Dimensionality Reduction in Multiobjective Optimization with (Partial) Dominance Structure Preservation: Generalized Minimum Objective Subset Problems

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

TIK-Report No. 247
Institut für Technische Informatik und Kommunikationsnetze
ETH Zürich
Gloriastrasse 35,CH–8092 Zürich, Schweiz

April 2006

Abstract. Most of the available multiobjective evolutionary algorithms (MOEA) for approximating the Pareto set have been designed for and tested on low dimensional problems (≤ 3 objectives). However, it is known that problems with a high number of objectives cause additional difficulties in terms of the quality of the Pareto set approximation and running time. Furthermore, the decision making process becomes the harder the more objectives are involved. In this context, the question arises whether all objectives are necessary to preserve the problem characteristics. One may also ask under which conditions such an objective reduction is feasible, and how a minimum set of objectives can be computed. In this paper, we propose a general mathematical framework, suited to answer these three questions and corresponding algorithms, exact and heuristic ones. The heuristic variants are geared towards direct integration into the evolutionary search process. Moreover, extensive experiments for four well-known test problems show that substantial dimensionality reductions are possible on the basis of the proposed methodology.

1 Motivation

The field of multiobjective evolutionary algorithms (MOEA) has been rapidly growing over the last decade, and most of the publications deal with two- or three-dimensional problems [4]; however, studies addressing high-dimensional problems are rare [10, 3]. The main reason is that problems with a high number of objectives cause additional challenges wrt low-dimensional problems. Current algorithms, developed for problems with a low number of objectives, have difficulties to find a good Pareto set approximation for higher dimensions [7]. Even with the availability of sufficient computing resources, some methods are practically not useable for a high number of objectives; for example, algorithms based on the hypervolume indicator [8] have running times exponential in the number of objectives [9, 13]. Moreover, the decision maker's choice of an appropriate trade-off solution from a set of alternative solutions, generated by a MOEA, becomes difficult or infeasible with many objectives. In this context, several questions arise. On the one hand, one may ask whether it is possible to omit some of the objectives while preserving the problem characteristics, under which conditions such an objective reduction is feasible, and how a minimum set of objectives can be computed. On the other hand, if one allows changes in the problem structure while omitting objectives, one may ask how to quantify such structural changes and how to compute a minimum set of objectives according to such a qualitative measure. These research topics have gained only little attention in the literature so far. In some studies [5, 11, 12], the issue of objective conflicts has been discussed; however, the issue under which conditions, *in general*, objectives can be omitted and how a minimum objective subset can be computed has not been addressed. Deb and Saxena [6] proposed a method for reducing the number of objectives, based on principal component analysis. Roughly speaking, their method aims at keeping those objectives that can explain most of the variance in the objective space. However, it is not clear (i) how the objective reduction alters the dominance structure and (ii) what the quality of a generated objective subset is (no minimum guarantee).

In a previous work [2], we have tackled the above questions for the case that the problem structure must not be changed. In particular, we have presented the minimum objective subset problem (`MOSS`) which asks which objective functions are essential, have introduced a general notion of conflicts between objective sets, and have proposed an exact algorithm and a greedy heuristic for the $\mathcal{NP}$ hard `MOSS` problem. In practice, though, one may be interested in "allowing errors", i.e., slight changes of the dominance structure, in order to obtain a smaller minimum set of objectives. This continuative study addresses this issue. The key contributions are

– a generalized notion of conflicting objective sets extending [2],
– the introduction of a measure for variations of the dominance structure
– the definition of the problems δ-`MOSS` and `k-EMOSS`, as an extension of the `MOSS` problem to the objective reduction with allowed problem structure variations,
– an exact algorithm, capable to solve both, the δ-`MOSS` and the `k-EMOSS` problem, as well as heuristics for both problems,
– experimental results on four different high-dimensional problems, and
– a comparison between our approach and Deb and Saxena's method [6].

As such, this paper provides a basis for online dimensionality reduction in evolutionary multiobjective algorithms.

2 A Measure for Changes of the Dominance Structure

Without loss of generality, in this paper we consider a minimization problem with k objective functions $f_i : X \to \mathbb{R}$, $1 \leq i \leq k$, where the vector function $f := (f_1, \ldots, f_k)$ maps each solution $\mathbf{x} \in X$ to an objective vector $f(x) \in \mathbb{R}^k$. Furthermore, we assume that the underlying dominance structure is given by the weak Pareto dominance relation which is defined as follows: $\preceq_{\mathcal{F}'} := \{(\mathbf{x}, \mathbf{y}) \mid \mathbf{x}, \mathbf{y} \in X \wedge \forall f_i \in \mathcal{F}' : f_i(\mathbf{x}) \leq f_i(\mathbf{y})\}$, where $\mathcal{F}'$ is a set of objectives with $\mathcal{F}' \subseteq \mathcal{F} := \{f_1, \ldots, f_k\}$. For better readability, we will sometimes only consider the objective functions' indices, e.g., $\mathcal{F}' = \{1, 2, 3\}$ instead $\mathcal{F}' = \{f_1, f_2, f_3\}$. We say $\mathbf{x}$ *weakly dominates* $\mathbf{y}$ *wrt the objective set* $\mathcal{F}'$ ($\mathbf{x} \preceq_{\mathcal{F}'} \mathbf{y}$) if $(\mathbf{x}, \mathbf{y}) \in \preceq_{\mathcal{F}'}$. A solution $\mathbf{x}^* \in X$ is called *Pareto optimal* if there is no other $\mathbf{x} \in X$ that weakly dominates $\mathbf{x}^*$ wrt the set of all objectives. The set of all Pareto optimal solutions is called *Pareto set*, for which an approximation is sought. 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[1], but not a partial order on X.

In [2] we have proposed a method that computes for given solution set $A \subseteq X$ a minimum subset $\mathcal{F}'$ of objectives with $\mathcal{F}' \subseteq \{f_1, \ldots, f_k\}$ such that the dominance structure is preserved. In other words, for $\mathcal{F}'$ holds that $\mathbf{x} \preceq_{\mathcal{F}'} \mathbf{y} \Longleftrightarrow \mathbf{x} \preceq_{\{f_1,\ldots,f_k\}} \mathbf{y}$ for all $\mathbf{x}, \mathbf{y} \in A$. This is illustrated in the following example.

Example 1 *Fig. 1 shows the parallel coordinates plot, cf. [11], of three solutions* $\mathbf{x}_1$ *(solid line),* $\mathbf{x}_2$ *(dashed) and* $\mathbf{x}_3$ *(dotted) that are pairwise incomparable.*
At a closer inspection, the objective functions f_1 *and* f_3 *indicate redundancy in the problem formulation, as the corresponding relations* $\preceq_{f_1}$ *and* $\preceq_{f_3}$ *are the same:* $\mathbf{x}_3 \preceq_{f_1} \mathbf{x}_1 \preceq_{f_1} \mathbf{x}_2$ *as well as* $\mathbf{x}_3 \preceq_{f_3} \mathbf{x}_1 \preceq_{f_3} \mathbf{x}_2$*. The approach of [2], therefore, computes the set* $\{f_1, f_2, f_4\}$ *as a minimum objective set which preserves the dominance structure, i.e.,* $\mathbf{x} \preceq_{\{f_1,f_2,f_4\}} \mathbf{y}$ *if and only if* $\mathbf{x} \preceq_{\mathcal{F}} \mathbf{y}$*, because all solutions are also pairwise incomparable wrt to* $\{f_1, f_2, f_4\}$*. That there is, for this example, no objective subset with less than three objectives, preserving the dominance structure, can be easily checked by hand.*

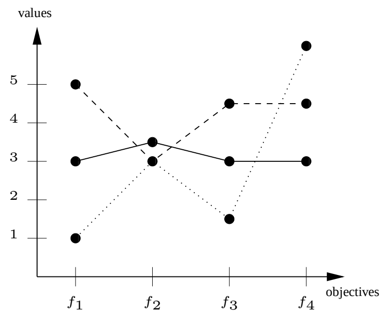


Fig. 1. Parallel coordinates plot for three solutions and four objectives.

In practice, one is often interested in a further dimensionality reduction at the cost of slight changes in the dominance structure. This poses the question how such a structural change can be quantitatively measured and how one can compute a minimum objective set for a given threshold on the degree of change.

A first attempt for a further objective reduction by using the ε-dominance[2] [16] relation instead of the weak dominance relation, as proposed in [2], failed. The use of the ε-dominance relation yields larger objective sets when allowing a larger error, what is both counterintuitive and impractical, cf. Fig. 2 for a random example. Nevertheless, we give another intuitive approach of further dimensionality reduction, including the definition of a measure for changes in the dominance structure.

Example 2 *Consider, once again, Fig. 1 and the objective subset* $\mathcal{F}' := \{f_3, f_4\}$*. We observe that by reducing the set of objectives to* $\mathcal{F}'$*, the dominances change: on the*

[1] A relation rel is called a preorder iff it is reflexive and transitive; a preorder that is antisymmetric is denoted as partial order.

[2] $\preceq^{\varepsilon}_{\mathcal{F}'} := \{(\mathbf{x}, \mathbf{y}) \,|\, \mathbf{x}, \mathbf{y} \in X \wedge \forall i \in \mathcal{F}' \subseteq \mathcal{F} : f_i(\mathbf{x}) - \varepsilon \leq f_i(\mathbf{y})\}$

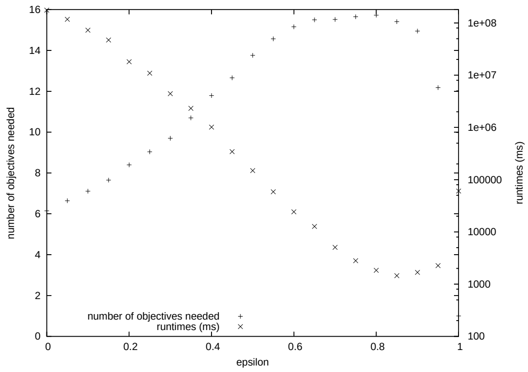


Fig. 2. The approach in [2] with different ε-dominance relations instead of the weak Pareto dominance yields larger objective subsets for an instance of 32 randomly chosen objective vectors and 16 objectives.

one hand $\mathbf{x}_1 \preceq_{\mathcal{F}'} \mathbf{x}_2$; on the other hand $\mathbf{x}_1 \not\preceq_{\mathcal{F}} \mathbf{x}_2$. In this sense, we make an error: the objective values of $\mathbf{x}_1$ had to be smaller by an additive term of $\delta = 0.5$, such that $\mathbf{x}_1 \preceq_{\mathcal{F}} \mathbf{x}_2$ would actually hold. This δ value can be used as a measure to quantify the difference in the dominance structure induced by $\mathcal{F}'$ and $\mathcal{F}$. By computing the δ values for all solution pairs $\mathbf{x}, \mathbf{y}$, we can then determine the maximum error. The meaning of the maximum δ value is that whenever we wrongly assume that $\mathbf{x} \preceq_{\mathcal{F}'} \mathbf{y}$, we also know that $\mathbf{x}$ is not worse than $\mathbf{y}$ in all objectives by an additive term of δ. For $\mathcal{F}' := \{f_3, f_4\}$, the maximum error is $\delta = 0.5$; for $\mathcal{F}' := \{f_2, f_4\}$, the maximum δ is 4.

