On Objective Conflicts and Objective Reduction in Multiple Criteria Optimization

Dimo Brockhoff and Eckart Zitzler
{brockhoff,zitzler}@tik.ee.ethz.ch

TIK-Report No. 243

Institut für Technische Informatik und Kommunikationsnetze
ETH Zürich
Gloriastrasse 35, ETH-Zentrum, CH–8092 Zürich, Switzerland

February 2006

Abstract. A common approach in multiobjective optimization is to perform the decision making process after the search process: first, a search heuristic approximates the set of Pareto-optimal solutions, and then the decision maker chooses an appropriate trade-off solution from the resulting approximation set. Both processes are strongly affected by the number of optimization criteria. The more objectives are involved the more complex is the optimization problem and the choice for the decision maker. In this context, the question arises whether all objectives are actually necessary and whether some of the objectives may be omitted; this question in turn is closely linked to the fundamental issue of conflicting and non-conflicting optimization criteria. Besides a general definition of conflicts between objective sets, we here introduce the problem of computing a minimum subset of objectives without losing information (MOSS) and show that this is an $\mathcal{NP}$-hard problem. Furthermore, we present for MOSS both an approximation algorithm with optimum approximation ratio and an exact algorithm which works well for small input instances. The paper concludes with experimental results for random sets and the multiobjective 0/1-knapsack problem.

1 Motivation

With the availability of sufficient computing resources, generating methods for identifying or approximating the set of Pareto-optimal solutions have become increasingly popular for tackling multiobjective optimization problems. The advantage of these methods is that the decision making process is postponed after the optimization process: the decision maker can choose an appropriate trade-off solution from a set of alternative solutions generated by the corresponding search algorithm. However, the complexity of both processes is strongly affected by the number of objectives involved. On the one hand, the running time of generating methods may be exponential in the number of objectives as, e. g., for algorithms based on the hypervolume indicator [14, 5, 10], and on the other hand comparing even only a few alternative solutions may become difficult or infeasible for a human decision maker, if too many objectives are considered simultaneously. In the light of this discussion, the question arises whether it is possible to omit some of the objectives without changing the characteristics of the underlying problem. Furthermore, one may ask under which conditions such an objective reduction is feasible and how a minimum set of objectives can be computed.

These questions have gained only little attention in the literature so far. There are closely related research topics such as principal component analysis [4] and dimension theory [9], which have a different focus though. Transferred to the multiobjective optimization setting, the corresponding methods aim at determining a (minimum) set of *arbitrary* objective functions that preserves (most of) the problem characteristics; however, here we are interested in determining a minimum subset of *original* objectives that maintains the order on the search space. Furthermore, there a few studies that investigate the relationships between objectives in terms of conflicting and nonconflicting optimization criteria. Deb [2] defines a set of objectives as conflicting, if there exists one solution that simultaneously achieves for each criterion the optimal value; otherwise the set is nonconflicting. Tan, Khor, and Lee [8] presented a refinement of this definition where a conflict denotes the existence of incomparable[1] solutions in the search space. A similar notion of conflict has been suggested by Purshouse and Fleming [6] who consider conflict as a binary relation between single objectives. However, these definitions are not sufficient to indicate whether objectives can be omitted or not as the following example demonstrates; although all objectives are conflicting according to [2, 6, 8], one of the three objectives can be removed while preserving the search space order.

Example 1 *Fig. 1 shows the parallel coordinates plot, cf. [6], of three solutions* $\mathbf{x}_1$ *(solid line),* $\mathbf{x}_2$ *(dotted) and* $\mathbf{x}_3$ *(dashed) that are pairwise incomparable.*

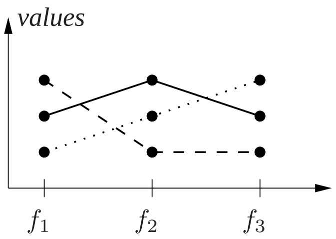


Fig. 1. Parallel coordinates plot for three solutions and three objectives f_1, f_2, f_3.

Assuming that $\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3$ *represent the entire search space, the original objective set* $\{f_1, f_2, f_3\}$ *is conflicting according to [2, 8] and all objective pairs "exhibit evidence of conflict" as defined in [6]. Nevertheless, the objective set* $\{f_1, f_2, f_3\}$ *contains redundant information: the objective* f_2 *can be omitted, and all solutions remain incomparable to each other with regard to the objective set* $\{f_1, f_3\}$.

This paper addresses two open issues: (i) deriving general conditions under which certain objectives may omitted and (ii) computing a minimum subset of objectives needed to preserve the problem structure. In particular, we

- propose a generalized notion of objective conflicts which comprises the definitions of Deb [2], Tan et al. [8], and Purshouse and Fleming [6],
- specify on this basis a necessary and sufficient condition under which objectives can be omitted,
- introduce the problem of minimum objective subsets (`MOSS`),
- show that `MOSS` is $\mathcal{NP}$-hard,
- provide an approximation algorithm with optimum approximation ratio as well as an exact algorithm which has polynomial runtime in the decision space size, and

[1] Two solutions are incomparable iff either is better than the other one in some objectives.

– validate our approach on both random problems and the 0/1-knapsack problem by comparing the algorithms and investigating the influence of the number of objectives and the search space size.

In addition, extensions of the proposed approach will be discussed in the last section.

2 A Notion of Objective Conflicts

2.1 The Relation Between Objectives and Orders

A general optimization problem can be considered as a quadruple (X, Z, f, rel) where X denotes the search space or decision space, Z represents the objective space, $f : X \to Z$ is a function that assigns to each solution or decision vector $\mathbf{x} \in X$ a corresponding objective vector $\mathbf{z} = f(\mathbf{x}) \in Z$, and $rel \subseteq Z \times Z$ represents a partial order[2] over Z. The goal is to find a solution $\mathbf{x}^* \in X$ that is mapped to a minimal element[3] $\mathbf{z}^* = f(\mathbf{x}^*)$ of $f(X) := \{\mathbf{z} \in Z | \exists \mathbf{x} \in X : \mathbf{z} = f(\mathbf{x})\}$ regarding the partially ordered set (Z, rel).

In the scenario considered in this paper, f consists of one or several objective functions $f_1, f_2, \ldots, f_k$ that are all to be minimized where $f = (f_1, \ldots, f_k)$, $f_i : X \to \mathbb{R}$ for $1 \leq i \leq k$, and $Z = \mathbb{R}^k$. Furthermore, we assume that rel is the relation $\leq$ on real vectors with $\mathbf{z} \leq \mathbf{z}' :\Longleftrightarrow \forall 1 \leq i \leq k : z_i \leq z'_i$ which induces a corresponding preorder $\preceq$ on X with $\mathbf{x}_1 \preceq \mathbf{x}_2 :\Longleftrightarrow f(\mathbf{x}_1) \leq f(\mathbf{x}_2)$. The relation $\preceq$ is also known as weak Pareto dominance, and we say $\mathbf{x}_1$ weakly dominates $\mathbf{x}_2$ whenever $\mathbf{x}_1 \preceq \mathbf{x}_2$; other dominance relations such as epsilon dominance, cf. [14], could be taken as well, and the following discussions applies to any preorder on X that is defined by a corresponding partial order on $\mathbb{R}^k$. The minimal elements of $f(X)$ with respect to $(\mathbb{R}^k, \leq)$ form the so-called Pareto front, and solutions that are mapped to elements of the Pareto front are denoted as Pareto-optimal and constitute the Pareto set. If there exist two incomparable Pareto-optimal solutions $\mathbf{x}_1, \mathbf{x}_2$, i. e., neither weakly dominates the other one ($\mathbf{x}_1 || \mathbf{x}_2$), then the cardinality of the Pareto front is greater than 1. If two solutions $\mathbf{x}_1, \mathbf{x}_2$ are indifferent, i. e., they are mapped to the same objective vector ($\mathbf{x}_1 \sim \mathbf{x}_2$), then the relation $\preceq$ is only a preorder, but not a partial order on X. However, we can define a partial order $\precsim$ on the set $X/\!\sim$ of equivalence classes regarding $(X, \sim)$ as follows:

$$\forall [p], [q] \in X/\!\sim: [p] \precsim [q] :\Longleftrightarrow p \preceq q \wedge p \not\sim q.$$

The remainder of this paper addresses the issue of finding a minimum subset of the objectives that induces the same preorder on the decision space as the complete set of objectives. To this end, we here introduce a generalization of the weak Pareto dominance relation defined above: a decision vector $\mathbf{x}_1 \in X$ weakly dominates a decision vector $\mathbf{x}_2 \in X$ w. r. t. the set $\mathcal{F} \subseteq \{f_1, f_2, \ldots, f_k\}$ of objective functions (written as

[2] A relation rel is called a preorder iff it is reflexive and transitive; a preorder that is antisymmetric is denoted as partial order. We call a partial order total order or linear order if it is total; a preorder that is total is called total preorder.

[3] Given a partial ordered set (Z, rel), an element $\mathbf{z}^* \in Z'$ with $Z' \subseteq Z$ is called minimal element of Z' iff for all $\mathbf{z} \in Z'$ holds: $\mathbf{z}\, rel\, \mathbf{z}^* \Rightarrow \mathbf{z} = \mathbf{z}^*$.

$\mathbf{x}_1 \preceq_{\mathcal{F}} \mathbf{x}_2$) iff $\forall f \in \mathcal{F} : f(\mathbf{x}_1) \leq f(\mathbf{x}_2)$. We will write $\preceq_i$ if we mean the weak dominance relation w. r. t. $\mathcal{F} = \{f_i\}$; in addition, we define $\preceq_\emptyset := X \times X$ for the case that $\mathcal{F}$ is empty. The following theorem shows that for any objective function set the generalized weak Pareto dominance relation can be derived from the objective-wise less than or equal relation on $\mathbb{R}$.

