

# Are All Objectives Necessary? On Dimensionality Reduction in Evolutionary Multiobjective Optimization

Dimo Brockhoff and Eckart Zitzler

Computer Engineering and Networks Laboratory, ETH Zurich, 8092 Zurich, Switzerland  
{brockhoff, zitzler}@tik.ee.ethz.ch

**Abstract.** Most of the available multiobjective evolutionary algorithms (MOEA) for approximating the Pareto set have been designed for and tested on low dimensional problems ( $\leq 3$  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 [1]; however, studies addressing high-dimensional problems are rare [2,3]. 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 [4]. 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 [5] have running times exponential in the number of objectives [6,7]. 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 [8,9,10], 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 [11] 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 [12], 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 [12],
- the introduction of a measure for variations of the dominance structure
- the definition of the problems  $\delta$ -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  $\delta$ -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 [11].

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

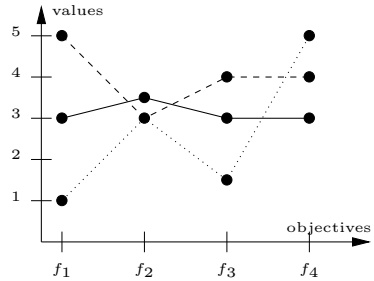
## 2 A Measure for Changes of the Dominance Structure

Without loss of generality, in this paper we consider a minimization problem with  $k$  objective functions  $f_i : X \rightarrow \mathbb{R}$ ,  $1 \leq i \leq k$ , where the vector function  $f := (f_1, \dots, f_k)$  maps each solution  $\mathbf{x} \in X$  to an objective vector  $f(\mathbf{x}) \in \mathbb{R}^k$ . 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  $\mathcal{F}' \subseteq \mathcal{F} := \{f_1, \dots, f_k\}$ . For better readability, we will sometimes only consider the objective functions’ indices, e.g.,  $\mathcal{F}' = \{1, 2, 3\}$  instead of  $\mathcal{F}' = \{f_1, f_2, f_3\}$ . We say  $\mathbf{x}$  *weakly dominates*  $\mathbf{y}$  wrt the objective set  $\mathcal{F}'$  ( $\mathbf{x} \preceq_{\mathcal{F}'} \mathbf{y}$ ) if  $(\mathbf{x}, \mathbf{y}) \in \preceq_{\mathcal{F}'}$ . Two solutions  $\mathbf{x}, \mathbf{y}$  are *incomparable* if neither weakly dominates the other one. A solution  $\mathbf{x}^* \in X$  is called *Pareto optimal* if there is no other  $\mathbf{x} \in X$  that dominates  $\mathbf{x}^*$  wrt the set of all objectives. The set of all Pareto optimal solutions is called *Pareto (optimal) set*, for which an approximation is sought.

In [12] we have proposed a method that computes for a given solution set  $A \subseteq X$  a minimum subset  $\mathcal{F}'$  of objectives with  $\mathcal{F}' \subseteq \mathcal{F} := \{f_1, \dots, f_k\}$  such that the dominance structure is preserved. In other words, for  $\mathcal{F}'$  holds that  $\mathbf{x} \preceq_{\mathcal{F}'} \mathbf{y} \iff \mathbf{x} \preceq_{\mathcal{F}} \mathbf{y}$  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. [9], 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 [12], 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 objective 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.

**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 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  $\delta$  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  $\delta$  values for all solution pairs  $\mathbf{x}, \mathbf{y}$ , we can then determine the maximum error. The meaning of the maximum  $\delta$  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  $\delta$ . For  $\mathcal{F}' := \{f_3, f_4\}$ , the maximum error is  $\delta = 0.5$ ; for  $\mathcal{F}' := \{f_2, f_4\}$ , the maximum  $\delta$  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)  $\varepsilon$ -dominance relation<sup>1</sup> [13] and a generalization of the notion of conflicts between objective sets, defined in [12].

**Definition 1.** Let  $\mathcal{F}_1$  and  $\mathcal{F}_2$  two objective sets. We define  $\mathcal{F}_1 \sqsubseteq^\delta \mathcal{F}_2 := \iff \preceq_{\mathcal{F}_1} \subseteq \preceq_{\mathcal{F}_2}^\delta$ .