In the following, we formalize the definition of error, according to the above example. The background for that is provided by the (additive) ε-dominance relation and a generalization of the notion of conflicts between objective sets, defined in [2]. Before, we introduce a general Pareto dominance relation. Subsequent observations on the new dominance relation show the properties of the $(\delta_1, \ldots, \delta_k)$-dominance relation, which are essential for the problems and algorithms proposed in the remainder of this paper.

Definition 1 *Let $\delta_1, \ldots, \delta_k \in \mathbb{R}$ and $\mathcal{F}_1, \ldots, \mathcal{F}_k$ objective subsets. We define the $(\delta_1, \ldots, \delta_k)$-dominance relation on X for all $\mathbf{x}, \mathbf{y} \in X$ as*

$$\mathbf{x} \preceq^{\delta_1, \ldots, \delta_k}_{\mathcal{F}_1, \ldots, \mathcal{F}_k} \mathbf{y} :\Longleftrightarrow \forall 1 \leq i \leq k : \forall j \in \mathcal{F}_i : f_j(\mathbf{x}) - \delta_i \leq f_j(\mathbf{y}).$$

Note, that the defined $(\delta_1, \ldots, \delta_k)$-dominance relation is a generalization of familiar dominance relations on all objectives $\mathcal{F}$ like the weak dominance relation $\preceq := \preceq^0_{\mathcal{F}}$ or the ε-dominance relation $\preceq^{\varepsilon} := \preceq^{\varepsilon}_{\mathcal{F}}$, defined above. Furthermore, we define $\preceq_{\mathcal{F}'} := \preceq^0_{\mathcal{F}'}$, $\preceq_i := \preceq^0_{\{f_i\}}$, and $\preceq^{\delta}_i := \preceq^{\delta}_{\{f_i\}}$ for arbitrary $\mathcal{F}' \subseteq \mathcal{F}$ and $\{f_i\} \in \mathcal{F}$. The notion for a relation's restriction to an objective subset $\mathcal{F}'$ will be used for any relation, such as $||_{\mathcal{F}'}$ and $\sim_{\mathcal{F}'}$.

Observation 1 *Let $\delta_1, \ldots, \delta_k, \delta'_1, \ldots, \delta'_k \in \mathbb{R}$ with $\forall 1 \leq i \leq k : \delta_i \leq \delta'_i$, and $\mathcal{F}_1, \ldots, \mathcal{F}_k, \mathcal{F}'_1, \ldots, \mathcal{F}_k$ objective sets with $\forall 1 \leq i \leq k : \mathcal{F}'_i \subseteq \mathcal{F}_i$. Then both $\preceq^{\delta_1, \ldots, \delta_k}_{\mathcal{F}_1, \ldots, \mathcal{F}_k} \subseteq \preceq^{\delta'_1, \ldots, \delta'_k}_{\mathcal{F}_1, \ldots, \mathcal{F}_k}$ and $\preceq^{\delta_1, \ldots, \delta_k}_{\mathcal{F}_1, \ldots, \mathcal{F}_k} \subseteq \preceq^{\delta_1, \ldots, \delta_k}_{\mathcal{F}'_1, \ldots, \mathcal{F}'_k}$ holds.*

Observation 2 *Furthermore,* $\preceq^{\delta_1,\ldots,\delta_k}_{\mathcal{F}_1,\ldots,\mathcal{F}_k} = \bigcap_{1\leq i\leq k} \preceq^{\delta_i}_{\mathcal{F}_i}$ *and* $\preceq^{\delta,\ldots,\delta}_{\mathcal{F}_1,\ldots,\mathcal{F}_k} = \preceq^{\delta}_{\bigcup_i \mathcal{F}_i}$.

Observation 3 *Let* $\delta \in \mathbb{R}$ *and* $f_i \in \mathcal{F}$ *for all* $1 \leq i \leq k$*. Then*

$$\bigcap_{i\in\mathcal{F}} \preceq^{\delta}_{i} = \preceq^{\delta}_{\mathcal{F}} .$$

Now, we come to the mathematical definition of error, according to Example 2, including a general definition of conflicting objective sets.

Definition 2 *Let* $\mathcal{F}_1$ *and* $\mathcal{F}_2$ *two objective sets. We define* $\mathcal{F}_1 \sqsubseteq^{\delta} \mathcal{F}_2 :\Longleftrightarrow \preceq_{\mathcal{F}_1} \subseteq \preceq^{\delta}_{\mathcal{F}_2}$.

Definition 3 *Let* $\mathcal{F}_1$ *and* $\mathcal{F}_2$ *two objective sets. We call*

- $\mathcal{F}_1$ δ-nonconflicting *with* $\mathcal{F}_2$ *iff* $\left(\mathcal{F}_1 \sqsubseteq^{\delta} \mathcal{F}_2\right) \wedge \left(\mathcal{F}_2 \sqsubseteq^{\delta} \mathcal{F}_1\right)$.
- $\mathcal{F}_1$ weakly δ-conflicting *with* $\mathcal{F}_2$ *if either* $\neg\left(\mathcal{F}_1 \sqsubseteq^{\delta} \mathcal{F}_2\right) \wedge \left(\mathcal{F}_2 \sqsubseteq^{\delta} \mathcal{F}_1\right)$ *or* $\left(\mathcal{F}_1 \sqsubseteq^{\delta} \mathcal{F}_2\right) \wedge \neg\left(\mathcal{F}_2 \sqsubseteq^{\delta} \mathcal{F}_1\right)$.
- $\mathcal{F}_1$ strongly δ-conflicting *with* $\mathcal{F}_2$ *if* $\neg\left(\mathcal{F}_1 \sqsubseteq^{\delta} \mathcal{F}_2\right) \wedge \neg\left(\mathcal{F}_2 \sqsubseteq^{\delta} \mathcal{F}_1\right)$.

The above definition of δ-nonconflicting objective sets is useful for changing a problem formulation by considering a different objective set. If a multiobjective optimization problem uses the objective set $\mathcal{F}_1$ and one can prove that $\mathcal{F}_1$ is δ-nonconflicting with another objective set $\mathcal{F}_2$, one can easily replace $\mathcal{F}_1$ with $\mathcal{F}_2$ and can be sure that in the new formulation, for any $\mathbf{x}, \mathbf{y} \in X$, $\mathbf{x}$ either weakly dominates $\mathbf{y}$ wrt $\mathcal{F}_2$ or $\mathbf{x}$ ε-dominates $\mathbf{y}$ wrt $\mathcal{F}_2$ if $\mathbf{x}$ weakly dominates $\mathbf{y}$ wrt $\mathcal{F}_1$ and $\varepsilon = \delta$. In the special case of an objective subset $\mathcal{F}' \subseteq \mathcal{F}$, δ-nonconflicting with all objectives $\mathcal{F}$, the definition fits the intuitive measure of error in Example 2. If an objective subset $\mathcal{F}' \subset \mathcal{F}$ is δ-nonconflicting with the set $\mathcal{F}$ of all objectives, $\mathbf{x}$ δ-dominates $\mathbf{y}$, i.e., $\forall i \in \mathcal{F} : f_i(\mathbf{x}) - \delta \leq f_i(\mathbf{y})$, whenever $\mathbf{x}$ weakly dominates $\mathbf{y}$ wrt the reduced objective set $\mathcal{F}'$. We, then, can omit all objectives in $\mathcal{F} \setminus \mathcal{F}'$ without making a larger error than δ in the omitted objectives.

The following theorems on the definitions of δ-conflicts are essential for the algorithms, we present in Sec. 3.2. The proofs are omitted here but can be found in Appendix A.

Theorem 1. *Let* $\mathcal{F}' \subseteq \mathcal{F}$*. Then* $\mathcal{F}'$ *is* δ*-nonconflicting with* $\mathcal{F}$ *if and only if* $\preceq_{\mathcal{F}'} \subseteq \preceq^{\delta}_{\mathcal{F}}$.

Theorem 2. *Let* $\mathcal{F}_1, \mathcal{F}_2$ *two objective sets and* X *a decision space. If*

$$\delta' := \max_{\substack{\mathbf{x},\mathbf{y}\in X \wedge \mathbf{x}\preceq_{\mathcal{F}_1}\mathbf{y} \\ i\in\mathcal{F}_2}} \{f_i(\mathbf{x}) - f_i(\mathbf{y})\} \text{ and } \delta'' := \max_{\substack{\mathbf{x},\mathbf{y}\in X \wedge \mathbf{x}\preceq_{\mathcal{F}_2}\mathbf{y} \\ i\in\mathcal{F}_1}} \{f_i(\mathbf{x}) - f_i(\mathbf{y}),\}$$

then, $\mathcal{F}_1$ *is* $\overline{\delta}$*-nonconflicting with* $\mathcal{F}_2$ *wrt* X *for all* $\overline{\delta} \geq \max(\delta', \delta'')$ *and no* $\underline{\delta} < \max\{\delta', \delta''\}$ *exists such that* $\mathcal{F}_1$ *is* $\underline{\delta}$*-nonconflicting with* $\mathcal{F}_2$.

Note, that if $\mathcal{F}_1 \subseteq \mathcal{F}_2$, the theorem can be shortened to $\mathcal{F}_1$ is δ-nonconflicting with $\mathcal{F}_2$ for all $\delta \geq \delta'$ but for no $\delta < \delta'$ if $\delta' := \max_{\mathbf{x},\mathbf{y}\in X\wedge\mathbf{x}\preceq_{\mathcal{F}_1}\mathbf{y}, i\in\mathcal{F}_2}\{f_i(\mathbf{x}) - f_i(\mathbf{y})\}$.

Based on the above conflict definitions, we will now formalize the notion of δ-minimal and δ-minimum objective sets including the corresponding notion for $\delta = 0$ in [2] and, furthermore, present a condition under which an objective reduction is possible.