Theorem 1. *Let $\mathcal{F} = \{f_1, \ldots, f_k\}$ be a set of k different objective functions. Then it holds:*

$$\preceq_{\mathcal{F}} = \bigcap_{1 \leq i \leq k} \preceq_i$$

Proof: For all $x, y \in X$:

$$\begin{aligned}(x,y) \in \preceq_{\mathcal{F}} &\iff x \preceq y \text{ w. r. t. } \mathcal{F} \iff \forall i \in \{1, \ldots, k\} : f_i(x) \leq f_i(y) \\ &\iff \forall i \in \{1, \ldots, k\} : x \preceq y \text{ w. r. t. } f_i \\ &\iff \forall i \in \{1, \ldots, k\} : (x,y) \in \preceq_i\end{aligned}$$

Note that the above equivalence also holds for the strict dominance relation and the multiplicative ε-dominance relation, cf. [14], but does not apply to the regular Pareto dominance relation $\prec$ defined as $\mathbf{x}_1 \prec \mathbf{x}_2 :\iff \mathbf{x}_1 \preceq \mathbf{x}_2 \wedge \neg(\mathbf{x}_2 \preceq \mathbf{x}_1)$. □

Finally, we will use a graphical notation for relations, called relation graphs. Given a certain ordered set (Z, rel), the relation graph for (Z, rel) has a vertex per element in Z and a directed edge between the vertices $\mathbf{z}$ and $\mathbf{z}'$ only if $\mathbf{z} \, rel \, \mathbf{z}'$. For a partial ordered set, the relation graph can be reduced to a Hasse diagram, with an edge between vertices $\mathbf{z}$ and $\mathbf{z}'$ iff $\mathbf{z}$ is a lower cover[4] of $\mathbf{z}'$. The relation graph is only another description of a relation but helps us to visualize our ideas.

Example 2 *Let $(X := \{A, B, C, D, E\}, \mathbb{R}^2, f = (f_1, f_2), \leq)$ be a multiobjective optimization problem where f is specified by the objective values in the following table. Fig. 2 shows the relation graph of $(X, \preceq_{\{f_1, f_2\}})$ and the relation graph and Hasse diagram for $(f(X), \leq)$.*

$\mathbf{x}$	A	B	C	D	E
$f_1(\mathbf{x})$	*1*	*2*	*3*	*4*	*4*
$f_2(\mathbf{x})$	*2*	*1*	*4*	*5*	*5*

The solutions A and B are the minimal elements of $(X, \preceq_{\{f_1, f_2\}})$, i. e., the Pareto set, whereas $f(A)$ and $f(B)$ form the Pareto front, i. e., they are the minimal elements of $f(X)$ with respect to $(\mathbb{R}^2, \leq)$. A and B are the only incomparable and D and E the only indifferent decision vectors according to the relation $\preceq_{\{f_1, f_2\}}$.

2.2 Partial Orders on Sets of Objectives

In this section, we introduce a general concept of conflicts between sets of objectives. On the basis of the following definitions, two algorithms to exactly resp. approximately compute a minimum set of objectives, which induces the same preorder on X as the whole set of objectives, will be proposed in Sec. 3.

[4] We say $\mathbf{z}$ is a lower cover of $\mathbf{z}'$ iff $\forall \mathbf{z}^* \in Z : \mathbf{z} \, rel \, \mathbf{z}^* \wedge \mathbf{z}^* \, rel \, \mathbf{z}' \Rightarrow \mathbf{z}^* = \mathbf{z} \vee \mathbf{z}^* = \mathbf{z}'$.

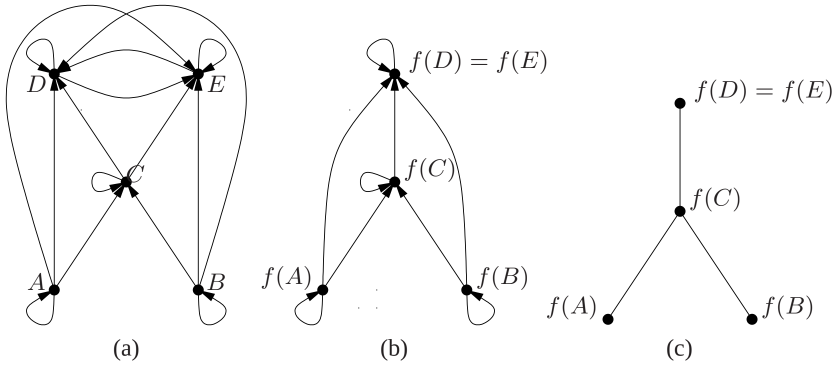


Fig. 2. (a) Relation graph of $(X, \preceq_{\{f_1, f_2\}})$, (b) relation graph of $(f(X), \leq)$, and (c) Hasse diagram of $(f(X), \leq)$ from Example 2.

Definition 1 *Let $\mathcal{F}_1, \mathcal{F}_2 \subseteq \mathcal{F}$ be two sets of objectives. Then $\mathcal{F}_1 \sqsubseteq \mathcal{F}_2 \;:\Leftrightarrow \preceq_{\mathcal{F}_1} \subseteq \preceq_{\mathcal{F}_2}$.*

Definition 2 *Let $\mathcal{F}_1, \mathcal{F}_2 \subseteq \mathcal{F}$ be two sets of objectives. We call*

- $\mathcal{F}_1$ nonconflicting *with $\mathcal{F}_2$ iff $\mathcal{F}_1 \sqsubseteq \mathcal{F}_2 \wedge \mathcal{F}_2 \sqsubseteq \mathcal{F}_1$*
- $\mathcal{F}_1$ weakly conflicting *with $\mathcal{F}_2$ iff $(\mathcal{F}_1 \sqsubseteq \mathcal{F}_2 \wedge \mathcal{F}_2 \not\sqsubseteq \mathcal{F}_1) \vee (\mathcal{F}_2 \sqsubseteq \mathcal{F}_1 \wedge \mathcal{F}_1 \not\sqsubseteq \mathcal{F}_2)$*
- $\mathcal{F}_1$ strongly conflicting *with $\mathcal{F}_2$ iff $\mathcal{F}_1 \not\sqsubseteq \mathcal{F}_2 \wedge \mathcal{F}_2 \not\sqsubseteq \mathcal{F}_1$*

By definition, $\sqsubseteq$ is a preorder since $\subseteq$ is a preorder. Two sets of objectives $\mathcal{F}_1, \mathcal{F}_2$ are called nonconflicting if and only if the corresponding relations $\preceq_{\mathcal{F}_1}$ and $\preceq_{\mathcal{F}_2}$ are identical but not necessarily $\mathcal{F}_1 = \mathcal{F}_2$; in other words, $\mathcal{F}_1$ and $\mathcal{F}_2$ are indifferent w. r. t. $\sqsubseteq$. If $\mathcal{F}_1 \subset \mathcal{F}_2$ and $\mathcal{F}_1$ is nonconflicting with $\mathcal{F}_2$ we can simply omit all objectives in $\mathcal{F}_2 \setminus \mathcal{F}_1$ without influencing the preorder on X. Furthermore, the term "strongly conflicting" corresponds to incomparability w. r. t. $\sqsubseteq$, while "weakly conflicting" means neither indifferent nor incomparable w. r. t. $\sqsubseteq$. These two types of conflicts are mutually exclusive which is useful in the context of the following result.

Theorem 2. *Let $\mathcal{F}$ be a set of objectives. Then $\sqsubseteq$ is a total preorder on $\mathcal{P}(\mathcal{F})$ if and only if there are no strongly conflicting pairs $\mathcal{F}_1, \mathcal{F}_2 \in \mathcal{P}(\mathcal{F})$.*

Proof: By definition, it is clear that $\sqsubseteq$ is always reflexive and transitive. Assume that there are no strongly conflicting pairs $\mathcal{F}_1, \mathcal{F}_2 \in \mathcal{P}(\mathcal{F})$, i. e.

$$\begin{aligned} &\nexists \mathcal{F}_1, \mathcal{F}_2 \in \mathcal{P}(\mathcal{F}) : \mathcal{F}_1 \not\sqsubseteq \mathcal{F}_2 \wedge \mathcal{F}_2 \not\sqsubseteq \mathcal{F}_1 \\ &\quad \Longleftrightarrow \forall \mathcal{F}_1, \mathcal{F}_2 \in \mathcal{P}(\mathcal{F}) : \mathcal{F}_1 \sqsubseteq \mathcal{F}_2 \vee \mathcal{F}_2 \sqsubseteq \mathcal{F}_1 \Longleftrightarrow \quad \sqsubseteq \text{ is total} \end{aligned}$$

Thus, $\sqsubseteq$ is total iff there are no strongly conflicting pairs of objective sets. □

Note that the above formulation of conflicting objectives can be regarded as a generalization of Purshouse and Fleming's definition [6] which only considers pairs of objectives; moreover, it also comprises the notions by Deb [2] and Tan et al. [8]. For a more detailed discussion of the connection to previous definitions of objective conflicts, we refer to the appendix.

2.3 Minimal, Minimum, and Redundant Objective Sets

Based on the above conflict relations, we will now formalize the notion of redundant objective sets.

Definition 3 *Let $\mathcal{F}$ be a set of objectives. An objective set $\mathcal{F}' \subseteq \mathcal{F}$ is denoted as*

- minimal w. r. t. *$\mathcal{F}$ iff (i) $\mathcal{F}'$ is nonconflicting with $\mathcal{F}$, and (ii) there exists no $\mathcal{F}'' \subset \mathcal{F}'$ that is nonconflicting with $\mathcal{F}$;*
- minimum w. r. t. *$\mathcal{F}$ iff (i) $\mathcal{F}'$ is minimal w. r. t. $\mathcal{F}$, and (ii) there exists no $\mathcal{F}'' \subset \mathcal{F}$ with $|\mathcal{F}''| < |\mathcal{F}'|$ that is minimal w. r. t. $\mathcal{F}$.*

A minimal objective set is a subset of the original objectives that cannot be further reduced without changing the associated preorder. A minimum objective set is the smallest possible set of original objectives that preserves the original order on the search space. By definition, every minimum objective set is minimal, but not all minimal sets are at the same time minimum.

