for modifying the current set of variants is zero. For many practical problems, there is a fixed set of current variants and costs arise when adding variants to the fixed set. A sensitivity analysis of MOCDP could be used for this case as well. The $\mathcal{Q}_{\mathcal{R}}$ measure has to be computed for the current setup. Then MOCDP is solved with m_j equal to the number of added variants in each dimension where the current variants specified by $\hat{\mathbf{y}}$ are added to $\mathbf{y}$ in the computation of the configurations. The improvement in $\mathcal{Q}^*_{\mathcal{R}}$ must now be compared to the cost of adding variants.

To investigate whether an existing variant should be removed is not possible without calculating $\mathcal{Q}_{\mathcal{R}}$ for all possible choices variant removals. That is to say, to analyze whether one variant should be added, n problems need to be solved, one for each dimension. To analyze whether one variant should be removed, $m = \sum_{j=1}^{n} m_j$ problems have to be solved, one for each current variant. The latter problems, however, are very easy since there are no decision variables at all. What has to be done is to compute the quality function $\mathcal{Q}_{\mathcal{R}}$ for the $\sum_{j=1}^{n} m_j$ reduced configuration sets.

5 Numerical experiments

The purpose of this section is to exemplify how the MOCDP can be utilized, by presenting some selected numerical experiments. By using the standard vector-valued test function *kursawe* [15] as the underlying MONP, MOCDP has been formulated and solved with the procedure proposed in Section 3 for different values of

$m_j,\ j = 1, \dots, n$. We have used a box constrained variable space of dimension three in the MONP.

The objective functions and the feasible region are given by (14).

$$
\begin{aligned}
f_1(\mathbf{x}) &= \sum_{i=1}^{n-1} \left(-10 \exp \left(-0.2 \sqrt{x_i^2 + x_{i+1}^2} \right) \right), \\
f_2(\mathbf{x}) &= \sum_{i=1}^{n} \left(|x_i|^{0.8} + 5 \sin^3(x_i) \right), \qquad (14) \\
\mathbf{x} &\in [-5, 5]^n.
\end{aligned}
$$

The *kursawe* function is a standard test function for the evaluation of multi-objective evolutionary algorithms (see [11] for an extensive review).

In step 1 of the solution procedure—to find a representation $\mathcal{R}$ of the Pareto optimal set—we have used `multiOb` [10], a population-based evolutionary algorithm. Examples of other evolutionary-based algorithms for solving MONP's that could be used are `NCGA` [20] and `NSGA-II` [6].

In step 2 of the solution process—in which a global optimization is to be performed—we have chosen the algorithms `DIRECT` [14, 7] and `NEWUOA` [17] to be used in sequence. The former is a space-filling algorithm sampling the decision space around points that either have low objective values or are far from already sampled points. The termination criterion for `DIRECT` can be the number of space-dividing iterations or the number of function evaluations. The output from the algorithm—the best point measured so far—is then provided as a starting point for `NEWUOA`. This is a local optimization algorithm for unconstrained derivative-free single-objective optimization based on quadratic approximations of the objective function.

An approximation of the image of the Pareto optimal set (found by applying `multiOb` with 2000 generations and with a population size of 4000) is shown in Figure 7.

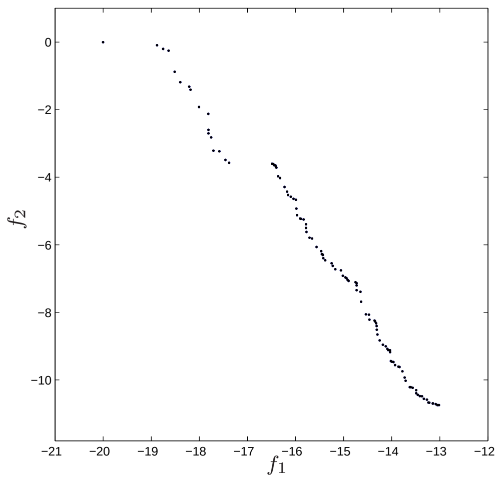


Figure 7: An approximate Pareto front for the test problem *kursawe* with $n = 3$.