Definition 4 *Let $\mathcal{F}$ be a set of objectives and $\delta \in \mathbb{R}$. An objective set $\mathcal{F}' \subseteq \mathcal{F}$ is denoted as*

– δ-minimal wrt $\mathcal{F}$ *iff (i) $\mathcal{F}'$ is δ-nonconflicting with $\mathcal{F}$, (ii) $\mathcal{F}'$ is δ'-conflicting with $\mathcal{F}$ for all $\delta' < \delta$, and (iii) there exists no $\mathcal{F}'' \subset \mathcal{F}'$ that is δ-nonconflicting with $\mathcal{F}$;*
– δ-minimum wrt $\mathcal{F}$ *iff (i) $\mathcal{F}'$ is δ-minimal wrt $\mathcal{F}$, and (ii) there exists no $\mathcal{F}'' \subset \mathcal{F}$ with $|\mathcal{F}''| < |\mathcal{F}'|$ that is δ-minimal wrt $\mathcal{F}$.*

A δ-minimal objective set is a subset of the original objectives that cannot be further reduced without changing the associated dominance structure with an error of at most δ. A δ-minimum objective set is the smallest possible set of original objectives that preserves the original dominance structure except for an error of δ. By definition, every δ-minimum objective set is δ-minimal, but not all δ-minimal sets are at the same time δ-minimum.

Definition 5 *A set $\mathcal{F}$ of objectives is called* δ-redundant *if and only if there exists $\mathcal{F}' \subset \mathcal{F}$ that is δ-minimal wrt $\mathcal{F}$.*

This definition of δ-redundancy represents a necessary and sufficient condition for the omission of objectives while the obtained dominance relation preserve the most of the initial dominance relation according to the definition of error in Example 2.

3 Identifying Minimum Objective Subsets

After the definition of an objective subset's error regarding its dominance structure, we present the two problems δ-MOSS and k-EMOSS, dealing with the two questions, mentioned in the introduction: On the one hand, the computation of an objective subset of minimum size, yielding a (changed) dominance structure with given error, and, on the other hand, the computation of an objective subset of given size with the minimum error. Furthermore, we present an exact algorithm, capable of solving both the δ-MOSS and the k-EMOSS problem, and afterwards approximation algorithms for each of the two problems, that are fast and designed for the integration into the search process.

3.1 The δ-MOSS and k-EMOSS Problems

Based on the definitions in Sec. 2, the problem MINIMUM OBJECTIVE SUBSET (MOSS), proposed in [2], can be characterized as follows. Given a multiobjective optimization problem, a given instance consists of the set A 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 $\bigcap_{1 \leq i \leq k} \preceq_i = \preceq_{\mathcal{F}}$. We then ask for a 0-minimum objective set $\mathcal{F}' \subseteq \mathcal{F}$ wrt $\mathcal{F}$. This problem can easily be generalized to the following problem, when allowing an error δ.

Definition 6 *Given a multiobjective optimization problem, the problem δ-MINIMUM OBJECTIVE SUBSET (δ-MOSS) is defined as follows.*

Instance:	*The objective vectors $f(\mathbf{x}_1), \ldots, f(\mathbf{x}_m) \in \mathbb{R}^k$ of the solutions $\mathbf{x}_1, \ldots, \mathbf{x}_m \in A \subseteq X$ and a $\delta \in \mathbb{R}$.*
Task:	*Compute a δ-minimum objective subset $\mathcal{F}' \subseteq \mathcal{F}$ wrt $\mathcal{F}$.*

Note, that the limitation of the instances to the whole search space description is not essential here. Since the objective values are only known for a small set of solutions in practice, and not for the entire search space, Pareto set approximations, e.g., given by a MOEA's population, can also be the underlying set A of solutions. Note also, that the set A and the relations $\preceq_i$, $\preceq_{\mathcal{F}}$ $\preceq_i$ are only given implicitly in a δ-MOSS instance. Nevertheless, δ-MOSS is a generalization of MOSS and therefore $\mathcal{NP}$ hard, as the following theorem shows.

Theorem 3. *δ-MOSS is $\mathcal{NP}$-hard, since 0-MOSS $=_T$ MOSS.*

Proof. The only difference between the problems 0-MOSS and MOSS are their input instances. Thus, we can show a Turing reduction 0-MOSS $\leq_T$ MOSS as well as MOSS $\leq_T$ 0-MOSS, and we solely have to show an efficient transformation from one instance into the other and vice versa. ($\leq_T$) If we pool together indifferent solutions in the relations $\preceq_i$ first, we can compute the objective values for the δ-MOSS instance with a topological sorting of the relations $\preceq_i$, simplified this way. The topological number of solution $\mathbf{x}$ in the topological sorting of $\preceq_i$ yields its ith objective value, i.e., two indifferent solutions wrt relation $\preceq_i$ get the same objective value and a solution $\mathbf{x}$ gets a higher value in objective f_i than $\mathbf{y}$ iff $\mathbf{x} \preceq_i \mathbf{x} \wedge \mathbf{y} \not\preceq_i \mathbf{x}$. The topological sorting needs time $O(k \cdot m^2)$ and the required search for indifferent solutions before time $O(m^2)$ for each of the k relations $\preceq_i$. The whole instance transformation, thus, needs time $O(k^2 \cdot m^2)$. ($\geq_T$) We can compute the relations $\preceq_i$ and $\preceq_{\mathcal{F}}$ simply from the given objective vectors in time $O(k \cdot m^2)$ by considering each pair $\mathbf{x}, \mathbf{y} \in X$ successively.

Since we know from [2] that MOSS is $\mathcal{NP}$-hard, 0-MOSS is $\mathcal{NP}$-hard as a result of the above transformation, i.e., δ-MOSS, in general, is $\mathcal{NP}$-hard, too. □

As a variation of the δ-MOSS problem, we introduce the problem of finding an objective subset of size $\leq$ k with minimum error according to $\mathcal{F}$.

Definition 7 *Given a multiobjective optimization problem, the problem MINIMUM OBJECTIVE SUBSET OF SIZE k WITH MINIMUM ERROR (k-EMOSS) is defined as follows.*

Instance: *The objective vectors $f(\mathbf{x}_1), \ldots, f(\mathbf{x}_m) \in \mathbb{R}^k$ of the solutions $\mathbf{x}_1, \ldots, \mathbf{x}_m \in A \subseteq X$ and a k $\in \mathbb{R}$.*

Task: *Compute an objective subset $\mathcal{F}' \subseteq \mathcal{F}$ which has size $|\mathcal{F}'| \leq$ k and is δ-nonconflicting with $\mathcal{F}$ with the minimal possible δ.*

3.2 Algorithms

An Exact Algorithm. Algorithm 1, as a generalization of the exact algorithm for the MOSS problem [2], solves both the δ-MOSS and the k-EMOSS problem exactly in exponential time. Thus, it can only solve small problem instances in reasonable time. The basic idea is to consider all solution pairs $(\mathbf{x}, \mathbf{y})$ successively and store in S_M all minimal objective subsets $\mathcal{F}'$ together with the minimal δ' value such that $\mathcal{F}'$ is δ'-nonconflicting with the set $\mathcal{F}$ of all objectives when taking into account only the solution pairs in M, considered so far.

The algorithm uses a subfunction $\delta_{\min}(\mathcal{F}_1, \mathcal{F}_2)$, that computes the minimal δ error for two solutions $\mathbf{x}, \mathbf{y} \in X$, such that $\mathcal{F}_1$ is δ-nonconflicting with $\mathcal{F}_2$ wrt $\mathbf{x}, \mathbf{y}$ according

Algorithm 1 An exact algorithm for δ-MOSS and k-EMOSS

1: Init:
2: $\quad M := \emptyset, \quad S_M := \emptyset$
3: **for all** pairs $\mathbf{x}, \mathbf{y} \in X$, $\mathbf{x} \neq \mathbf{y}$ of solutions **do**
4: $\quad S_{\{(\mathbf{x},\mathbf{y})\}} := \emptyset$
5: $\quad$ **for all** objective pairs $i, j \in \mathcal{F}$, not necessary $i \neq j$ **do**
6: $\quad\quad$ compute $\delta_{ij} := \delta_{\min}(\{i\} \cup \{j\}, \mathcal{F})$ wrt $\mathbf{x}, \mathbf{y}$
7: $\quad\quad S_{\{(\mathbf{x},\mathbf{y})\}} := S_{\{(\mathbf{x},\mathbf{y})\}} \sqcup (\{i\} \cup \{j\}, \delta_{ij})$
8: $\quad$ **end for**
9: $\quad S_{M \cup \{(\mathbf{x},\mathbf{y})\}} := S_M \sqcup S_{\{(\mathbf{x},\mathbf{y})\}}$
10: $\quad M := M \cup \{(\mathbf{x}, \mathbf{y})\}$
11: **end for**
12: Output for δ-MOSS: $\quad (s_{\min}, \delta_{\min})$ in S_M with minimal size $|s_{\min}|$ and $\delta_{min} \leq \delta$
13: Output for k-EMOSS: $\quad (s, \delta)$ in S_M with size $|s| \leq$ k and minimal δ

to Theorem 2. Furthermore, Algorithm 1 computes the union $\sqcup$ of two sets of objective subsets with simultaneous deletion of not δ'-minimal pairs $(\mathcal{F}', \delta')$:

$$\begin{aligned} S_1 \sqcup S_2 := \{ & (\mathcal{F}_1 \cup \mathcal{F}_2, \max\{\delta_1, \delta_2\}) \mid (\mathcal{F}_1, \delta_1) \in S_1 \wedge (\mathcal{F}_2, \delta_2) \in S_2 \\ & \wedge \nexists (\mathcal{F}'_1, \delta'_1) \in S_1, (\mathcal{F}'_2, \delta'_2) \in S_2 : \big(\mathcal{F}'_1 \cup \mathcal{F}'_2 \subset \mathcal{F}_1 \cup \mathcal{F}_2 \wedge \max\{\delta'_1, \delta'_2\} \leq \max\{\delta_1, \delta_2\}\big) \\ & \wedge \nexists (\mathcal{F}'_1, \delta'_1) \in S_1, (\mathcal{F}'_2, \delta'_2) \in S_2 : \big(\mathcal{F}'_1 \cup \mathcal{F}'_2 \subseteq \mathcal{F}_1 \cup \mathcal{F}_2 \wedge \max\{\delta'_1, \delta'_2\} < \max\{\delta_1, \delta_2\}\big) \} \end{aligned}$$

The correctness proof of Algorithm 1—as well as the proof of its running time of $O(m^2 \cdot k \cdot 2^k)$—can be found in Appendix B. Note, that the exact algorithm can be easily parallelized, as the computation of the sets $S_{\{(\mathbf{x},\mathbf{y})\}}$ are independent for different pairs $(\mathbf{x}, \mathbf{y})$. It also can be accelerated if line 9 of Algorithm 1 is tailored to either the δ-MOSS or the k-EMOSS problem by including a pair $(\mathcal{F}', \delta')$ into $S_{M \cup \{(\mathbf{x},\mathbf{y})\}}$ only if $\delta' \leq \delta$, and $|\mathcal{F}'| \leq$ k respectively.