Definition 4 *A set $\mathcal{F}$ of objectives is called* redundant *if and only if there exists $\mathcal{F}' \subset \mathcal{F}$ that is minimal w. r. t. $\mathcal{F}$.*

This definition of redundancy represents a necessary and sufficient condition for the omission of objectives.

3 The Minimum Objective Subset Problem

Given a multiobjective optimization problem with the set $\mathcal{F}$ of objectives, the question arises whether objectives can be omitted without changing the order on the search space. If an objective subset $\mathcal{F}' \subseteq \mathcal{F}$ can be computed and $\mathbf{x} \preceq_{\mathcal{F}'} \mathbf{y}$ holds for all solutions $\mathbf{x}, \mathbf{y} \in X$ if and only if $\mathbf{x} \preceq_{\mathcal{F}} \mathbf{y}$, we can omit all objectives in $\mathcal{F} \setminus \mathcal{F}'$ while preserving the preorder on X. Concerning the last section, we are interested in identifying a minimum objective subset with respect to $\mathcal{F}$, yielding a slighter representation of the same multiobjective optimization problem. Formally, this problem can be stated as follows.

Definition 5 *The search problem* `MINIMUM OBJECTIVE SUBSET` *(MOSS) is defined as follows.*

Given: *A multiobjective optimization problem $(X, Z, \mathcal{F} = \{f_1, \ldots, f_k\}, \leq)$*

Instance: *The set X of solutions, the generalized weak Pareto dominance relation $\preceq_{\mathcal{F}}$ and for all objective functions $f_i \in \mathcal{F}$ the single relations $\preceq_i$ where $\bigcap_{1 \leq i \leq k} \preceq_i = \preceq_{\mathcal{F}}$.*

Task: *Compute an index $I \subseteq \{1, \ldots, k\}$ of minimum size with $\bigcap_{i \in I} \preceq_i = \preceq_{\mathcal{F}}$.*

Note that the limitation of the instances to the whole search space description is not essential here. One can think of situations where the underlying set is the Pareto set or an approximation of it. The restriction to the partial order $\leq$ and its corresponding preorder $\preceq_{\mathcal{F}}$ is not essential as well, but instead of any partially ordered set (Z, rel) we consider only $(\mathbb{R}^n, \leq)$ here. Note that we are not interested in a minimal objective subset but in a minimum objective set w. r. t. the set of all objectives. The approach of finding a minimum objective subset is related to dimension theory [9]. Given a partial order rel, the dimension of rel is defined as the minimum number of linear extensions[5] of rel, the intersection of which is rel. A set of linear extensions the intersection of which is rel is called a realizer for rel. The main difference between the computation of the dimension of a partial order and our approach of finding the size of a minimum objective subset w. r. t. the set of all objectives is the fact, that the corresponding realizer contains linear extensions which do not bear relation to the relations $\preceq_i$. Instead in a realizer for the partial order $\precsim$, we are interested in a set of given relations $\preceq_i$ the intersection of which is $\preceq_{\mathcal{F}}$. For simplification, let us assume that there are no indifferent solutions, i. e., $\preceq_{\mathcal{F}}$ is a partial order. The dimension of $\preceq_{\mathcal{F}}$ gives us only a lower bound for the size of a minimum subset of objectives w. r. t. $\mathcal{F}$. For example, the dimension of $\preceq_{\mathcal{F}}$ is always 2 if all decision vectors are incomparable, but the size of the minimum objective set can be greater than 2. Instead of the computation of a minimum realizer in dimension theory, which is $\mathcal{NP}$-hard [11], we are interested in a shorter description of our problem with a selection of the given objectives, the complexity of which will emerge as $\mathcal{NP}$-hard, too, in the next section.

3.1 Proof of $\mathcal{NP}$-hardness

That `MOSS` is a set problem does not directly arise from the definition of the `MOSS` problem but, obviously, the relations $\preceq_i$ in Def. 5 as well as $\preceq_{\mathcal{F}}$ are subsets of $X \times X$. Considering the complementary sets $\preceq^C_{\mathcal{F}'} := (X \times X) \setminus \preceq_{\mathcal{F}'}$ for any $\mathcal{F}' \subseteq \mathcal{F}$ and De Morgan's laws, the task of the `MOSS` problem can be restated as finding a minimum index I such that $\bigcup_{i \in I} \preceq^C_i = \preceq^C_{\mathcal{F}}$. Hence, the $\mathcal{NP}$-hard problem `SET COVER` introduced in [5] is closely related to the `MOSS` problem.

Definition 6 *We define the search problem `SET COVER`, or `SCP` for short, as follows.*

Instance: *A Collection $C = \{C_1, \ldots, C_k\}$ of subsets of a finite set $S = \{1, \ldots, m\}$.*

Task: *Compute an index $I \subseteq \{1, \ldots, k\}$ of minimum size with $\bigcup_{i \in I} C_i = S$.*

The set S in an `SCP` instance complies with the relation $\preceq^C_{\mathcal{F}}$ in a `MOSS` instance just as each subset C_i corresponds to the relation $\preceq^C_i$. Just as the C_i's are subsets of S, the $\preceq_i$'s are supersets of $\preceq_{\mathcal{F}}$, i. e., the complementary relations $\preceq^C_i$ are subsets of $\preceq^C_{\mathcal{F}}$. Nevertheless, `SCP` and `MOSS` are not identical problems due to the fact that the allowed instances for `MOSS` have to ensure that the relations correspond to preorders on X whereas for `SCP`, instances with arbitrary sets are allowed. More precisely, the relations $\preceq_i$ in an allowed `MOSS` instance are always linear orders, written as $[\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n]$ with $\mathbf{x}_i \in X$, augmented with additional relations between indifferent solution pairs, thus, the relations $\preceq_i$ are preorders, cf. Fig. 3 for an example. Because of the similarity between `SCP` and `MOSS` it is not surprising that also `MOSS` is $\mathcal{NP}$-hard. In the following we use a Turing reduction `SCP` $\leq_T$ `MOSS` to prove the $\mathcal{NP}$-hardness of `MOSS`.

[5] A linear extension of a relation $rel \subseteq Z \times Z$ is a linear order on $Z \times Z$, containing rel.

Theorem 3. *The problem* `MOSS` *is* $\mathcal{NP}$*-hard.*

Sketch of Proof: To simplify the notations below, we denote the input size of `MOSS` by n, where $n = \Theta(k \cdot m^2)$, k denotes the number of objectives, and $m := |X|$. For the $\mathcal{NP}$-hardness proof, a Turing reduction `SCP` $\leq_T$ `MOSS` is required. Due to space limitations, we only provide a sketch of the transformation and refer for the correctness proof of this transformation to the appendix. For a small instance, Fig. 3 visualizes the basic idea of the transformation.

Starting from an `SCP` instance, consisting of the set $S = \{s_1, \dots, s_m\}$ and the subsets C_i with $1 \leq i \leq k$, all relations $\preceq_i$ as well as $\preceq_\mathcal{F}$ in the `MOSS` instance are defined as subsets of $X \times X$ with $X := \{\mathbf{x}_1, \dots, \mathbf{x}_m, \mathbf{x}'_1, \dots, \mathbf{x}'_m\}$. According to the similarity of the two problems, each set in the `SCP` instance has its counterpart in the generated `MOSS` instance. The relation $\preceq_\mathcal{F}$ corresponds to the set S and is the reflexive closure of the antichain[6] on X, i.e., $\preceq_\mathcal{F}$ only contains the elements $(\mathbf{x}_j, \mathbf{x}_j)$ and $(\mathbf{x}'_j, \mathbf{x}'_j)$ for $1 \leq j \leq m$. For each subset C_i of S with $1 \leq i \leq k$ we create the relation $\preceq_i$ in the `MOSS` instance. The relation $\preceq_i$ includes the linear order $[\mathbf{x}_1, \mathbf{x}'_1, \mathbf{x}_2, \mathbf{x}'_2, \dots, \mathbf{x}_m, \mathbf{x}'_m]$ and additionally, the relation $\preceq_i$ contains the element $(\mathbf{x}'_j, \mathbf{x}_j)$ iff $s_j \notin C_i$. In addition to the k relations $\preceq_i$, we compute the relation $\preceq_{k+1}$ which is the reverse linear order $[\mathbf{x}'_m, \mathbf{x}_m, \mathbf{x}'_{m-1}, \mathbf{x}_{m-1}, \dots, \mathbf{x}'_1, \mathbf{x}_1]$. After this transformation, we question our `MOSS` oracle once. The resulting index $I_{\texttt{SCP}}$ for the `SCP` problem will be then $I_{\texttt{SCP}} := I_{\text{oracle}} \setminus \{k+1\}$ if the oracle produces I_{oracle} as its output. The whole transformation takes time $O(km^2)$ and produces an `MOSS` instance of size $O(km^2)$. □

3.2 An Approximation Algorithm

As the computation of a minimum objective subset of objectives is $\mathcal{NP}$-hard, we cannot expect to find an exact deterministic algorithm for the problem with polynomial running time, unless $\mathcal{P} = \mathcal{NP}$. Instead, we present an approximation algorithm with polynomial running time in the following; an exact algorithm will be proposed in Sec. 3.3. With Algorithm 1, we propose a greedy strategy for the `MOSS` problem. For `SCP`, an approximation algorithm with a similar greedy strategy is already known the approximation ratio of which is $\ln m - \ln \ln m + \Theta(1)$ where m is the number of elements in the set S [7]. This knowledge is useful for proving the following result on Algorithm 1.