Many other test problems in the literature are limited to two decision variables and/or have a Pareto optimal set that has a special structure in X that seems unnatural for a practical problem. To generate more test problems we have kept the objectives of (14) but chosen to rotate the decision space for the first objective. That is, we let the objectives be $\{f_1(A^p\mathbf{x}), f_2(\mathbf{x})\}$, $p = 0, \ldots, 3$, where A^p denote the rotation matrices

$$A^0 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \text{(no rotation)}, \quad A^1 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & -1 & 0 \end{bmatrix} \text{(rotation around } x_1\text{)},$$
$$A^2 = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ -1 & 0 & 0 \end{bmatrix} \text{(rotation around } x_2\text{)}, \quad A^3 = \begin{bmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \text{(rotation around } x_3\text{)}. \tag{15}$$

This yields four corresponding MONP's, denoted by MONP^p, $p = 0, \ldots, 3$. In the numerical experiments we have for each MONP^p tested all combinations of numbers $[m_1, m_2, m_3]$ of allowed variants in the set $M = \{\mathbf{m} \in \mathbb{N}^3 \mid 1 \leq \sum_{j=1}^{3} m_j \leq 8\}$. We assume here that the variants in the three dimensions are equally expensive and that the important issue is the total number of variants used.

For all numerical results presented below the quality measure

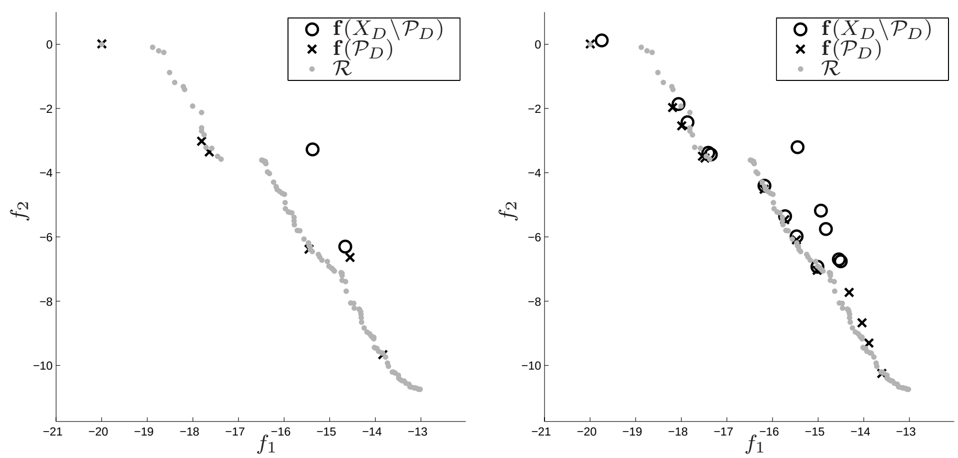
$$\mathcal{Q}_{\mathcal{R}} = 0.01 Dist1_{\mathcal{R}} + 0.99 Dist2_{\mathcal{R}}.$$

The first step of the solution procedure is to find a reference set $\mathcal{R}$ to use in the second step and in the sensitivity analyses. We applied `multiOb` to each MONP^p

using 2000 generations and a population size of 4000 to generate the corresponding reference set $\mathcal{R}^p$, selected to be all found non-dominated points.

In the second step of the procedure we applied `DIRECT` with a termination criterion defined as a maximum number of function evaluations. The best point found by `DIRECT` was then used as a starting point for `NEWUOA`.