A Greedy Algorithm for δ-MOSS. Algorithm 2, as an approximation algorithm for δ-MOSS, computes an objective subset $\mathcal{F}'$, δ-nonconflicting with the set $\mathcal{F}$ of all objectives in a greedy way. Starting with an empty set $\mathcal{F}'$, Algorithm 2 chooses in each step the objective f_i which yields the smallest set $\preceq_{\mathcal{F}'} \cap \preceq_i$ without considering the relationships in $\preceq^{0,\delta}_{\mathcal{F}' \cup \{i\}, \mathcal{F}}$ until $\mathcal{F}'$ is δ-nonconflicting with $\mathcal{F}$. For the correctness proof of Algorithm 2 and the proof of its running time of $O(\min\{k^3 \cdot m^2, k^2 \cdot m^4\})$ we once again refer to Appendix B. Note, that Algorithm 2 not necessarily yields a δ-minimal or even δ-minimum objective set wrt $\mathcal{F}$.

A Greedy Algorithm for k-EMOSS. Algorithm 3 is an approximation algorithm for k-EMOSS. It supplies always an objective subset of size k but does not guarantee to find the set with minimal δ. The greedy algorithm needs time $O(m^2 \cdot k^3)$ since at most k $\leq k$ loops with k calls of the $\delta_{\min}$ subfunction are needed. One call of the $\delta_{\min}$ function needs time $\Theta(m^2 \cdot k)$ and all other operations need time $O(1)$ each. Note, that Algorithm 3 can be accelerated in a concrete implementation as the while loop can be aborted if either $|\mathcal{F}'| =$ k or $\delta_{min}(\mathcal{F}', \mathcal{F}) = 0$.

Algorithm 2 A greedy algorithm for δ-`MOSS`.

Init:
  compute the relations $\preceq_i$ for all $1 \leq i \leq k$ and $\preceq_{\mathcal{F}}$
  $\mathcal{F}' := \emptyset$
  $R := X \times X \setminus \preceq_{\mathcal{F}}$
while $R \neq \emptyset$ **do**
  $i^* = \underset{i \in \mathcal{F} \setminus \mathcal{F}'}{\operatorname{argmin}} \{ |(R \cap \preceq_i) \setminus \preceq^{0,\delta}_{\mathcal{F}' \cup \{i\}, \mathcal{F} \setminus (\mathcal{F}' \cup \{i\})} | \}$
  $R := (R \cap \preceq_{i^*}) \setminus \preceq^{0,\delta}_{\mathcal{F}' \cup \{i^*\}, \mathcal{F} \setminus (\mathcal{F}' \cup \{i^*\})}$
  $\mathcal{F}' := \mathcal{F}' \cup \{i^*\}$
end while

Algorithm 3 A greedy algorithm for `k-EMOSS`

Init:
  $\mathcal{F}' := \emptyset$
while $|\mathcal{F}'| < \mathtt{k}$ **do**
  $\mathcal{F}' := \mathcal{F}' \cup \underset{i \in \mathcal{F} \setminus \mathcal{F}'}{\operatorname{argmin}} \{ \delta_{\min} (F' \cup \{i\}, \mathcal{F}) \text{ wrt } X \}$
end while

4 Experiments

In the following experiments, we apply the suggested algorithms to Pareto set approximations, generated by a MOEA, in order to investigate (i) whether the proposed dimensionality reduction method yields noticeable smaller sets of objectives, (ii) how the greedy algorithms perform, compared to the exact counterparts, and (iii) how our approach compares to the method proposed by Deb and Saxena. The experimental results indicate that our method is not only useful to analyze the output of MOEAs but also qualified for using it within an evolutionary algorithm. The Pareto set approximations, used in the experiments, are generated with the IBEA algorithm [14] on a linux computer (SunFireV60x with 3060 MHz).

Are all objectives necessary? This issue has been studied for 9 different 0-1-knapsack problem instances [15] and 3 instances for three different continuous test problems, namely DTLZ2, DTLZ5, and DTLZ7 [7]. The populations of the indicator-based algorithm IBEA after 100 generations were used as inputs for the greedy algorithms on the δ-`MOSS` and the `k-EMOSS` problem. The population size was increased for higher dimensional problems (5 objectives/ 100 solutions, 15 objectives/ 200 solutions, 25 objectives/ 300 solutions), where the other parameters of IBEA were chosen according to the standard settings in the PISA package [1]. To compare the 18 instances with their different numbers of objectives and their different ranges of objective values, we choose the δ values in percent of the population's spread[3] and the `k` values in percent of the instance's objective number k.

The results in Table 1 show for all instances that an objective reduction is possible without changing the dominance structure between the solutions, except for the

[3] We define the maximal spread S of a population P as the maximal difference of the solutions' objective values: $S = \max_{f_i \in \mathcal{F}} \max_{\mathbf{x},\mathbf{y} \in P} \{|f_i(\mathbf{x}) - f_i(\mathbf{y})|\}$.

	δ-MOSS				k-EMOSS		
	0%	10%	20%	40%	30%	60%	90%
knapsack: 100 items, 5 objectives, 100 solutions	5	5	5	5	0.926	0.516	0.486
knapsack, 100 items, 15 objectives, 200 solutions	11	10	10	9	0.818	0.348	0.000
knapsack, 100 items, 25 objectives, 300 solutions	13	13	13	11	0.597	0.000	0.000
knapsack: 250 items, 5 objectives, 100 solutions	5	5	5	4	0.859	0.697	0.280
knapsack, 250 items, 15 objectives, 200 solutions	11	11	10	9	0.762	0.342	0.000
knapsack, 250 items, 25 objectives, 300 solutions	12	12	12	11	0.575	0.000	0.000
knapsack: 500 items, 5 objectives, 100 solutions	5	5	5	4	0.748	0.504	0.237
knapsack, 500 items, 15 objectives, 200 solutions	15	15	14	10	0.643	0.435	0.278
knapsack, 500 items, 25 objectives, 300 solutions	25	23	17	13	0.472	0.320	0.138
DTLZ2: 5 objectives, 100 solutions	5	5	5	5	0.991	0.970	0.920
DTLZ2: 15 objectives, 200 solutions	13	13	13	13	0.942	0.891	0.000
DTLZ2: 25 objectives, 300 solutions	18	18	18	18	0.832	0.782	0.000
DTLZ5: 5 objectives, 100 solutions	5	5	5	5	0.952	0.906	0.896
DTLZ5: 15 objectives, 200 solutions	11	11	11	11	0.860	0.803	0.000
DTLZ5: 25 objectives, 300 solutions	13	13	13	13	0.820	0.000	0.000
DTLZ7: 5 objectives, 100 solutions	5	5	1	1	0.135	0.134	0.132
DTLZ7: 15 objectives, 200 solutions	10	1	1	1	0.078	0.070	0.000
DTLZ7: 25 objectives, 300 solutions	11	1	1	1	0.050	0.000	0.000

Table 1. Sizes (for δ-MOSS) and relative errors (for k-EMOSS) of objective subsets for different problems, computed with the greedy algorithms. For δ-MOSS, the δ value is chosen relatively to the maximum spread of the IBEA population after 100 generations; in the case of k-EMOSS the specified size k of the output subset is noted relatively to the problem's number of objectives.

5-objective-instances and the knapsack instances with 500 items. It turns out that the number of omissible objectives becomes the greater, the more objectives an instance posesses. If we allow changes of the dominance structure within the dimensionality reduction, further objectives can be omitted. However, the influence of a greater error on the resulting objective set size depends significantly on the problems. For example, only small errors yield fundamentally smaller objective sets for the DTLZ7 instances, while even a large error produces no further reduction for all DTLZ2 and DTLZ5 instances. Similar results for the δ-MOSS problem apply for another study, regarding the dominance structure on the whole search space for a small knapsack instance, cf. Fig. 3 and the next paragraph. By examining the k-EMOSS problem for the 18 instances in Table 1, we see similar results in a different manner. The smaller the chosen size of the resulting objective sets, the larger the error in the corresponding dominance structure.

Does the exact algorithm outperform the greedy one? Fig. 3 shows both the resulting objective set sizes and the running times for the exact and the greedy algorithm on the δ-MOSS problem for the 0-1-knapsack problem with four different numbers of objectives and 7 items. The small number of items allows the examination of the whole search space instead of a Pareto set approximation. We performed the dimensionality reduction for four different objective numbers k, five different δ values, and five independant instances for each k-δ combination. For all four choices of the objective set size and all

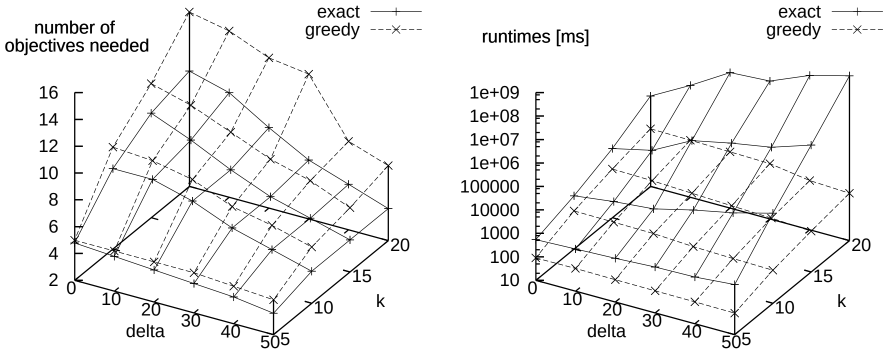


Fig. 3. Analysis of the whole search space for the knapsack problem with 7 items and comparison between the exact algorithm (solid lines) and the greedy algorithm (dashed lines). The sizes of the computed objective subsets are shown in the left plot and the running times of the two algorithms in the right one. Each data point is the average of five independent knapsack instances.

allowed errors δ, the exact algorithm yields smaller objective subsets than the greedy algorithm, while the running times, however, are considerably smaller for the greedy algorithm. Note in this context, that Fig. 3 shows a log scale plot for the running times. Also note, that the running time of the greedy algorithm decreases with higher δ which is not self-evident but significant, e.g., in a Wilcoxon rank sum test. Altogether, the results confirm the above observation, that more objectives can be omitted, the more error is allowed. This effect strengthens with instances of higher dimension.