Theorem 4. *Algorithm 1 is an approximation algorithm for the* `MOSS` *problem with approximation ratio* $\Theta(\log m)$ *and needs time* $O(k \cdot m^2) = O(n)$.

Proof: First, we show that Algorithm 1 always computes a correct solution for the `MOSS` problem, i.e., an index I with $\bigcap_{i \in I} \preceq_i = \preceq_\mathcal{F}$. By construction, Algorithm 1 provides always an index I with $\bigcup_{i \in I} \preceq_i^C \supseteq \preceq_\mathcal{F}^C$, i.e., $\bigcap_{i \in I} \preceq_i \subseteq \preceq_\mathcal{F}$. As $\bigcap_{1 \leq i \leq k} \preceq_i = \preceq_\mathcal{F}$, and thus $\bigcap_{1 \leq i \leq k} \preceq_i \supseteq \preceq_\mathcal{F}$ holds, the equivalence $\bigcap_{i \in I} \preceq_i = \preceq_\mathcal{F}$ is always true.

To show the upper bound on the approximation ratio, we sketch the proof of a Turing reduction `MOSS` $\leq_T$ `SCP` and refer to the appendix for the correctness proof. Given

[6] The reflexive closure of an antichain is simply a relation with only reflexive edges in their graph representation.

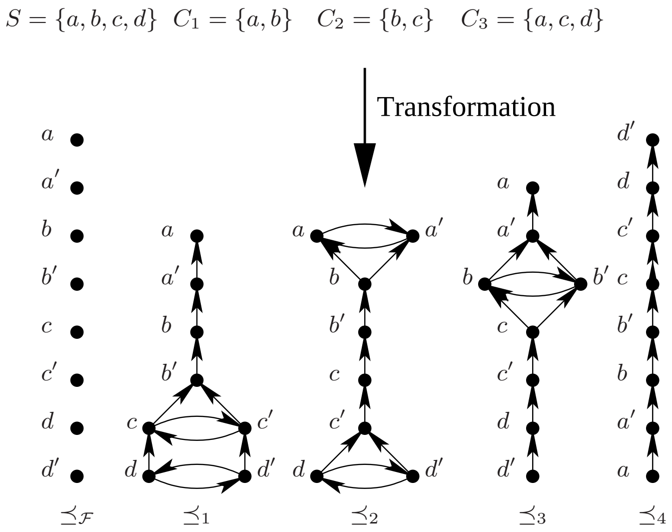


Fig. 3. An example for the Turing reduction from SCP to MOSS. The reflexive and transitive edges are omitted for clarity.

an instance for MOSS, consisting of the relations $\preceq_{\mathcal{F}} \subseteq X \times X$ and $\preceq_i \subseteq X \times X$ with $\bigcap_{1 \leq i \leq k} \preceq_i = \preceq_{\mathcal{F}}$, we can compute an SCP instance as follows. The set S in the SCP instance contains an element $s_{x,y}$ for each $(\mathbf{x}, \mathbf{y}) \in \preceq_{\mathcal{F}}$. A subset C_i of S in the SCP instance contains an element $s_{x,y}$ iff $\mathbf{x} \not\preceq_i \mathbf{y}$. The output for the MOSS problem, is the index I, computed by the SCP oracle. The Turing reduction needs time $O(n)$ and produces an SCP instance of size $O(n)$. Since Algorithm 1 uses this transformation and then acts like the greedy algorithm for SCP, the upper bound $O(\log m)$ for the approximation ratio of the greedy algorithm for SCP is directly translated to Algorithm 1.

For proving that Algorithm 1 has an approximation ratio of $\Omega(\log m)$, we use conclusions made for SCP. Feige showed in [3], that there is no $\varepsilon > 0$ such that an approximation algorithm can solve SCP with approximation ratio $(1 - \varepsilon) \ln m$, unless $\mathcal{NP} \subset TIME(m^{O(\log \log m)})$. With our transformation from SCP to MOSS, Feige's lower bound for SCP yields to a lower bound of $\Omega(\log 2m) = \Omega(\log m)$ for MOSS. This is due to the fact that in the transformation from SCP to MOSS the size m of the set S is transformed into the set X of size $2m$. Assuming, that there is a polynomial approximation algorithm for MOSS with an approximation ratio of $o(\log m)$, we get a contradiction to Feige's results, because we can transform each SCP instance in polynomial time into a MOSS instance with X of size $2m$ and solve SCP via the $o(\log m)$ algorithm for MOSS.

The worst-case running time of Algorithm 1 is $O(k \cdot m^2) = O(n)$: The computation of the complementary relations during initialization needs time $O(k \cdot m^2)$ and the total runtime—amortized over all $O(m^2)$ loop cycles—is $O(k \cdot m^2)$ for the update of the $\preceq_i^C$'s, and $\preceq_i^C \cap E$ respectively, together with the computation of E. Furthermore, each

Algorithm 1 A greedy algorithm for MOSS

Init:
  $E := \preceq_{\mathcal{F}}^{C}$ where $\preceq_{\mathcal{F}}^{C} := \ (X \times X) \setminus \preceq_{\mathcal{F}}$
  $I := \emptyset$
while $E \neq \emptyset$ **do**
  choose an $i \in (\{1, \ldots, k\} \setminus I)$ such that $| \preceq_{i}^{C} \cap E|$ is maximal
  $E := E \setminus \preceq_{i}^{C}$
  $I := I \cup \{i\}$
end while

of the $O(m^2)$ steps of the while loop costs additionally time $O(k)$ for the calculation of the maximum and the update of I. □

3.3 An Exact Algorithm

In this section, we present an exact algorithm for the MOSS problem, the running time of which is polynomial in the size of X but exponential in the number of objectives. In order to solve the MOSS problem exactly it is in general not sufficient to take information about conflicts between pairs of objectives into account. Example 1 shows a simple instance with three objectives. Even though all pairs of objective functions are strongly conflicting according to Def. 2, the whole set of objectives is redundant, i. e., f_2 can be omitted. Almost the same situation emerges, if we want to solve the MOSS problem with the help of information about conflicts between pairs of sets with larger but constant size. The observation that there is no possibility for a correct predication whether a set of objectives is redundant, by observing only relations between objective subsets of constant size, can be likewise derived from the $\mathcal{NP}$-hardness of the MOSS problem. Thus, we are forced to examine the type of conflict between all possible objective subsets if we want to solve the MOSS problem exactly.

Algorithm 2 examines all possible objective subset pairs $\mathcal{F}_1, \mathcal{F}_2 \in \mathcal{P}(\mathcal{F})$[7] in combination with all solution pairs $\mathbf{x}, \mathbf{y} \in X$ separately by calculating the set S_{xy} of all minimal objective subsets w. r. t. $\mathcal{F}$ explaining the relation between $\mathbf{x}$ and $\mathbf{y}$ w. r. t. $\preceq_{\mathcal{F}}$. The set S of objective subsets always contains all minimal subsets as solutions for the MOSS problem restricted to the solution pairs considered so far. S is updated whenever a new solution pair is observed. To simplify the notation, we use the symbol $\sqcup$ for a union of two sets $S_1, S_2 \subseteq \mathcal{P}(\mathcal{F})$ containing themselves objective subsets. $S_1 \sqcup S_2$ contains the pairwise union $s_1 \cup s_2$ of sets $s_1 \in S_1$ and $s_2 \in S_2$ only if there is no subset of $s_1 \cup s_2$ in $S_1 \cup S_2$:

$$S_1 \sqcup S_2 := \{s_1 \cup s_2 \,|\, s_1 \in S_1 \wedge s_2 \in S_2 \wedge (\nexists p_1 \in S_1, p_2 \in S_2 : p_1 \cup p_2 \subset s_1 \cup s_2)\}$$

When all solution pairs are processed, S contains all minimal objective subsets w. r. t. $\mathcal{F}$ from which Algorithm 2 chooses a minimum one as an exact solution for the MOSS problem.

[7] With $\mathcal{P}(\mathcal{F})$ we denote the power set of $\mathcal{F} := \{f_1, \ldots, f_k\}$.

Algorithm 2 An exact algorithm for MOSS

Init:
  $S := \emptyset$
for each pair $\mathbf{x}, \mathbf{y} \in X$ of solutions **do**
  $S_x := \{\, \{i\} \,|\, i \in \{1, \ldots, k\} \wedge \mathbf{x} \preceq_i \mathbf{y} \wedge \mathbf{y} \not\preceq_i \mathbf{x}\}$
  $S_y := \{\, \{i\} \,|\, i \in \{1, \ldots, k\} \wedge \mathbf{y} \preceq_i \mathbf{x} \wedge \mathbf{x} \not\preceq_i \mathbf{y}\}$
  $S_{xy} := S_x \sqcup S_y$
  if $S_{xy} = \emptyset$ **then** $S_{xy} := \{1, \ldots, k\}$
  $S := S \sqcup S_{xy}$
end for
Output:
  smallest set $s_{\min}$ in S

Theorem 5. *Algorithm 2 solves the MOSS problem exactly in time* $O(m^2 \cdot k \cdot 2^k)$.