The Figures 8 (a) and (b) show the objective values for the solutions found to MOCDP for MONP^0 using $\mathbf{m} = [2, 2, 2]$ and $\mathbf{m} = [3, 3, 3]$, respectively. It is interesting to note that, even if this it not the aim, our algorithm in the second solution step manages to find solutions that dominate parts of $\mathcal{R}$ (at $(f_1, f_2) \approx (-18, -3)$ both in Figure 8 (a) and (b)). It is not a large part of $\mathcal{R}$ that is dominated. However, the computational time for finding $\mathcal{R}$ in the first solution step was around 10 minutes while the time for the second step of solving the MOCDP was only around 20 seconds. It may be possible to create a new class of algorithms for solving multi-objective optimization problems based on ideas similar to ours. Another interesting observation is that the resulting solutions in X_D seem to form a good approximation at the "knee" regions of $\mathbf{f}(\mathcal{R})$ which have the character that a small improvement in either objective will cause a large deterioration in the other (see Figure 8(a)). The knee regions are the most interesting solutions for decision makers whose evaluation of the trade-offs between the conflicting objectives are relatively constant.



(a) 2 variants in each dimension.

(b) 3 variants in each dimension.

Figure 8: The objective space with the solutions found using MONP^0 as the underlying problem and with $\mathbf{m} = [2, 2, 2]$ and $\mathbf{m} = [3, 3, 3]$, respectively.

The Tables 1–4 contain results on the quality measures found together with their corresponding variant distributions. Due to space limitations we present the results for the subset of combinations for which $\max_{i,j}\{|m_i - m_j|\} \leq 1$ only. In each row, i.e., for each value of m, the best solution(s) is (are) written in bold face.

$\sum_j m_j$	$\mathbf{m}$	$\mathcal{Q}^*_{\mathcal{R}}$	$\mathbf{m}$	$\mathcal{Q}^*_{\mathcal{R}}$	$\mathbf{m}$	$\mathcal{Q}^*_{\mathcal{R}}$
3	[1 1 1]	**0.555**				
4	[1 1 2]	**0.344**	[1 2 1]	**0.344**	[2 1 1]	**0.344**
5	[1 2 2]	**0.222**	[2 1 2]	0.331	[2 2 1]	**0.222**
6	[2 2 2]	**0.100**				
7	[2 2 3]	0.064	[2 3 2]	0.081	[3 2 2]	**0.063**
8	[2 3 3]	0.058	[3 2 3]	**0.046**	[3 3 2]	0.058
9	[3 3 3]	**0.046**				
10	[3 3 4]	0.048	[3 4 3]	**0.045**	[4 3 3]	0.048
11	[3 4 4]	**0.040**	[4 3 4]	0.042	[4 4 3]	0.048
12	[4 4 4]	**0.040**				

Table 1: Numerical results for the solution of MOCDP with different numbers of variants of the three underlying decision variables. The underlying multi-objective problem is MONP^0.

$\sum_j m_j$	$\mathbf{m}$	$\mathcal{Q}^*_{\mathcal{R}}$	$\mathbf{m}$	$\mathcal{Q}^*_{\mathcal{R}}$	$\mathbf{m}$	$\mathcal{Q}^*_{\mathcal{R}}$
3	[1 1 1]	**0.558**				
4	[1 1 2]	0.345	[1 2 1]	**0.344**	[2 1 1]	**0.344**
5	[1 2 2]	**0.221**	[2 1 2]	**0.221**	[2 2 1]	0.329
6	[2 2 2]	**0.104**				
7	[2 2 3]	0.072	[2 3 2]	**0.061**	[3 2 2]	**0.061**
8	[2 3 3]	0.061	[3 2 3]	0.061	[3 3 2]	**0.046**
9	[3 3 3]	**0.046**				
10	[3 3 4]	0.047	[3 4 3]	**0.046**	[4 3 3]	**0.046**
11	[3 4 4]	0.046	[4 3 4]	0.045	[4 4 3]	**0.041**
12	[4 4 4]	**0.041**				

Table 2: Numerical results for the solution of MOCDP with different numbers of variants of the three underlying decision variables. The underlying multi-objective problem is MONP^1.