Is our method comparable to the dimensionality reduction method by Deb and Saxena? Last, we compare our approach to the method of Deb and Saxena [6] on `k-EMOSS` for a knapsack instance with 20 objectives. We apply both methods on a Pareto front approximation for a knapsack instance with 100 items and 20 objectives, generated with an IBEA run (100 generations, population size 50). Deb and Saxena's approach is implemented according to [6]. Because the principal-component-analysis-based objective reduction method of Deb and Saxena cannot handle the `k-EMOSS` problem directly, we choose different threshold cuts (TC) such that all possible sizes of objective subsets are computed, where the TC determines the number of examined eigenvectors. Because an additional eigenvector causes either 0, 1, or 2 additional objectives in the objective subset, objective subsets with 1, 5, 6, and 10 objectives cannot be generated by the method for the considered knapsack instance. Note, that Deb and Saxena's method also performs an additive reduction of objectives using a reduced correlation matrix. Nevertheless, the method does not necessary yield, in general, δ-minimal sets, similar to our greedy algorithm.

Table 2 shows the computed objective subsets together with the absolute and relative[4] δ failures for the objective subsets computed with the method of Deb and Saxena, the exact and the greedy algorithm. In addition, Table 2 presents the used TC vales for

[4] The relative failure δ_{rel} is the absolute failure δ_{abs} divided by the spread of the IBEA population.

the method of Deb and Saxena and Fig. 4 provides parallel coordinates plots for the computed sets, 0-nonconflicting with the set of all objectives. With more objectives, the δ failure gets smaller for all methods. Although the exact algorithm shows, that only 7 objectives are necessary to yield no failure, the other two approaches perform noticeable reductions of objectives. But since Deb and Saxena's method is not especially developed for k-EMOSS, the resulting objective sets causes larger errors in the dominance structure than the corresponding sets, computed with the greedy algorithm. Note, that the method of Deb and Saxena yields a 0-nonconflicting subset of size 11 if one chooses the proposed TC of 95% [6].

5 Conclusions

In this paper we covered the problem of objective reduction in multiobjective optimization. We presented a necessary and sufficient condition for the possibility of an omission of objectives with a small change in the dominance structure. Besides that, we defined a measure of the dominance structure's variation when omitting a certain objective set and gave a general notion of conflicts between objective sets. We introduced the problem of finding a minimum objective subset, maintaining the given dominance structure with a given error and the problem of finding an objective subset with given size, changing the dominance structure least. In addition, we proposed an exact algorithm and fast heuristics for both problems. The capability of this objective reduction method was shown in experiments for outcomes of an MOEA on four different test problems and in comparison with a recently published dimensionality reduction approach.

The presented approach is useful for reducing the number of objectives *after* a MOEA run to simplify the decision maker's process, and we are currently working on the adequate integration of the presented dimensionality reduction method into an existing MOEA to reduce the number of objectives adaptively *during* an EA run.

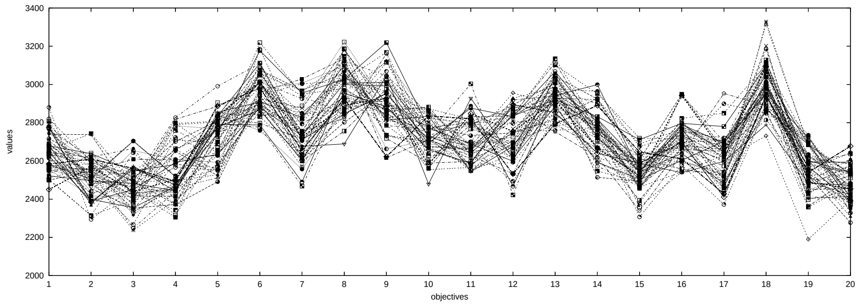
References

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. Dimo Brockhoff and Eckart Zitzler. On Objective Conflicts and Objective Reduction in Multiple Criteria Optimization. TIK Report 243, Institut für Technische Informatik und Kommunikationsnetze, ETH Zürich, February 2006. submitted to ICALP.
3. Carlos A. Coello Coello and Arturo Hernández Aguirre. Design of Combinational Logic Circuits through an Evolutionary Multiobjective Optimization Approach. *Artificial Intelligence for Engineering, Design, Analysis and Manufacture*, 16(1):39–53, January 2002.
4. Carlos A. Coello Coello, David A. Van Veldhuizen, and Gary B. Lamont. *Evolutionary Algorithms for Solving Multi-Objective Problems*. Kluwer Academic Publishers, New York, 2002.
5. Kalyanmoy Deb. *Multi-objective optimization using evolutionary algorithms*. Wiley, Chichester, UK, 2001.
6. Kalyanmoy Deb and Dhish Kumar Saxena. On finding pareto-optimal solutions through dimensionality reduction for certain large-dimensional multi-objective optimization problems.

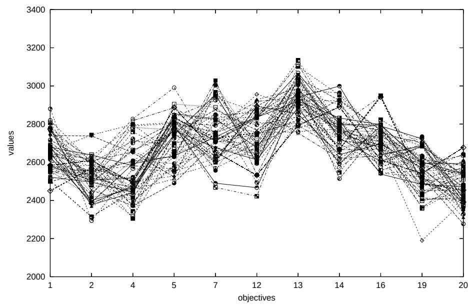
Technical Report KanGAL Report No. 2005011, Kanpur Genetic Algorithms Laboratory, December 2005.
7. Kalyanmoy Deb, Lothar Thiele, Marco Laumanns, and Eckart Zitzler. Scalable test problems for evolutionary multi-objective optimization. In A. Abraham, R. Jain, and R. Goldberg, editors, *Evolutionary Multiobjective Optimization: Theoretical Advances and Applications*, chapter 6, pages 105–145. Springer, 2005.
8. Michael Emmerich, Nicola Beume, and Boris Naujoks. An emo algorithm using the hypervolume measure as selection criterion. In *EMO 2005 Proceedings*, pages 62–76. Springer, 2005.
9. 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.
10. Ben Paechter, R. C. Rankin, Andrew Cumming, and Terence C. Fogarty. Timetabling the classes of an entire university with an evolutionary algorithm. In Agoston E. Eiben, Thomas Bäck, Marc Schoenauer, and Hans-Paul Schwefel, editors, *Parallel Problem Solving from Nature – PPSN V*, pages 865–874, Berlin, 1998. Springer. Lecture Notes in Computer Science 1498.
11. 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.
12. Kay Chen Tan, Eik Fun Khor, and Tong Heng Lee. *Multiobjective Evolutionary Algorithms and Applications*. Springer, London, 2005.
13. Lyndon While. A new analysis of the lebmeasure algorithm for calculating hypervolume. In *EMO 2005 Proceedings*, pages 326–340. Springer, 2005.
14. Eckart Zitzler and Simon Künzli. Indicator-based selection in multiobjective search. In *Proc. 8th International Conference on Parallel Problem Solving from Nature (PPSN VIII)*, volume 3242 of *Lecture Notes in Computer Science*, pages 832–842, Heidelberg, Germany, September 2004. Springer.
15. 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.
16. 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.

# obj	PCA-based				k-EMOSS exact			k-EMOSS greedy		
	TC	δ_{abs}	δ_{rel}	objective set	δ_{abs}	δ_{rel}	objective set	δ_{abs}	δ_{rel}	objective set
1	-	-	-	-	552	0.9154	18	552	0.9154	18
2	0.0000-0.5410	603	1.0000	4,14*	485	0.8043	8,9	508	0.8425	6,18
3	0.5411-0.6704	546	0.9055	4,7,14	447	0.7413	6,12,15	462	0.7662	6,9,18
4	0.6705-0.7702	546	0.9055	4,14,16,19	363	0.6020	7,8,9,11	418	0.6932	6,9,14,18
5	-	-	-	-	289	0.4793	3,4,8,9,20	369	0.6119	4,6,9,14,18
6	-	-	-	-	129	0.2139	3,4,5,8,9,18	356	0.5904	2,4,6,9,14,18
7	0.7703-0.8442	466	0.7728	2,4,7,12,14,16,19	0	0.0000	1,5,8,11,15,17,20	324	0.5373	2,4,6,9,13,14,18
8	0.8443-0.9235	466	0.7728	2,4,5,7,12,14,16,19	0	0.0000	1,5,8,11,15,17,20	287	0.4760	2,4,6,8,9,13,14,18
9	0.9236-0.9472	357	0.5920	1,2,4,5,7,12,14,16,19	0	0.0000	1,5,8,11,15,17,20	0	0.0000	2,3,4,6,8,9,13,14,18
≥ 11	$\geq$ 0.9473	0	0.0000	1,2,4,5,7,12,13,14,16,19,20	0	0.0000	1,5,8,11,15,17,20	0	0.0000	2,3,4,6,8,9,13,14,18

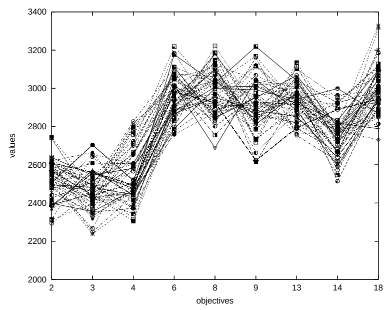
Table 2. Comparison between the PCA-based approach of Deb and Saxena [6] with the exact and greedy algorithm for k-EMOSS on a Pareto front approximation of a knapsack instance with 20 objectives. *Note, that for $0.3983 \leq TC \leq 0.5410$, the original set is $4, 7, 14$, but the final reduction using the reduced correlation matrix omits objective 7.



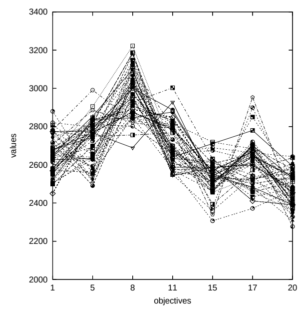
(a) original problem formulation with 20 objectives



(b) output of PCA-based approach



(c) output of greedy algorithm



(d) output of exact algorithm

Fig. 4. Visualization of the results from Table 2. The plots show the objective values for the 50 solutions computed by an IBEA run on a knapsack instance with 20 objectives. Figure (a) shows the values for the complete set of 20 objectives. The other figures show the objective subsets, 0-nonconflicting with the whole objective set, computed by the approach of Deb and Saxena (b), the greedy algorithm (c), and the exact algorithm (d). Note, that instead of the real objective values, the negative values $-f(\mathbf{x})$ are shown.