Proof: For a correctness proof, we have to ensure that Algorithm 2 computes the sets in S_{xy} correctly. Then, the invariant, that S contains all minimal sets of objectives which explain the relationships between all considered pairs of solutions, is always correct. The sets are always minimal, because we delete all supersets during the $S := S \sqcup S_{xy}$ command. For the first pair $\mathbf{x}, \mathbf{y}$ of solutions, $S = S_{xy}$ is computed correctly and the invariant holds as a result of induction. We now distinguish between the three possible relationships between solution pairs and show for each type that our algorithm computes S_{xy} correctly. (i) In the case of an indifferent solution pair $\mathbf{x} \sim \mathbf{y}$, i.,e., $\forall f_i \in \mathcal{F} : f_i(\mathbf{x}) = f_i(\mathbf{y})$, both S_x and S_y are empty sets, yielding to $S_{xy} = \{1, \ldots, k\}$. Because indifferent vectors $\mathbf{x}, \mathbf{y}$ have the same objective vector, each single objective f_i is a possible minimal set which explain the indifference. (ii) If we consider comparable solutions, without loss of generality $\mathbf{x} \preceq \mathbf{y} \wedge \neg(\mathbf{x} \sim \mathbf{y})$, i. e., $\forall f \in \mathcal{F} : f(\mathbf{x}) \leq f(\mathbf{y}) \wedge \exists f' \in \mathcal{F} : f'(\mathbf{x}) < f'(\mathbf{y})$, Algorithm 2 computes $S_y = \emptyset$ and therefore $S_{xy} = S_x$. S_x contains by definition only single objectives f_i, where $f_i(\mathbf{x}) < f_i(\mathbf{y})$. Thus, S_{xy} contains all objective sets, which explain the relationship $\mathbf{x} \preceq_{\mathcal{F}} \mathbf{y} \wedge \neg(\mathbf{x} \sim \mathbf{y})$ w. r. t. $\preceq_{\mathcal{F}}$. (iii) For an incomparable solution pair $\mathbf{x}||\mathbf{y}$, no $f_i \in \mathcal{F}$ will be both in S_x and in S_y. Thus, S_{xy} contains only sets of objectives $\{i, j\}$ with cardinality 2 which matches the minimal size of S_{xy} if $\mathbf{x}||\mathbf{y}$ and for which $f_i(\mathbf{x}) < f_i(\mathbf{y}) \wedge f_j(\mathbf{x}) > f_j(\mathbf{y})$.

The computation of S_x and S_y can be done in time $O(k)$ and the calculation of S_{xy} is possible in time $O(k^2)$, as S_{xy} contains only $|S_{xy}| \leq |S_x| \cdot |S_y| \leq k^2$ sets. Since we know that S is a subset of $\mathcal{P}(\{1, \ldots, k\})$, S contains at most 2^k sets each of size $O(k)$. Hence, the computation of $S \sqcup S_{xy}$ needs time $O(k \cdot 2^k)$. Due to the fact that Algorithm 2 computes the sets for each pair of individuals, the whole running time results in $O(m^2 \cdot k \cdot 2^k)$. □

As the last aspect of our theoretical analysis, we present an instance for MOSS, for which the exact algorithm needs time $\Omega(m^2 \cdot 2^{k/3})$.

Theorem 6. *The worst-case running time of Algorithm 2 for the MOSS problem is* $\Omega(m^2 \cdot 2^{k/3})$.

Proof: Fig. 4 shows the idea of an instance I for which Algorithm 2 needs time $\Omega(m^2 \cdot 2^{k/3})$. Let us assume that I consists of an even number of m solutions $X :=$

$\{\mathbf{x}_1, \ldots, \mathbf{x}_m\}$ together with the relation $\preceq_{\mathcal{F}}$ and $k = 3/2 \cdot m$ relations $\preceq_i$ corresponding to the objective functions $\mathcal{F} := \{f_1, \ldots, f_{3/2 \cdot m}\}$ where only the solutions $\mathbf{x}_{2i-1}$ and $\mathbf{x}_{2i}$ for $1 \leq i \leq m/2$ are incomparable. The incomparability of such pairs is only caused by their $3i$th, $(3i+1)$th, and $(3i+2)$th objective values, i. e., we need either the objective pair f_{3i-2}, f_{3i-1} or the pair f_{3i-1}, f_{3i} to describe the incomparability, cf. Fig. 4. Thus, whenever Algorithm 2 considers a new pair $\mathbf{x}_{2i-1}, \mathbf{x}_{2i}$ of incomparable solutions, the size of the set S reduplicates. Because we have $m/2 = k/3$ of those incomparable pairs, S is of size $2^{k/3}$ after the algorithm considered all of the $k/3$ incomparable pairs. This is possible after the first $k/3$ of altogether $\binom{m}{2}$ steps of the algorithm, which results in a running time of at least $(\binom{m}{2} - k/3) \cdot 2^{k/3} = \Omega(m^2 \cdot 2^{k/3})$. In addition, this restricted example can be easily extended to the case $m > k$. $\square$

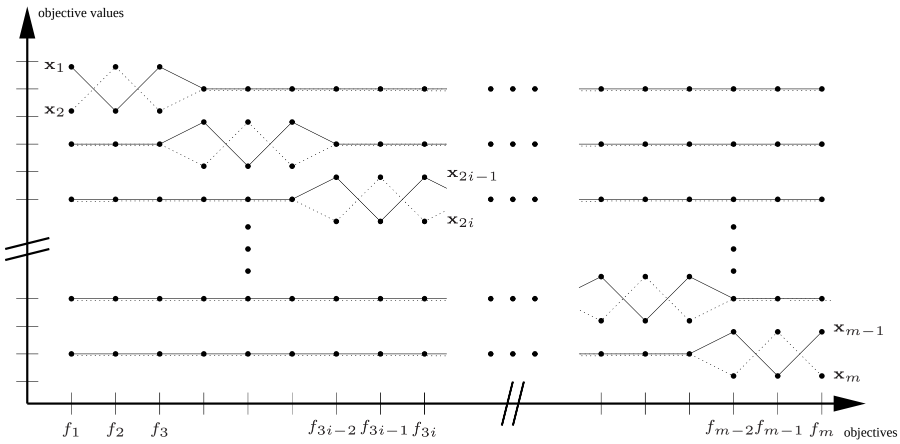


Fig. 4. The parallel coordinates plot of an instance for which the exact algorithm needs time $\Omega(m^2 \cdot 2^{k/3})$.

4 Experiments

The following experiments serve two goals: (i) to investigate the size of a minimum objective subset depending on the size of the search space and the number of original objective functions, and (ii) to compare the approximative and the exact algorithm with respect to the size of the generated objective subsets and the corresponding running times. Both issues have been considered both for a random problem and the multiobjective 0/1-knapsack problem.

4.1 Random Problem

In a first experiment we generated the objective values for a set of solutions X at random where the objective values were chosen uniformly distributed in $[0, 1] \subset \mathbb{R}$. For

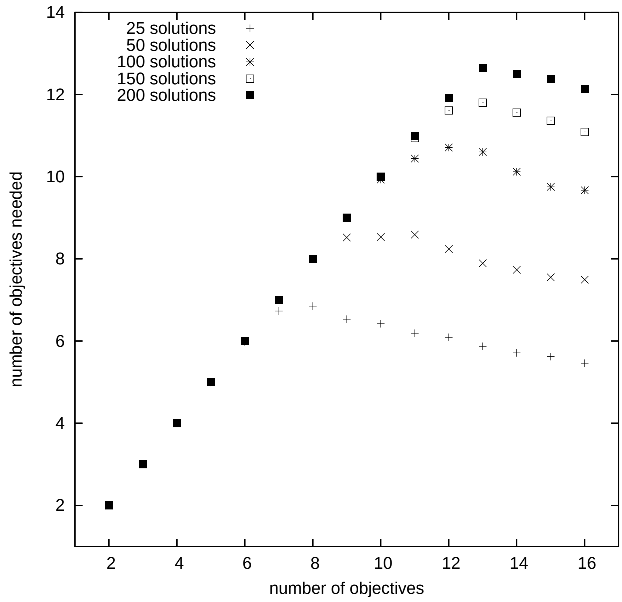


Fig. 5. Random model: The size of a minimum subset plotted against the number of objectives in the problem formulation.

each combination of search space size $|X|$ and number of objectives k, 100 independent random samples were considered. The results for Algorithm 2 are shown in Figure 5. For different sizes of the search space, the number $k_{\min}$ of objectives in a minimum objective subset is plotted against the number k of objectives used in the problem formulation. Two main observations can be made. First, the minimum number of objectives decreases the more objectives are involved as the fraction $k_{\min}/k$ decreases with rising number k of objectives in the problem formulation. Second, the larger the search space the more objectives are in a minimum objective set. Although there is no possibility to determine the course of the curves for arbitrary large number k of objectives with experiments, the question how $k_{\min}$ will behave with k increasing to infinity, arises. We expect $\lim_{k\to\infty} k_{\min} = 2$ because the probability that an existing objective pair occurs, the intersection of which fits the preorder on X, increases with higher k.

Concerning the comparison of the two algorithms, Fig. 6 reveals that the greedy algorithm yields similar sizes of the computed sets in comparison to the exact algorithm but is much faster than the latter. Already for a small search space of 32 solutions, the exact algorithm is only usable for k smaller than 15, whereas the running time of the greedy algorithm is competitive even for 50 objectives.

4.2 Knapsack Problem

We did further experiments on the 0/1-knapsack problem [13] with 10 items, the implementation was taken from the PISA package [1]. Instead of examining the whole

number k of objectives in problem formulation	5	10	15	20	25	30
exact algorithm: size of computed objective subset	4	5	8	13	16	13
greedy algorithm: size of computed objective subset	4	5	8	13	16	14
exact algorithm: running time in milliseconds	196	2,271	87,113	90,524	$\approx 2.5 \cdot 10^6$	$\approx 15 \cdot 10^6$
greedy algorithm running time in milliseconds	47	46	67	88	78	87

Table 1. The number of objectives in the computed subsets and the runtimes for an approximation of the Pareto Front, generated with SPEA2 after 1000 generations for the knapsack problem. The running times correspond to experiments on a linux computer (SunFireV60x with 3060 Mhz).

search space as in the random example, we generated an approximation of the Pareto set with a multiobjective evolutionary algorithm, namely SPEA2 [12] with the standard settings (population size $\mu = 50$, offspring population size $\lambda = 50$, $X = \{0,1\}^{10}$, 1000 generations). Both the exact and the approximation algorithm were applied to the generated Pareto set approximation. In addition, we recorded the running times of both algorithms. Table 1 shows the results for different sizes of the objective space.