$\sum_j m_j$	$\mathbf{m}$	$\mathcal{Q}^*_{\mathcal{R}}$	$\mathbf{m}$	$\mathcal{Q}^*_{\mathcal{R}}$	$\mathbf{m}$	$\mathcal{Q}^*_{\mathcal{R}}$
3	[1 1 1]	**0.551**				
4	[1 1 2]	**0.342**	[1 2 1]	**0.342**	[2 1 1]	**0.342**
5	[1 2 2]	**0.220**	[2 1 2]	0.331	[2 2 1]	**0.220**
6	[2 2 2]	**0.104**				
7	[2 2 3]	**0.064**	[2 3 2]	0.076	[3 2 2]	**0.064**
8	[2 3 3]	0.064	[3 2 3]	**0.049**	[3 3 2]	0.064
9	[3 3 3]	**0.053**				
10	[3 3 4]	**0.049**	[3 4 3]	0.053	[4 3 3]	**0.049**
11	[3 4 4]	**0.049**	[4 3 4]	**0.049**	[4 4 3]	0.050
12	[4 4 4]	**0.049**				

Table 3: Numerical results for the solution of MOCDP with different numbers of variants of the three underlying decision variables. The underlying multi-objective problem is MONP^2.

$\sum_j m_j$	$\mathbf{m}$	$\mathcal{Q}^*_{\mathcal{R}}$	$\mathbf{m}$	$\mathcal{Q}^*_{\mathcal{R}}$	$\mathbf{m}$	$\mathcal{Q}^*_{\mathcal{R}}$
3	[1 1 1]	**0.557**				
4	[1 1 2]	**0.344**	[1 2 1]	**0.344**	[2 1 1]	**0.344**
5	[1 2 2]	0.329	[2 1 2]	**0.221**	[2 2 1]	**0.221**
6	[2 2 2]	**0.101**				
7	[2 2 3]	**0.065**	[2 3 2]	**0.065**	[3 2 2]	0.084
8	[2 3 3]	**0.049**	[3 2 3]	0.065	[3 3 2]	0.065
9	[3 3 3]	**0.058**				
10	[3 3 4]	0.049	[3 4 3]	**0.048**	[4 3 3]	0.058
11	[3 4 4]	**0.041**	[4 3 4]	0.049	[4 4 3]	0.048
12	[4 4 4]	**0.046**				

Table 4: Numerical results for the solution of MOCDP with different numbers of variants of the three underlying decision variables. The underlying multi-objective problem is MONP^3.

An important point that has to be kept in mind is that this optimization problem is non-convex and non-linear. Thus, there is no guarantee for the optimality of the solutions found. Study Table 4 and compare for $\mathbf{m} = [3, 4, 4]$ and $\mathbf{m} = [4, 4, 4]$. The latter corresponds to a relaxed MOCDP compared to the former; however it possesses a higher objective value. This shows an optimality gap, i.e., a relative distance from the global optimum, of at least $\frac{0.046-0.041}{0.041} \approx 12\%$ for the latter problem.

The explanation for the similarity of the results in the four problems and for the frequent non-unique solutions found for a certain number of variants comes from the symmetries in the underlying MONP's. One interesting point is that it is not always advantageous to use the largest number of variants in a certain dimension. See Table 4 and compare the rows corresponding to $m = 5$ and $m = 8$. In the former, an optimal distribution of variants is $\mathbf{m} = [1, 2, 2]$ however $\mathbf{m} = [2, 3, 3]$ is not an optimal choice in the latter. Another point which might be interesting in a real application is that (however, we state no generality of this), it is possible to reach optimal variant distributions at all m-levels by local steps, adding one variant at a time, moving to the optimal distribution.

From the Tables 1–4 it seems like the solution sets are being "saturated", meaning that adding more variants do not decrease the objective value significantly. One algorithmic reason for this is that the larger m, the higher dimensional space has to be (globally) searched. Since the global search is limited by the number of function evaluations, this means that the quality of its output decreases when the dimension of the decision space is increased.

6 Extending to simulation-based MOCDP's

The purpose of this section is to is propose how to adapt the procedure developed to MOCDP for the case when MONP belongs to the class of simulation-based optimization problems.