A Proofs omitted in Section 2

Theorem 1. *Let $\mathcal{F}' \subseteq \mathcal{F}$. Then $\mathcal{F}'$ is δ-nonconflicting with $\mathcal{F}$ if and only if $\preceq_{\mathcal{F}'} \subseteq \preceq^{\delta}_{\mathcal{F}}$.*

Proof. Let $\mathcal{F}' \subseteq \mathcal{F}$. Then for all $\delta \geq 0$ $\preceq_{\mathcal{F}} \subseteq \preceq_{\mathcal{F}'} \subseteq \preceq^{\delta}_{\mathcal{F}'}$, because $\forall i \in \mathcal{F} : \mathbf{x} \preceq_i \mathbf{y} \Rightarrow \forall i \in \mathcal{F}' \subseteq \mathcal{F} : \mathbf{x} \preceq_i \mathbf{y} \Rightarrow \forall i \in \mathcal{F}' : f_i(\mathbf{x}) \leq f_i(\mathbf{y}) \Rightarrow \forall i \in \mathcal{F}' : f_i(\mathbf{x}) - \delta \leq f_i(\mathbf{y}) \Rightarrow \forall i \in \mathcal{F}' : \mathbf{x} \preceq^{\delta}_i \mathbf{y}$ for all $x, y \in X$ and $\delta > 0$. But then $\mathcal{F}'$ δ-nonconflicting with $\mathcal{F}$ $\iff \mathcal{F}' \sqsubseteq^{\delta} \mathcal{F} \wedge \mathcal{F} \sqsubseteq^{\delta} \mathcal{F}' \iff \preceq_{\mathcal{F}'} \subseteq \preceq^{\delta}_{\mathcal{F}} \wedge \preceq_{\mathcal{F}} \subseteq \preceq^{\delta}_{\mathcal{F}'} \iff \preceq_{\mathcal{F}'} \subseteq \preceq^{\delta}_{\mathcal{F}}$. □

Theorem 2. *Let $\mathcal{F}_1, \mathcal{F}_2$ two objective sets and X a decision space. If*

$$\delta' := \max_{\substack{\mathbf{x},\mathbf{y} \in X \wedge \mathbf{x} \preceq_{\mathcal{F}_1} \mathbf{y} \\ i \in \mathcal{F}_2}} \{f_i(\mathbf{x}) - f_i(\mathbf{y})\} \text{ and } \delta'' := \max_{\substack{\mathbf{x},\mathbf{y} \in X \wedge \mathbf{x} \preceq_{\mathcal{F}_2} \mathbf{y} \\ i \in \mathcal{F}_1}} \{f_i(\mathbf{x}) - f_i(\mathbf{y})\},$$

then, $\mathcal{F}_1$ is $\overline{\delta}$-nonconflicting with $\mathcal{F}_2$ wrt X for all $\overline{\delta} \geq \max(\delta', \delta'')$ and no $\underline{\delta} < \max\{\delta', \delta''\}$ exists such that $\mathcal{F}_1$ is $\underline{\delta}$-nonconflicting with $\mathcal{F}_2$.

Proof. Let $\delta', \delta'' \in \mathbb{R}$ as defined above. Then

$$\begin{aligned}
&\forall \mathbf{x}, \mathbf{y} \in X : [\mathbf{x} \preceq_{\mathcal{F}_1} \mathbf{y} \Rightarrow \forall i \in \mathcal{F}_2 : f_i(\mathbf{x}) - f_i(\mathbf{y}) \leq \delta'] \\
&\qquad \wedge [\mathbf{x} \preceq_{\mathcal{F}_2} \mathbf{y} \Rightarrow \forall i \in \mathcal{F}_1 : f_i(\mathbf{x}) - f_i(\mathbf{y}) \leq \delta''] \\
&\iff \forall \mathbf{x}, \mathbf{y} \in X : [\mathbf{x} \preceq_{\mathcal{F}_1} \mathbf{y} \Rightarrow \forall i \in \mathcal{F}_2 : f_i(\mathbf{x}) - \delta' \leq f_i(\mathbf{y})] \\
&\qquad \wedge [\mathbf{x} \preceq_{\mathcal{F}_2} \mathbf{y} \Rightarrow \forall i \in \mathcal{F}_1 : f_i(\mathbf{x}) - \delta'' \leq f_i(\mathbf{y})] \\
&\overset{(*)}{\Longrightarrow} \forall \overline{\delta} \geq \max\{\delta', \delta''\} : \forall \mathbf{x}, \mathbf{y} \in X : \left[\mathbf{x} \preceq_{\mathcal{F}_1} \mathbf{y} \Rightarrow \forall i \in \mathcal{F}_2 : f_i(\mathbf{x}) - \overline{\delta} \leq f_i(\mathbf{y})\right] \\
&\qquad \wedge \left[\mathbf{x} \preceq_{\mathcal{F}_2} \mathbf{y} \Rightarrow \forall i \in \mathcal{F}_1 : f_i(\mathbf{x}) - \overline{\delta} \leq f_i(\mathbf{y})\right] \\
&\iff \forall \overline{\delta} \geq \max\{\delta', \delta''\} : \forall \mathbf{x}, \mathbf{y} \in X : [\mathbf{x} \preceq_{\mathcal{F}_1} \mathbf{y} \Rightarrow \mathbf{x} \preceq^{\overline{\delta}}_{\mathcal{F}_2} \mathbf{y}] \wedge [\mathbf{x} \preceq_{\mathcal{F}_2} \mathbf{y} \Rightarrow \mathbf{x} \preceq^{\overline{\delta}}_{\mathcal{F}_1} \mathbf{y}] \\
&\iff \forall \overline{\delta} \geq \max\{\delta', \delta''\} : \preceq_{\mathcal{F}_1} \subseteq \preceq^{\overline{\delta}}_{\mathcal{F}_2} \wedge \preceq_{\mathcal{F}_2} \subseteq \preceq^{\overline{\delta}}_{\mathcal{F}_1} \\
&\iff \forall \overline{\delta} \geq \max\{\delta', \delta''\} : \mathcal{F}_1 \sqsubseteq^{\overline{\delta}} \mathcal{F}_2 \wedge \mathcal{F}_2 \sqsubseteq^{\overline{\delta}} \mathcal{F}_1 \\
&\iff \mathcal{F}_1 \ \overline{\delta}\text{-nonconflicting with } \mathcal{F}_2 \text{ for all } \overline{\delta} \geq \max\{\delta', \delta''\}
\end{aligned}$$

As a result of implication $(*)$, it is clear that $\mathcal{F}_1$ is either weakly $\underline{\delta}$-conflicting or strongly $\underline{\delta}$-conflicting with $\mathcal{F}_2$ for any $\underline{\delta} < \max\{\delta', \delta''\}$ if δ' and δ'' are defined as above. □

B Correctness proofs

In this section we provide the correctness proofs for the algorithms proposed in Sec. 3.2.

B.1 Greedy Algorithm for δ-MOSS

Before proving the correctness of Algorithm 2, we prove the next useful Lemma.

Lemma 1. *Let $\mathcal{F}' \subseteq \mathcal{F}$ and $\delta > 0$. Then*

$$\left(\forall \mathbf{x}, \mathbf{y} \in X : \mathbf{x} \preceq_{\mathcal{F}'} \mathbf{y} \Longleftrightarrow \mathbf{x} \preceq^{0,\delta}_{\mathcal{F}',\mathcal{F}\setminus\mathcal{F}'} \mathbf{y}\right) \Longrightarrow \mathcal{F}' \text{ is } \delta\text{-nonconflicting with } \mathcal{F}.$$

Proof. Let $\mathcal{F}' \subseteq \mathcal{F}$ and $A := \left(\forall \mathbf{x}, \mathbf{y} \in X : \mathbf{x} \preceq_{\mathcal{F}'} \mathbf{y} \Longleftrightarrow \mathbf{x} \preceq^{0,\delta}_{\mathcal{F}',\mathcal{F}\setminus\mathcal{F}'} y\right)$. Then $\preceq_{\mathcal{F}'} = \preceq^{0}_{\mathcal{F}'} \overset{A}{=} \preceq^{0,\delta}_{\mathcal{F}',\mathcal{F}\setminus\mathcal{F}'} = (\preceq^{0}_{\mathcal{F}'} \cap \preceq^{\delta}_{\mathcal{F}\setminus\mathcal{F}'}) \subseteq \preceq^{\delta}_{\mathcal{F}'} \cap \preceq^{\delta}_{\mathcal{F}\setminus\mathcal{F}'} = \preceq^{\delta}_{\mathcal{F}}$, i.e., $\mathcal{F}'$ is δ-nonconflicting with $\mathcal{F}$ according to Theorem 1. □

Theorem 3. *Given the objective vectors $f(\mathbf{x}_1), \ldots, f(\mathbf{x}_m) \in \mathbb{R}^k$ and a $\delta \in \mathbb{R}$, Algorithm 2 always provides an objective subset $\mathcal{F}' \subseteq \mathcal{F}$, δ-nonconflicting with $\mathcal{F} := \{1, \ldots, k\}$ in time $O(\min\{k^3 \cdot m^2, k^2 \cdot m^4\})$.*

Proof. If we show that the invariant

$$\forall (\mathbf{x}, \mathbf{y}) \in \overline{R} := (X \times X) \setminus R : \quad \mathbf{x} \preceq_{\mathcal{F}'} \mathbf{y} \Longleftrightarrow \mathbf{x} \preceq^{0,\delta}_{\mathcal{F}',\mathcal{F}} \mathbf{y} \tag{I}$$

holds during each step of Algorithm 2, the theorem is proved, due to Lemma 1 and the fact that $\mathbf{x} \preceq_{\mathcal{F}'} \mathbf{y} \Longleftrightarrow \mathbf{x} \preceq^{0,\delta}_{\mathcal{F}',\mathcal{F}\setminus\mathcal{F}'} \mathbf{y}$ holds for all $(\mathbf{x}, \mathbf{y}) \in X \times X$ if Algorithm 2 terminates, i.e., if $R = \emptyset$. We proof the invariant with induction over $|\overline{R}|$.