The experiments show that the omission of objectives without information loss is possible even for a structured problem as the 0/1-knapsack problem. In comparison to the exact algorithm, the greedy algorithm shows nearly the same output quality for the used knapsack instances regarding the size of the computed objective set but is much faster. Due to the sizes of the computed subsets which are—in all of our experiments—less than one objective away from the optimum, the greedy algorithm seems to be applicable for more complex problems, particularly by virtue of its small running time.

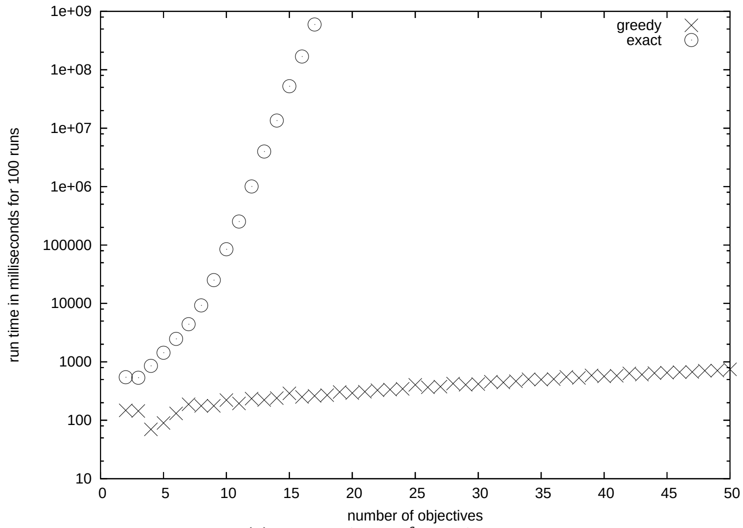
5 Discussion

This paper has investigated the minimum objective subset problem (`MOSS`) that asks which objective functions are essential for a given multiobjective optimization problem. To this end, we have introduced a general notion of conflicts between objective sets and showed that the answer to the above question can generally not be deduced from the information about conflicts between single objectives or objective sets of a predefined limited size. The latter observation motivates why `MOSS` turns out to be NP-hard. Furthermore, we have proposed an exact algorithm for `MOSS`, the running time of which is polynomial in the size m of the decision space but exponential in the number of objectives, and a polynomial greedy algorithm with an optimal approximation ratio of $\Theta(\log m)$.

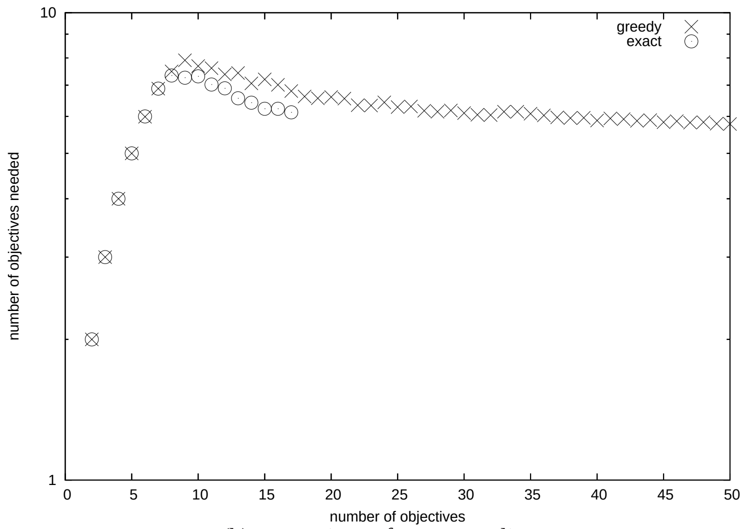
From a practical point of view, the present study provides a first step towards dimensionality reduction of the objective space in multiple criteria optimization scenarios. The proposed algorithms can be particularly useful to analyze Pareto sets or Pareto set approximations generated by exact resp. heuristic search procedures, but it is clear that an analysis of the entire search space is infeasible for most problems. Therefore, an important issue is the conflict analysis if only partial information about the search space is available as, e. g., during the optimization process. Furthermore, the experimental results for random objective functions as well as for the knapsack problem have revealed that a high percentage of objective can be omitted, especially if the number of objectives is high (10 or more). However, one may also be interested in a substantial reduction of the objective set in the case of few objectives; here, a modified `MOSS` problem where the search space order needs to be preserved only partially would be of high practical relevance.

References

1. Stefan Bleuler, Marco Laumanns, Lothar Thiele, and Eckart Zitzler. PISA — a platform and programming language independent interface for search algorithms. In *EMO 2003 Proceedings*, pages 494–508. Springer, Berlin, 2003.
2. Kalyanmoy Deb. *Multi-objective optimization using evolutionary algorithms*. Wiley, Chichester, UK, 2001.
3. Uriel Feige. A threshold of ln n for approximating set cover. *J. ACM*, 45(4):634–652, 1998.
4. Ian T. Jolliffe. *Principal component analysis*. Springer, 2002.
5. Joshua Knowles and David Corne. Properties of an adaptive archiving algorithm for storing nondominated vectors. *IEEE Transactions on Evolutionary Computation*, 7(2):100–116, 2003.
6. Robin C. Purshouse and Peter J. Fleming. Conflict, harmony, and independence: Relationships in evolutionary multi-criterion optimisation. In *EMO 2003 Proceedings*, pages 16–30. Springer, Berlin, 2003.
7. Petr Slavík. A tight analysis of the greedy algorithm for set cover. In *STOC '96: Proceedings of the twenty-eighth annual ACM symposium on Theory of computing*, pages 435–441, New York, NY, USA, 1996. ACM Press.
8. Kay Chen Tan, Eik Fun Khor, and Tong Heng Lee. *Multiobjective Evolutionary Algorithms and Applications*. Springer, London, 2005.
9. William T. Trotter. *Combinatorics and Partially Ordered Sets: Dimension Theory*. The Johns Hopkins University Press, Baltimore and London, 1992.
10. Lyndon While. A new analysis of the lebmeasure algorithm for calculating hypervolume. In *EMO 2005 Proceedings*, pages 326–340. Springer, 2005.
11. Mihalis Yannakakis. The complexity of the partial order dimension problem. *SIAM Journal on Algebraic and Discrete Methods, Vol. 3, No. 3, September 1982*, pages 351–358, 1982.
12. Eckart Zitzler, Marco Laumanns, and Lothar Thiele. SPEA2: Improving the Strength Pareto Evolutionary Algorithm for Multiobjective Optimization. In K.C. Giannakoglou et al., editors, *Evolutionary Methods for Design, Optimisation and Control with Application to Industrial Problems (EUROGEN 2001)*, pages 95–100. International Center for Numerical Methods in Engineering (CIMNE), 2002.
13. 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, 1999.
14. Eckart Zitzler, Lothar Thiele, Marco Laumanns, Carlos M. Foneseca, and Viviane Grunert da Fonseca. Performance assessment of multiobjective optimizers: An analysis and review. *IEEE Transactions on Evolutionary Computation*, 7(2):117–132, 2003.



(a) comparison of run times



(b) comparison of output quality

Fig. 6. Comparison between the greedy and the exact algorithm for the random problem and 32 solutions. Note that the plot of the running times in a) is a logscale plot and only the summed running times over 100 runs on a linux computer (SunFireV60x with 3060 MHz) are shown. Figure b) shows the sizes of the computed minimum / minimal sets averaged over 100 runs.

A Proofs of $\mathcal{NP}$-hardness

Here, we additionally provide the proofs omitted in Sec. 3.

Theorem 3. *The problem* MOSS *is* $\mathcal{NP}$*-hard.*

Proof: First, we denote the input size of MOSS by n, where $n = \Theta(k \cdot m^2)$ with $m := |X|$. We refer to Fig. 3 for a visualization of the ideas behind the Turing transformation SCP$\leq_T$ MOSS, which we recapitulate first.

Starting from the SCP instance consisting of the set $S = \{s_1, \dots, s_m\}$ and the subsets C_i with $1 \leq i \leq k$, all relations $\preceq_i$ as well as $\preceq_\mathcal{F}$ in the MOSS instance are defined on the basic set $X := \{\mathbf{x}_1, \dots, \mathbf{x}_m, \mathbf{x}'_1, \dots, \mathbf{x}'_m\}$. The relation $\preceq_\mathcal{F}$ will be the reflexive closure of the antichain on X, i. e., $\preceq_\mathcal{F}$ only contains the elements $(\mathbf{x}_j, \mathbf{x}_j)$ and $(\mathbf{x}'_j, \mathbf{x}'_j)$ for $1 \leq j \leq m$. The relations $\preceq_i$ with $1 \leq i \leq k$ are all constructed in the same way. They include the linear order $[\mathbf{x}_1, \mathbf{x}'_1, \mathbf{x}_2, \mathbf{x}'_2, \dots, \mathbf{x}_m, \mathbf{x}'_m]$ as well as the reflexive relations. Additionally, relation $\preceq_i$ contains the element $(\mathbf{x}'_j, \mathbf{x}_j)$ iff $s_j \notin C_i$. In addition, we have to compute another relation $\preceq_{k+1}$ which is the reverse linear order $[\mathbf{x}'_m, \mathbf{x}_m, \mathbf{x}'_{m-1}, \mathbf{x}_{m-1}, \dots, \mathbf{x}'_1, \mathbf{x}_1]$. After this transformation, we question our MOSS oracle once. The resulting index $I_{\texttt{SCP}}$ for the SCP problem will be then $I_{\texttt{SCP}} := I_{\text{oracle}} \setminus \{k + 1\}$ if the oracle produces I_{oracle} as its output.