A general simulation-based optimization problem has expensive objective (or constraint) function(s), e.g., involving computationally intense simulations. Such problems require special treatment since the total number of function evaluations is limited. A simulation-based MOCDP is a problem in which at least one of the objective functions $\{f_1, \ldots, f_k\}$ of the underlying MONP is expensive.

When solving MOCDP, a very large number of function evaluations is required in step 2 of the solution process, since for each variable vector $\mathbf{y}$, a total number of $|X_D| = \prod_{j=1}^{n} m_j$ configurations must be evaluated.

The good thing, however, is that the solution procedure is divided into two steps and that step 1 can be used not only for finding a good reference set $\mathcal{R}$. Simultaneously, it can be used for constructing explicit, computationally cheap response surfaces $\{\hat{f}_1, \ldots, \hat{f}_k\}$ (see [13, 2]) that can be used instead of the expensive simulation-based functions in step 2. The response surfaces can be continuously updated during step 1, such that, by using the response surfaces within step 1, the number of expensive function evaluations also in this step is limited. In [12] the algorithm `qualSolve` is described. This algorithm uses radial basis functions [9] with the aim of approximating the expensive functions that are sampled iteratively such that a certain quality measure is maximized. The algorithm can be applied to multi-objective optimization problems, and the quality measure is then related to how good the approximations are in regions near the Pareto optimal set of the approximating problem. `qualSolve`, or a similar algorithm, can be used in step 1, producing response surfaces to use in place of the original functions from there on.

As already stated, for sensitivity analyses, which might constitute a large part of the computation time for a real application of the MOCDP-procedure, the expensive functions must only be used once, since only step 2 is repeated when these analyses are carried through. Furthermore, even in step 1 the number of expensive function calls must not be that large, since the algorithm (e.g., `qualSolve`) mostly uses its current response surfaces and only now and then samples the original functions.

7 Conclusions and future work

We have presented a two-step procedure which can be regarded as an implicit clustering of points in the objective space of a multi-objective optimization problem such that the structure of the points in the decision space is controlled.

We have demonstrated the procedure on some test problems and discussed the potential of using different types of sensitivity analyses to perform depending on the actual application.

We have also proposed how the procedure can be adapted to simulation-based problems for which the number of (expensive) function evaluations must be kept low. The solution procedure consists of two steps and we have discussed how the first step can be used for finding computationally cheap approximate functions to use instead of the original ones in the second step. Thus, by construction of the method, the large number of function calls that have to be made in the second step is not a bigger issue for simulation-based MONP's than for regular MONP's with explicit objective functions.

The results are encouraging and we see a potential to apply the methodology to many real-world problems in industry.

There are some issues that should be addressed in order to adapt the current methodology to a larger class of problems such that it will apply to more real-world problems. One improvement would be to develop the procedure presented such that it can handle more general constraints than box constraints in MOCDP. Examples of such are general linear and non-linear constraints on the decision variables. Other examples are constraints on the objective function values in the underlying MONP. Finally, constraints in the decision space that are more connected to real configuration applications are important to govern, e.g., that combinations of certain values of the decision variables are forbidden.

In the original formulation of MOCDP, the decision variables are required to be continuous. For many real-world applications, the decision variables are required to be discrete. Also, the incorporation of the special type of discrete variables, *categorical variables* [1], that can be assigned a discrete number, but where this number has no physical meaning (e.g., representing a certain material, a certain suspension type, etcetera), would substantially increase the range of applications for the procedure presented.

References

[1] C. Audet and J. J. E. Dennis, *Pattern search algorithms for mixed variable programming*, SIAM Journal on Optimization, 11 (2001), pp. 573–594.

[2] A. J. Booker, J. E. Dennis, Jr., P. D. Frank, D. B. Serafini, V. Torczon, Trosset, and M. W., *A rigorous framework for optimization of expensive functions by surrogates*, 17, pp. 1–13.