Induction basis: When the algorithm starts, $R = X \times X \setminus \preceq_{\mathcal{F}}$, i.e., $\overline{R} = \preceq_{\mathcal{F}}$. For each $(\mathbf{x}, \mathbf{y}) \in \overline{R} = \preceq_{\mathcal{F}}$ with $\mathbf{x} \preceq_{\mathcal{F}'} \mathbf{y}$, i.e., $\mathbf{x} \preceq_{\emptyset} \mathbf{y}$ with $\preceq_{\emptyset} = X \times X$, $\mathbf{x} \preceq_{\mathcal{F}} \mathbf{y}$ holds and therefore $\mathbf{x} \preceq^{0,\delta}_{\mathcal{F}',\mathcal{F}\setminus\mathcal{F}'} \mathbf{y}$. The other direction $\mathbf{x} \preceq^{0,\delta}_{\mathcal{F}',\mathcal{F}\setminus\mathcal{F}'} \mathbf{y} \Rightarrow \mathbf{x} \preceq_{\mathcal{F}'} \mathbf{y}$ always holds trivially. Thus, the invariant is correct for the smallest possible $|\overline{R}|$, after the initialization of the algorithm.

Induction step: Now let $|\mathcal{F}'| > 0$. Then, the invariant can only become false, if we change R (and with it $\overline{R}$) in line 7 of Algorithm 2. Note, first, that R becomes only smaller by-and-by, i.e., $\overline{R}$ contains more and more pairs $(\mathbf{x}, \mathbf{y}) \in X \times X$. Such a pair $(\mathbf{x}, \mathbf{y})$, already contained in $\overline{R}$, stays in $\overline{R}$ forever and fulfills the implication in the invariant (I) for every $\mathcal{F}'' \supseteq \mathcal{F}'$ if the pair fulfills it for at least one $\mathcal{F}' \subseteq \mathcal{F}$. If an $\{i\}$ is inserted in $\mathcal{F}'$ to gain $\mathcal{F}'' \supseteq \mathcal{F}'$, two possibilities for a pair $(\mathbf{x}, \mathbf{y}) \in \overline{R}$ exist. First, if $\mathbf{x} \not\preceq_{\mathcal{F}'} \mathbf{y}$, then $\mathbf{x} \not\preceq_{\mathcal{F}''} \mathbf{y}$ for any $\mathcal{F}'' \supseteq \mathcal{F}'$ and also $\mathbf{x} \not\preceq^{0,\delta}_{\mathcal{F}'',\mathcal{F}\setminus\mathcal{F}''} \mathbf{y}$. Second, if $\mathbf{x} \preceq_{\mathcal{F}'} \mathbf{y}$, then $\mathbf{x} \preceq^{0,\delta}_{\mathcal{F}',\mathcal{F}\setminus\mathcal{F}'} \mathbf{y}$ by induction hypothesis. Thus, $\mathbf{x} \preceq^{\delta}_{\mathcal{F}\setminus\mathcal{F}'} \mathbf{y}$ and $\mathbf{x} \preceq^{\delta}_{\mathcal{F}\setminus\mathcal{F}''} \mathbf{y}$ for any $\mathcal{F}'' \supseteq \mathcal{F}'$. If $\mathbf{x} \preceq_{\mathcal{F}''} \mathbf{y}$ for any $\mathcal{F}'' \supseteq \mathcal{F}'$, then $\mathbf{x} \preceq^{0,\delta}_{\mathcal{F}'',\mathcal{F}\setminus\mathcal{F}''} \mathbf{y}$ and if $\mathbf{x} \not\preceq_{\mathcal{F}'} \mathbf{y}$ for any $\mathcal{F}'' \subseteq \mathcal{F}'$ then $\mathbf{x} \not\preceq^{0,\delta}_{\mathcal{F}'',\mathcal{F}\setminus\mathcal{F}''} \mathbf{y}$. Thus, a pair $(\mathbf{x}, \mathbf{y}) \in \overline{R}$ will always fulfill the implication in (I) for any $\mathcal{F}'' \supseteq \mathcal{F}'$ if it fulfills it for $\mathcal{F}'$. Beyond, a pair $(\mathbf{x}, \mathbf{y}) \in X \times X$ will only be included in $\overline{R}$ during the update of R in line 7 if

(i) $(\mathbf{x}, \mathbf{y}) \notin (R \cap \preceq_{i^*})$ or if
(ii) $(\mathbf{x}, \mathbf{y}) \in \preceq^{0,\delta}_{\mathcal{F}' \cup \{i^*\}, \mathcal{F} \setminus (\mathcal{F}' \cup \{i^*\})}$

In case (i), the invariant stays true because for all new pairs $(\mathbf{x}, \mathbf{y})$ in $\overline{R}$, $(\mathbf{x}, \mathbf{y}) \in R \wedge (\mathbf{x}, \mathbf{y}) \notin \preceq_{i^*}$ holds. Thus, $(\mathbf{x}, \mathbf{y}) \notin \cap_{i \in (\mathcal{F}' \cup \{^*\})} \preceq_i = \preceq_{\mathcal{F}'}$ and, therefore, $(\mathbf{x}, \mathbf{y}) \notin \preceq^{0,\delta}_{\mathcal{F}' \cup \{i^*\}, \mathcal{F} \setminus (\mathcal{F}' \cup \{i^*\})}$ as well. In the case (ii), $(\mathbf{x}, \mathbf{y}) \in \preceq^{0,\delta}_{\mathcal{F}' \cup \{i^*\}, \mathcal{F} \setminus (\mathcal{F}' \cup \{i^*\})}$ and trivially $(\mathbf{x}, \mathbf{y}) \in \preceq_{\mathcal{F}' \cup \{i^*\}}$, i.e., the invariant remains true, too.

The running time of Algorithm 2 results mainly from the computation of the relations in line 6. The initialization needs time $O(k \cdot m^2)$ altogether. As the relation $\preceq^{0,\delta}_{\mathcal{F}' \cup \{i^*\}, \mathcal{F} \setminus (\mathcal{F}' \cup \{i^*\})}$ is known from line 6, the calculation of the new R in line 7 needs time $O(m^2)$; line 8 needs only constant time. The computation of the relations $\preceq^{0,\delta}_{\mathcal{F}' \cup \{i\}, \mathcal{F} \setminus (\mathcal{F}' \cup \{i\})}$ in line 6 needs time $O(k \cdot m^2)$ for each i, thus, line 6 needs time $O(k^2 \cdot m^2)$ altogether. Hence, the computation time for each while loop cycle lasts time $O(k^2 \cdot m^2)$. Because in each loop cycle, $|\mathcal{F}'|$ increases by one, there are at most k cycles before Algorithm 2 terminates. On the other hand, Algorithm 2 terminates if $R = \emptyset$, i.e., after at most $|X \times X| = O(m^2)$ cycles of the while loop, if in each cycle $|R|$ decreases by at least one—what is true due to Theorem 3. The total running time of Algorithm 2 is, therefore, $O(\max\{k, m^2\} \cdot k^2 \cdot m^2) = O(\max\{k^3 \cdot m^2, k^2 \cdot m^4\})$. $\square$

B.2 Exact Algorithm

Theorem 4. *Algorithm 1 solves both the δ-*`MOSS` *and the* `k-EMOSS` *problem exactly in time $O(m^2 \cdot k \cdot 2^k)$.*

Proof. To prove the correctness of Algorithm 1, we use Lemma 2. It states that Algorithm 1 computes for each considered set M of solution pairs a set of pairs $(\mathcal{F}', \delta')$ of an objective subset $\mathcal{F}' \subseteq \mathcal{F}$ with the corresponding correct δ' value (i, ii) that are minimal (iii, iv). Moreover, the algorithm computes solely minimal pairs (v, vi). With Lemma 2, the correctness of Algorithm 2 follows directly from the lines 12 and 13.

The upper bound on the running time of Algorithm 1 results from the size of the set S_M. For all of the $O(m^2)$ solution pairs, the set $S_{\{(\mathbf{x},\mathbf{y})\}}$ can be computed in time $O(k^2) = o(k \cdot 2^k)$, but the computation time for $S_M \sqcup S_{\{(\mathbf{x},\mathbf{y})\}}$ can be exponential in k. As S_M contains at most $O(2^k)$ objective subsets of size $O(k)$, the computation of $S_M \sqcup S_{\{(\mathbf{x},\mathbf{y})\}}$ in line 9 is possible in time $O(k \cdot 2^k)$ and, therefore, the whole algorithm runs in time $O(m^2 \cdot k \cdot 2^k)$. $\square$

For the following Lemma, we need a new short notation for δ failures regarding a set M of solution pairs.

Definition 8 *Let $\mathcal{F}' \subseteq \mathcal{F}$ and $M \subseteq X \times X$. Then $\delta(\mathcal{F}', M) := \delta_{\min}(\mathcal{F}', \mathcal{F})$ wrt all solution pairs $(\mathbf{x}, \mathbf{y}) \in M$.*

Lemma 2. *Given an instance of the δ-*`MOSS` *or the* `k-EMOSS` *problem. Let $\mathcal{F}' \subseteq \mathcal{F}$, $\mathcal{F}' \neq \emptyset$, an arbitrary objective set and*

$$M := \{(\mathbf{x}, \mathbf{y}) \in X \times X \mid (\mathbf{x}, \mathbf{y}) \textit{ considered in Algorithm 1 so far}\}.$$

Then there exists always a $(\mathcal{F}'' \subseteq \mathcal{F}', \delta'') \in S_M$, such that the following six statements hold.

(i) $\delta(\mathcal{F}'', M) = \delta''$
(ii) $\delta(\mathcal{F}', M) = \delta''$
(iii) $\nexists(\mathcal{F}''', \delta''') \in S_M : \mathcal{F}''' \subset \mathcal{F}' \wedge \delta''' \leq \delta''$
(iv) $\nexists(\mathcal{F}''', \delta''') \in S_M : \mathcal{F}''' \subseteq \mathcal{F}' \wedge \delta''' < \delta''$
(v) $\nexists(\mathcal{F}''', \delta''') \in S_M : \mathcal{F}''' \supset \mathcal{F}' \wedge \delta''' \geq \delta''$
(vi) $\nexists(\mathcal{F}''', \delta''') \in S_M : \mathcal{F}''' \supseteq \mathcal{F}' \wedge \delta''' > \delta''$

Proof. The statements (iii)-(vi) hold for any M due to the definition of the $\sqcup$-union in line 9. We, therefore, prove only (i) and (ii) by mathematical induction on $|M|$.

Induction basis: Let $|M| = 1$, i.e., $M := \{(\mathbf{x}, \mathbf{y})\}$.