It remains to show that the transformation yields to an exact algorithm for SCP with polynomial running time, under the assumption that there is an exact polynomial time algorithm A for MOSS. Let us assume that $(S = \{s_1, \dots, s_m\}, C_1, \dots, C_l)$ is the SCP instance with $C_i = \{c_1, \dots, c_{|C_i|}\} \subseteq S$. Via the described transformation and the hypothetical algorithm A, we can compute the index $I_{\texttt{SCP}} := I_A \setminus \{k+1\}$ as the output corresponding to the SCP instance S. Obviously, the computation of $I_{\texttt{SCP}}$ is possible in polynomial time using a polynomial algorithm for MOSS. To complete the proof, we still have to show (i) why always $k + 1 \in I_A$, (ii) why $I_A \setminus \{k + 1\}$ is a correct output for our SCP instance, and (iii) why the computed index $I_A \setminus \{k + 1\}$ is minimum.

First, we will take a look at the question (i) why always $k + 1 \in I_A$ for an exact MOSS algorithm A, i. e., why $\preceq_{k+1}$ is always needed to yield $\preceq_\mathcal{F}$ as the intersection of some $\preceq_i$. Because in $\preceq_\mathcal{F}$ no pair $\mathbf{x}, \mathbf{y} \in X$ with $\mathbf{x} \neq \mathbf{y}$ is comparable, for each pair $\mathbf{x}, \mathbf{y} \in X$, $\mathbf{x} \neq \mathbf{y}$, there has to be at least one $i \in I_A$ where $\mathbf{x} \npreceq_i \mathbf{y}$ and at least one $j \in I_A$ with $\mathbf{y} \npreceq_j \mathbf{x}$. Considering a pair $\mathbf{x}, \mathbf{y}$, for all $\preceq_i$ with $i \in \{1, \dots, k\}$ $\mathbf{x} \preceq_i \mathbf{y}$ holds. By construction, only $\mathbf{x} \npreceq_{k+1} \mathbf{y}$. Consequently, $\preceq_{k+1}$ is always needed, to construct $\preceq_\mathcal{F}$ as the intersection of single $\preceq_i$'s. Now we show (ii) why $I := I_A \setminus \{k+1\}$ is always a correct output for the given SCP instance. As we have seen before, $k + 1 \in I_A$ and therefore, the intersection of the $\preceq_i$'s does not contain any pairs $(\mathbf{x}_\nu, \mathbf{x}_\mu)$ and $(\mathbf{x}'_\nu, \mathbf{x}'_\mu)$ with $1 \leq \nu < \mu \leq m$ and no pairs $(\mathbf{x}_\nu, \mathbf{x}'_\nu)$ with $1 \leq \nu \leq m$. The construction of the relations $\preceq_i$ with $i \in \{1, \dots, k\}$ results in the absence of pairs $(\mathbf{x}_\nu, \mathbf{x}_\mu)$ and $(\mathbf{x}'_\nu, \mathbf{x}'_\mu)$ with $1 \leq \mu < \nu \leq m$ in the intersection if there will be at least one $i \in I_A$ with $1 \leq i \leq k$. There only remains the possibility of pairs $(\mathbf{x}'_\nu, \mathbf{x}_\nu)$ with $1 \leq \nu \leq m$ in the intersection. To avoid this, for each $\nu \in \{1, \dots, m\}$ there must be at least one $i \in \{1, \dots, k\}$ in I_A with $\mathbf{x}'_\nu \npreceq_i \mathbf{x}_\nu$. By construction of the Turing transformation, this can only occur, if $c_\nu \in C_i$. Thus, $\bigcup_{i \in I_A \setminus \{k+1\}} C_i = \{1, \dots, m\} = S$. Last, we have to show (iii) why the computed index $I_A \setminus \{k + 1\}$ is a minimum index for SCP. Assume that $I_A \setminus \{k+1\}$ is not a minimum index for SCP, i. e., there is a smaller index J with $|J| < |I|$ and $\bigcup_{j\in J} C_j = S$. As one can easily see from the above transformation, $J \cup \{k+1\}$ would be a smaller index for MOSS than I_A. □

Theorem 7. *The MOSS problem is Turing reducable to SCP.*

Proof: Given an instance for MOSS, consisting of the relations $\preceq_{\mathcal{F}} \subseteq X \times X$ and $\preceq_i \subseteq X \times X$ with $\bigcap_{1\leq i\leq k} \preceq_i = \preceq_{\mathcal{F}}$, a polynomial time algorithm A can compute an SCP instance as follows. The set S in the SCP instance contains one element $s_{x,y}$ for each $(\mathbf{x},\mathbf{y}) \notin \preceq_{\mathcal{F}}$. A subset C_i of S in the SCP instance contains an element $s_{\mathbf{x},\mathbf{y}}$ iff $\neg(\mathbf{x} \preceq_i \mathbf{y})$. The algorithm A can then use a hypothetical polynomial time bounded exact algorithm for SCP, to compute the index I as an output for the MOSS problem.

The index I, computed by the SCP algorithm, is always a correct output for the MOSS problem. To see that, we show $\forall 1 \leq i \leq k : C_i \subseteq S$, first. Let $s_{x,y} \in C_i$ for any $\mathbf{x}, \mathbf{y} \in X$ and any $1 \leq i \leq k$. By definition, $\neg(\mathbf{x} \preceq_i \mathbf{y})$, i. e., $\neg(f_i(\mathbf{x}) \leq f_i(\mathbf{y})) \iff f_i(\mathbf{x}) > f_i(\mathbf{y})$ holds. But then $\neg(\mathbf{x} \preceq_{\mathcal{F}} \mathbf{y})$, thus, $s_{x,y} \in S$ by definition.

Now, we are able to show that I is always a correct output for the MOSS problem. We only have to use the rules of deMorgan and the fact that $C_i \subseteq S$ holds for all $1 \leq i \leq k$.

$$
\begin{aligned}
\bigcup_{i\in I} C_i = S &\iff \forall s_{x,y} \in S : \exists i \in I : s_{x,y} \in C_i \\
&\iff \forall \mathbf{x}, \mathbf{y} \in X : [(\exists i \in I : s_{x,y} \in C_i) \Leftrightarrow s_{x,y} \in S] \\
&\iff \forall \mathbf{x}, \mathbf{y} \in X : [(\exists i \in I : \neg(\mathbf{x} \preceq_i \mathbf{y})) \Leftrightarrow \neg(\mathbf{x} \preceq_{\mathcal{F}} \mathbf{y})] \\
&\iff \forall \mathbf{x}, \mathbf{y} \in X : \left[\left(\exists i \in I : \mathbf{x} \preceq_i^C \mathbf{y}\right) \Leftrightarrow \mathbf{x} \preceq_{\mathcal{F}}^C \mathbf{y}\right] \\
&\iff \bigcup_{i\in I} \preceq_i^C = \preceq_{\mathcal{F}}^C \iff \bigcap_{i\in I} \preceq_i = \preceq_{\mathcal{F}}
\end{aligned}
$$

By construction, it is clear that a minimum I is always a minimum index for MOSS. □

B Relations between the different definitions of conflict

Before we present the relations between the different concepts of conflict, mentioned in Sec. 1, we restate the definitions of conflict according to the notation in Sec. 2 and prove a lemma we use later.

Definition 7 (Conflict by Deb [2]) *A multiobjective optimization problem (X, Z, f, rel) contains conflicting objectives if and only if there are trade-offs, i. e., the partially ordered set $(f(X), rel)$ has no unique minimal element.*

Definition 8 (Conflict by Tan et al. [8]) *A set $\mathcal{F}$ of objective functions is said to be nonconflicting according to the weak dominance relation $\preceq_{\mathcal{F}}$[8] if and only if there are no incomparable solution pairs, i. e., $\forall \mathbf{x}, \mathbf{y} \in X : \mathbf{x} \preceq_{\mathcal{F}} \mathbf{y} \vee \mathbf{y} \preceq_{\mathcal{F}} \mathbf{x}$.*

[8] Instead of $\preceq$, the dominance relation $\prec$ is used in the original definition in [8].

Definition 9 (Conflict by Purshouse and Fleming [6]) *Two objectives f_i and f_j are conflicting if there exists at least one solution pair $\mathbf{x}, \mathbf{y} \in X$ with $f_i(\mathbf{x}) < f_i(\mathbf{y}) \wedge f_j(\mathbf{x}) > f_j(\mathbf{y})$. If $f_i(\mathbf{x}) < f_i(\mathbf{y}) \wedge f_j(\mathbf{x}) > f_j(\mathbf{y})$ holds for all pairs, f_i and f_j are totally conflicting. There is no conflict between f_i and f_j if no such pair $\mathbf{x}, \mathbf{y}$ exist.*

Lemma 1. *For any set of objectives $\mathcal{F}$, there is no subset $\mathcal{F}' \subseteq \mathcal{F}$ which is strongly conflicting with $\mathcal{F}$ according to Def. 2.*

Proof: With Theorem 1 it is clear that $\bigcap_{1 \leq i \leq k} \preceq_i = \preceq_{\mathcal{F}}$ and therefore $\forall \mathcal{F}' \subseteq \mathcal{F}$: $(\mathbf{x}, \mathbf{y}) \in \preceq_{\mathcal{F}'}$ holds for all $(\mathbf{x}, \mathbf{y}) \in \preceq_{\mathcal{F}}$. For this reason it is impossible that

$$\preceq_{\mathcal{F}} \supset \preceq_{\mathcal{F}'} \iff \preceq_{\mathcal{F}} \not\subseteq \preceq_{\mathcal{F}'} \iff \mathcal{F} \not\trianglelefteq \mathcal{F}',$$

i. e., $\mathcal{F}'$ cannot strongly conflicting with $\mathcal{F}$ according to Def. 2. □

B.1 The relation to Deb's definition of conflict [2]