[3] W. M. Carlyle, J. W. Fowler, E. S. Gel, and B. Kim, *Quantitative comparison of approximate solution sets for bi-criteria optimization problems*, Decision Sciences, 34 (1) (2003), pp. 63–82.

[4] P. Czyżak and A. Jaszkiewicz, *Pareto simulated annealing - a metaheuristic technique for multiple-objective combinatorial optimization*, Journal of Multi-Criteria Decision Analysis, 7 (1998), pp. 34–47.

[5] K. Deb, *Multi-objective optimization using evolutionary algorithms*, Wiley-Interscience Series in Systems and Optimization, John Wiley & Sons Ltd., Chichester, England, 2001. With a foreword by David E. Goldberg.

[6] K. Deb, S. Agrawal, A. Pratab, and T. Meyarivan, *A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: NSGA-II*, in Proceedings of the Parallel Problem Solving from Nature VI Conference, Paris, France, 2000, Springer-Verlag, pp. 849–858.

[7] D. E. Finkel, *DIRECT Optimization Algorithm User Guide*, Tech. Report, Center for Research in Scientific Computation, North Carolina State University, Raleigh, NC, USA, 2003.

[8] G. Gan, C. Ma, and J. Wu, *Data Clustering: Theory, Algorithms, and Applications (ASA-SIAM Series on Statistics and Applied Probability)*, SIAM, Philadelphia, ASA, Alexandria, VA, USA, 2007.

[9] H.-M. Gutmann, *A radial basis function method for global optimization*, Journal of Global Optimization, 19 (2001), pp. 201–227.

[10] T. Hanne, *A multiobjective evolutionary algorithm for approximating the efficient set*, European Journal of Operational Research, 176 (2007), pp. 1723–1734.

[11] 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 (2006), pp. 477–506.

[12] S. Jakobsson, M. Patriksson, J. Rudholm, and A. Wojciechowski, *A method for simulation based optimization using radial basis functions*, tech. report, Department of Mathematical Sciences, Chalmers University of Technology and Department of Mathematical Sciences, University of Gothenburg, Göteborg, Sweden, 2008.

[13] D. R. Jones, *A taxonomy of global optimization methods based on response surfaces*, Journal of Global Optimization, 21 (2001), pp. 345–383.

[14] D. R. Jones, C. D. Perttunen, and B. E. Stuckman, *Lipschitzian optimization without the Lipschitz constant*, Journal of Optimization Theory and Applications, 79 (1993), pp. 157–181.

[15] F. Kursawe, *A variant of evolution strategies for vector optimization*, in Parallel Problem Solving from Nature. 1st Workshop, PPSN I, H. P. Schwefel and R. Männer, eds., vol. 496, Berlin, Germany, 1991, Springer-Verlag, pp. 193–197.

[16] M. Pilegaard Hansen and A. Jaszkiewicz, *Evaluating the quality of approximations to the non-dominated set*, Technical report IMM-REP-1998-7, Institute of Mathematical Modelling, Technical University of Denmark, Lyngby, Denmark, 1998.

[17] M. J. D. Powell, *The NEWUOA software for unconstrained optimization without derivatives*, in Large-scale Nonlinear Optimization, P. Pardalos, G. Pillo, and M. Roma, eds., vol. 83 of Nonconvex Optimization and its Application, Springer-Verlag, New York, NY, USA, 2006, pp. 255–297.

[18] S. Sayin, *Measuring the quality of discrete representations of efficient sets in multiple objective mathematical programming*, Mathematical Programming, 87 (2000), pp. 543–560.

[19] A. Viana and J. Pinho de Sousa, *Using metaheuristics in multiobjective resource constrained project scheduling*, European Journal of Operational Research, 120 (2000), pp. 359–374.

[20] S. Watanabe, T. Hiroyasu, and M. Miki, *NCGA: Neighborhood cultivation genetic algorithm for multi-objective optimization*, in Proceedings of the Genetic and Evolutionary Computation Conference, E. Cantú-Paz, ed., New York, NY, USA, 2002, pp. 458–465.

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