(a) $\mathbf{x} \sim_{\mathcal{F}} \mathbf{y}$: Thus, $\forall i \in \mathcal{F} : f_i(\mathbf{x}) = f_i(\mathbf{y})$ and $\forall \mathcal{F}' \subseteq \mathcal{F}, \mathcal{F}' \neq \emptyset$: $\delta(\mathcal{F}', \{(\mathbf{x}, \mathbf{y})\}) = 0$. By definition of $\sqcup$, Algorithm 2 computes $S_{\{(\mathbf{x},\mathbf{y})\}} = \{(\{i\}, 0) \,|\, 1 \leq i \leq k\}$ correctly according to (i) and (ii).

(b) WLOG $\mathbf{x} \preceq_{\mathcal{F}} \mathbf{y} \wedge \neg(\mathbf{y} \preceq_{\mathcal{F}} \mathbf{x})$: We can divide $\mathcal{F}$ into two disjoint sets $\mathcal{F}_=, \mathcal{F}_<$ with $\mathcal{F}_= \cup \mathcal{F}_< = \mathcal{F}$, $\mathcal{F}_< \neq \emptyset$, $\forall i \in \mathcal{F}_= : \mathbf{x} \preceq_i \mathbf{y} \wedge \mathbf{y} \preceq_i \mathbf{x}$, and $\forall i \in \mathcal{F}_< : \mathbf{x} \preceq_i \mathbf{y} \wedge \neg\,(\mathbf{y} \preceq_i \mathbf{x})$, i.e., $\forall i \in \mathcal{F}_= : f_i(\mathbf{x}) = f_i(\mathbf{y})$ and $\forall i \in \mathcal{F}_< : f_i(\mathbf{x}) < f_i(\mathbf{y})$. Furthermore, $\forall i \in \mathcal{F}_< : \delta(\{i\}, \{(\mathbf{x}, \mathbf{y})\}) = 0$ and $\forall i \in \mathcal{F}_= : \delta(\{i\}, \{(\mathbf{x}, \mathbf{y})\}) = \delta > 0$ with $\delta := \max_{j \in \mathcal{F}_<}\{f_j(\mathbf{y}) - f_j(\mathbf{x})\}$ independent of the choice of i. Therefore, $S_{\{(\mathbf{x},\mathbf{y})\}}$ contains all pairs $(\{i\}, \delta_i)$ with $1 \leq i \leq k$ and $\delta_i := \begin{cases} 0 & \text{if } i \in \mathcal{F}_< \\ \delta & \text{if } i \in \mathcal{F}_= \end{cases}$. (i) and (ii) hold, because for any $\mathcal{F}' \subseteq \mathcal{F}$, $\mathcal{F}' \neq \emptyset$, $\delta' := \delta(\mathcal{F}', \{(\mathbf{x}, \mathbf{y})\})$ is either 0 or δ, depending on $\mathcal{F}' \subseteq \mathcal{F}_=$ ($\Rightarrow \delta' = \delta > 0$) or $\mathcal{F}' \not\subseteq \mathcal{F}_=$ ($\Rightarrow \delta' = 0$).

(c) $\mathbf{x} \parallel_{\mathcal{F}} \mathbf{y}$: We can divide $\mathcal{F}$ into three well-defined disjoint sets $\mathcal{F}_<$, $\mathcal{F}_>$, and $\mathcal{F}_=$ with $\mathcal{F}_< \cup \mathcal{F}_> \cup \mathcal{F}_= = \mathcal{F}$, $\mathcal{F}_< \neq \emptyset$, $\mathcal{F}_> \neq \emptyset$, $\forall i \in \mathcal{F}_< : f_i(\mathbf{x}) < f_i(\mathbf{y})$, $\forall i \in \mathcal{F}_> : f_i(\mathbf{x}) > f_i(\mathbf{y})$, and $\forall i \in \mathcal{F}_= : f_i(\mathbf{x}) = f_i(\mathbf{y})$. For all singletons $\{i\}$ with $1 \leq i \leq k$, $\delta_i := \delta(\{i\}, \{(\mathbf{x}, \mathbf{y})\}) > 0$ holds, i.e., $(\{i\}, \delta_i) \in S_{\{(\mathbf{x},\mathbf{y})\}}$ for all $i \in \mathcal{F}$ and

$$\delta_i := \begin{cases} \delta_< := \max_{j \in \mathcal{F}_>}\{f_j(\mathbf{x}) - f_j(\mathbf{y})\} & \text{if } i \in \mathcal{F}_< \\ \delta_> := \max_{j \in \mathcal{F}_<}\{f_j(\mathbf{y}) - f_j(\mathbf{x})\} & \text{if } i \in \mathcal{F}_> \\ \delta_= := \max_{j \in \mathcal{F} \setminus \{i\}}\{|f_j(\mathbf{x}) - f_j(\mathbf{y})|\} & \text{if } i \in \mathcal{F}_= \end{cases}.$$

In addition, $S_{\{(\mathbf{x},\mathbf{y})\}}$ contains only those pairs $(\{i, j\}, 0)$ with $i \in \mathcal{F}_< \wedge j \in \mathcal{F}_>$. Other pairs $(\{i, j\}, \delta)$ with $i \neq j \wedge \delta > 0$ are not in $S_{\{(\mathbf{x},\mathbf{y})\}}$ due to the $\sqcup$-union in line 7.

Now, let $\mathcal{F}' \subseteq \mathcal{F}$. Then $\mathcal{F}'_<, \mathcal{F}'_>, \mathcal{F}'_= \subseteq \mathcal{F}'$ can be defined similarly to $\mathcal{F}_>$, $\mathcal{F}_>$, and $\mathcal{F}_=$ for $\mathcal{F}$. The statement (i) holds due to the $\sqcup$-union and (ii) holds since $\delta(\mathcal{F}', \{(\mathbf{x}, \mathbf{y})\})$ can only take a value $\delta' \in \{0, \delta_<, \delta_>, \delta_=\}$ and a pair $(\mathcal{F}'' \subseteq, \delta')$ exists in $S_{\{(\mathbf{x},\mathbf{y})\}}$:

1. $\delta(\mathcal{F}', \{(\mathbf{x}, \mathbf{y})\}) = 0$ if $\mathcal{F}'_> \leq \emptyset \wedge \mathcal{F}'_< \leq \emptyset$. But then, $i \in \mathcal{F}'_>$ and $j \in \mathcal{F}'_<$ exist and $(\{i, j\}, 0) \in S_{\{(\mathbf{x},\mathbf{y})\}}$.
2. WLOG $\delta(\mathcal{F}', \{(\mathbf{x}, \mathbf{y})\}) = \delta_<$ if $\mathcal{F}'_> = \emptyset \wedge \mathcal{F}'_< \neq \emptyset$. Then there exists an $i \in \mathcal{F}'_<$ and $(\{i\}, \delta_<) \in S_{\{(\mathbf{x},\mathbf{y})\}}$

3. $\delta(\mathcal{F}', \{(\mathbf{x}, \mathbf{y})\}) = \delta_{=}$ if $\mathcal{F}'_{>} = \emptyset \wedge \mathcal{F}'_{<}\emptyset$. Then $\mathcal{F}' \subseteq \mathcal{F}_{=}$ and there exists at least one $i \in \mathcal{F}'_{=}$ such that $(\{i\}, \delta_{=}) \in S_{\{(\mathbf{x},\mathbf{y})\}}$.

Induction step: Let $\mathcal{F}' \subseteq \mathcal{F}$ an arbitrary objective set with $\delta(\mathcal{F}', M \cup \{(\mathbf{x}, \mathbf{y})\})$. Assume that (i)-(vi) holds for M and $\{\mathbf{x}, \mathbf{y}\}$. Thus, $\exists(\mathcal{F}''', \delta''') \in S_M$ with $\mathcal{F}''' \subseteq \mathcal{F}'$ and (i)-(vi) and $\exists(\mathcal{F}'''', \delta'''') \in S_{\{(\mathbf{x},\mathbf{y})\}}$ with $\mathcal{F}'''' \subseteq \mathcal{F}'$ and (i)-(vi).

To show that an $(\mathcal{F}'' \subseteq \mathcal{F}', \delta'')$ exists in $S_{M\cup\{(\mathbf{x},\mathbf{y})\}} := S_M \sqcup S_{\{(\mathbf{x},\mathbf{y})\}}$ that fulfills (i) and (ii), we define $\mathcal{F}'' := \mathcal{F}''' \cup \mathcal{F}'''' \subseteq \mathcal{F}'$ and $\delta'' := \max\{\delta''', \delta''''\}$. Because $\delta(\mathcal{F}''', M) = \delta(\mathcal{F}', M)$, $\delta(\mathcal{F}''', M) = \delta(\hat{\mathcal{F}}, M)$ holds for any $\mathcal{F}''' \subseteq \hat{\mathcal{F}} \subseteq \mathcal{F}'$ and because of $\delta(\mathcal{F}'''', \{(\mathbf{x}, \mathbf{y})\}) = \delta(\mathcal{F}', \{(\mathbf{x}, \mathbf{y})\})$, $\delta(\mathcal{F}''', \{(\mathbf{x}, \mathbf{y})\}) = \delta(\hat{\mathcal{F}}, \{(\mathbf{x}, \mathbf{y})\})$ holds for any $\mathcal{F}''' \subseteq \hat{\mathcal{F}} \subseteq \mathcal{F}'$. Together with $\mathcal{F}''' \cup \mathcal{F}'''' \subseteq \mathcal{F}'$, this yields $\delta(\mathcal{F}''' \cup \mathcal{F}'''', M) = \delta(\mathcal{F}', M) = \delta'''$ as well as $\delta(\mathcal{F}''' \cup \mathcal{F}'''', \{(\mathbf{x}, \mathbf{y})\}) = \delta(\mathcal{F}', M) = \delta''''$. This follows (i) and (ii):

$$
\begin{aligned}
\delta'' &= \max\{\delta(\mathcal{F}''' \cup \mathcal{F}'''', M), \delta(\mathcal{F}''' \cup \mathcal{F}'''', \{(\mathbf{x}, \mathbf{y})\})\} \\
&= \delta(\mathcal{F}''' \cup \mathcal{F}'''', M \cup \{(\mathbf{x}, \mathbf{y})\}) && \text{(i)} \\
&= \max\{\delta(\mathcal{F}', M), \delta(\mathcal{F}', \{(\mathbf{x}, \mathbf{y})\})\} = \delta(\mathcal{F}', M \cup \{(\mathbf{x}, \mathbf{y})\}) && \text{(ii)}
\end{aligned}
$$

□