Theorem 8. *If a multiobjective optimization problem $(X, Z, f := (f_1, \ldots, f_k), \leq)$ contains conflicting objectives according to Def. 7 it is possible that there is an objective set $\mathcal{F}' \subset \mathcal{F} := \{f_1, \ldots, f_k\}$ which is nonconflicting or weakly conflicting with $\mathcal{F}$ but no $\mathcal{F}'$ which is strongly conflicting with $\mathcal{F}$. The same holds if the optimization problem contains no conflicts according to Def. 7*

Proof: Due to the fact that Def. 7 defines a conflict globally and only depending on the small set of minimal elements of the dominance relation, there is only weak relation between Def. 7 and our definition of conflict in Def. 2. Given a multiobjective optimization problem $(X, Z, f := (f_1, \ldots, f_k), \leq)$ with $\mathcal{F} := \{f_1, \ldots, f_k\}$, we know from Lemma 1 that there is no $\mathcal{F}' \subseteq \mathcal{F}$ which is strongly conflicting with $\mathcal{F}$. Fig. 7 shows for the case of a conflicting problem (a) and for a nonconflicting problem (b) that subsets $\mathcal{F}' \subseteq \mathcal{F}$ can be either nonconflicting or weakly conflicting with $\mathcal{F}$. □

Theorem 9. *If all subsets $\mathcal{F}' \subseteq \mathcal{F}$ are nonconflicting with $\mathcal{F}$ w. r. t. Def. 2, $\mathcal{F}$ contains no conflicting objectives according to Def. 7.*

Proof: If all subsets $\mathcal{F}' \subseteq \mathcal{F} := \{f_1, \ldots, f_k\}$ of a multiobjective optimization problem $(X, Z, f := (f_1, \ldots, f_k), \leq)$ are nonconflicting with $\mathcal{F}$ according to Def. 2, $f(X)$ cannot contain incomparable solutions w. r. t. $\preceq_{\mathcal{F}}$. Otherwise the relations $\preceq_i$ corresponding to single objective functions cannot be nonconflicting with $\preceq_{\mathcal{F}}$, because the $\preceq_i$'s are always total preorders, i. e., all solution pairs are comparable w. r. t. each $\preceq_i$. □

B.2 The relation to the conflict definitions of Tan, Khor, and Lee [8]

Theorem 10. *If a set $\mathcal{F}$ of objective functions is not conflicting according to Def. 8 it is possible that a subset $\mathcal{F}' \subseteq \mathcal{F}$ is nonconflicting with $\mathcal{F}$ or weakly conflicting with $\mathcal{F}$ according to Def. 2.*

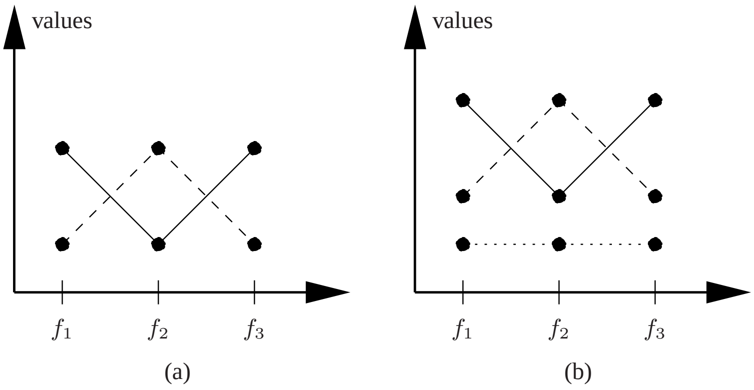


Fig. 7. Parallel coordinates plots of two multiobjective optimization problems with three objectives $\mathcal{F} := \{f_1, f_2, f_3\}$ which contain (a) a conflict and (b) no conflict according to Def. 7. The multiobjective optimization problem in (a) contains only two solutions and the problem in (b) three, where the dotted solution is the unique minimal element of $\preceq_{\mathcal{F}}$. Independant of Def. 7, there are subsets $\mathcal{F}', \mathcal{F}'' \subseteq \mathcal{F}$ which are both weakly conflicting with $\mathcal{F}$ ($\mathcal{F}' := \{f_1\}$) and nonconflicting with $\mathcal{F}$ ($\mathcal{F}'' := \{f_1, f_2\}$).

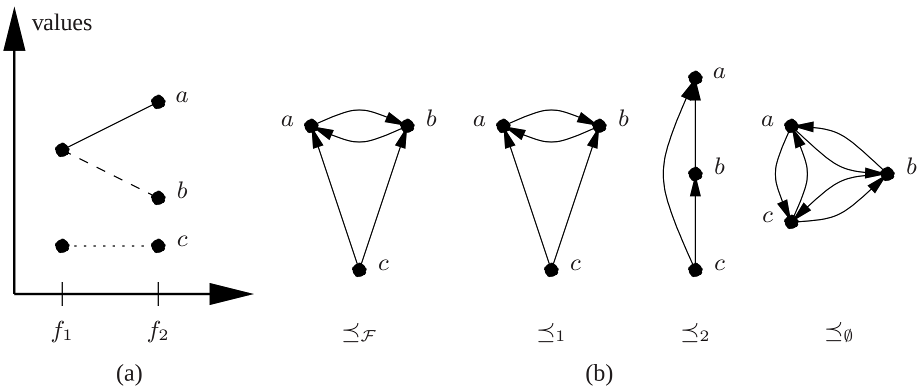


Fig. 8. (a) Parallel coordinates plot for an example with three solutions a (solid line), b (dashed), and c (dotted) and two objectives $\mathcal{F} := \{f_1, f_2\}$ with no conflict according to Def. 8. $\{f_1\}$ is nonconflicting with $\mathcal{F}$ whereas $\{f_2\}$ is weakly conflicting with $\mathcal{F}$. (b) shows the corresponding relation graphs of the involved relations $\mathcal{F}' \subseteq \mathcal{F}$.

Proof: Starting from a set $\mathcal{F}$ of objective functions which is not conflicting according to Def. 8, conclusions about the type of conflict (weak conflict or no conflict) between subsets of $\mathcal{F}' \subseteq \mathcal{F}$ and $\mathcal{F}$ itself are impossible. Fig. 8 shows that for an objective set $\mathcal{F}$ it is possible to have both a subset $\mathcal{F}' \subseteq \mathcal{F}$ which is nonconflicting with $\mathcal{F}$ and a subset $\mathcal{F}'' \subseteq \mathcal{F}$ which is weakly conflicting with $\mathcal{F}$. □

Theorem 11. *If all subsets $\mathcal{F}' \subseteq \mathcal{F}$ are nonconflicting with $\mathcal{F}$ according to Def. 2, $\mathcal{F}'$ is nonconflicting according to Def. 8.*

Proof: Given a multiobjective optimization problem $(X, Z, f := (f_1, \dots, f_k), \leq)$ where all subsets $\mathcal{F}' \subseteq \mathcal{F} := \{f_1 \dots, f_k\}$ are nonconflicting with $\mathcal{F}$ acoording to Def. 2. Then, there cannot be incomparable solutions $\mathbf{x}, \mathbf{y} \in X$ with respect to $\preceq_{\mathcal{F}}$, i. e., $\mathcal{F}$ is nonconflicting according to Def. 8 as at least one set $\{f_i\}$ will be strongly conflicting with $\mathcal{F}$, because two solutions $\mathbf{x}$ and $\mathbf{y}$ are always comparable with respect to each $\preceq_i$ and $\bigcap_{1 \leq i \leq k} \preceq_i = \preceq_{\mathcal{F}}$. □

B.3 The relation to the definitions of conflict by Purshouse and Fleming [6]

Theorem 12. *Between the two objectives f_i and f_j is no conflict according to Def. 9 if and only if f_i and f_j are nonconflicting according to Def. 2*

Proof: Let there be no conflict between the two objectives f_i and f_j according to Def. 9, i. e.,

$$\begin{aligned}
&\nexists \mathbf{x}, \mathbf{y} \in X : (f_i(\mathbf{x}) < f_i(\mathbf{y})) \wedge (f_j(\mathbf{x}) > f_j(\mathbf{y})) \\
&\iff \forall \mathbf{x}, \mathbf{y} \in X : [(f_i(\mathbf{x}) \leq f_i(\mathbf{y}) \wedge f_j(\mathbf{x}) \leq f_j(\mathbf{y})) \\
&\qquad\qquad \vee (f_i(\mathbf{x}) \geq f_i(\mathbf{y}) \wedge f_j(\mathbf{x}) \geq f_j(\mathbf{y}))] \\
&\iff \forall \mathbf{x}, \mathbf{y} \in X : [(\mathbf{x} \preceq_i \mathbf{y} \wedge \mathbf{x} \preceq_j \mathbf{y}) \vee (\mathbf{y} \preceq_i \mathbf{x} \wedge \mathbf{y} \preceq_j \mathbf{x})] \\
&\iff \forall \mathbf{x}, \mathbf{y} \in X : [(\mathbf{x}, \mathbf{y}) \in \preceq_i \Leftrightarrow (\mathbf{x}, \mathbf{y}) \in \preceq_j] \\
&\iff \preceq_i = \preceq_j,
\end{aligned}$$

which is the same than f_i and f_j are nonconflicting according to Def. 2. □

Theorem 13. *Two objectives f_i and f_j are in conflict according to Def. 9 if and only if f_i and f_j are either strongly conflicting or weakly conflicting according to Def. 2.*

Proof: By definition, f_i and f_j are in conflict according to Def. 9 if and only if

$$\begin{aligned}
&\exists x, y \in X : [f_i(x) < f_j \wedge f_j(x) > f_j(y)] \\
&\iff \neg \left(\nexists x, y \in X : [f_i(x) < f_j \wedge f_j(x) > f_j(y)] \right),
\end{aligned}$$

which is, by Theorem 12, the same as

$$\neg \left(f_i \text{ and } f_j \text{ are nonconflicting according to Def. 2} \right).$$

Because the different kinds of conflict in Def. 2 are disjoint, this is the same as f_i and f_j are either weakly conflicting or strongly conflicting. □