**Definition 2.** Let  $\mathcal{F}_1$  and  $\mathcal{F}_2$  two objective sets. We call

- $\mathcal{F}_1$   $\delta$ -nonconflicting with  $\mathcal{F}_2$  iff  $(\mathcal{F}_1 \sqsubseteq^\delta \mathcal{F}_2) \wedge (\mathcal{F}_2 \sqsubseteq^\delta \mathcal{F}_1)$ ;
- $\mathcal{F}_1$   $\delta$ -conflicting with  $\mathcal{F}_2$  iff  $\neg(\mathcal{F}_1 \delta\text{-nonconflicting with } \mathcal{F}_2)$ .

<sup>1</sup>  $\preceq_{\mathcal{F}'}^\varepsilon := \{(\mathbf{x}, \mathbf{y}) \mid \mathbf{x}, \mathbf{y} \in X \wedge \forall i \in \mathcal{F}' \subseteq \mathcal{F} : f_i(\mathbf{x}) - \varepsilon \leq f_i(\mathbf{y})\}$ .

The above definition of  $\delta$ -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  $\delta$ -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  $\mathbf{x}$   $\varepsilon$ -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  $\mathcal{F}' \subseteq \mathcal{F}$ ,  $\delta$ -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  $\delta$ -nonconflicting with the set  $\mathcal{F}$  of all objectives,  $\mathbf{x}$   $\delta$ -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  $\delta$  in the omitted objectives.

Based on the above conflict definitions, we will now formalize the notion of  $\delta$ -minimal and  $\delta$ -minimum objective sets including the corresponding notion for  $\delta = 0$  in [12] and, furthermore, present a condition under which an objective reduction is possible.

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

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

A  $\delta$ -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  $\delta$ . A  $\delta$ -minimum objective set is the smallest possible set of original objectives that preserves the original dominance structure except for an error of  $\delta$ . By definition, every  $\delta$ -minimum objective set is  $\delta$ -minimal, but not all  $\delta$ -minimal sets are at the same time  $\delta$ -minimum.

**Definition 4.** A set  $\mathcal{F}$  of objectives is called  $\delta$ -redundant if and only if there exists  $\mathcal{F}' \subset \mathcal{F}$  that is  $\delta$ -minimal wrt  $\mathcal{F}$ .

This definition of  $\delta$ -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  $\delta$ -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  $\delta$ -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 $\delta$ -MOSS and $k$ -EMOSS Problems

Based on Sec. 2, the problem MINIMUM OBJECTIVE SUBSET (MOSS), proposed in [12], 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  $\delta$ .

**Definition 5.** *Given a multiobjective optimization problem, the problem  $\delta$ -MINIMUM OBJECTIVE SUBSET ( $\delta$ -MOSS) is defined as follows.*

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

*Task:* Compute a  $\delta$ -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  $\preceq_i, \preceq_{\mathcal{F}}$  are only given implicitly in a  $\delta$ -MOSS instance. Nevertheless,  $\delta$ -MOSS is a generalization of MOSS and therefore  $\mathcal{NP}$  hard, cf. the accompanying technical report [14]. As a variation of the  $\delta$ -MOSS problem, we introduce the problem of finding an objective subset of size  $\leq k$  with minimum error according to  $\mathcal{F}$ .

**Definition 6.** *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  $f(\mathbf{x}_1), \dots, f(\mathbf{x}_m) \in \mathbb{R}^k$  of the solutions  $\mathbf{x}_1, \dots, \mathbf{x}_m \in A \subseteq X$  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  $\delta$ -nonconflicting with  $\mathcal{F}$  with the minimal possible  $\delta$ .

### 3.2 Algorithms

**An Exact Algorithm.** Algorithm 1, as a generalization of the exact algorithm for the MOSS problem [12], solves both the  $\delta$ -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  $\delta'$  value such that  $\mathcal{F}'$  is  $\delta'$ -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  $\delta$  error for two solutions  $\mathbf{x}, \mathbf{y} \in X$ , such that  $\mathcal{F}_1$  is  $\delta$ -nonconflicting with  $\mathcal{F}_2$  wrt  $\mathbf{x}, \mathbf{y}$ . Due to space limitations, we cannot show here, how this minimal  $\delta$  can be computed in time  $O(k \cdot m^2)$  and refer to [14]. Furthermore, Algorithm 1 computes the union  $\sqcup$  of two sets of objective subsets with simultaneous deletion of not  $\delta'$ -minimal pairs  $(\mathcal{F}', \delta')$ :

---

**Algorithm 1.** An exact algorithm for  $\delta$ -MOSS and  $k$ -EMOSS.

---

```

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

```

---

**Algorithm 2.** A greedy algorithm for  $\delta$ -MOSS.

---

```

1: Init:
2: compute the relations  $\preceq_i$  for all  $1 \leq i \leq k$  and  $\preceq_{\mathcal{F}}$ 
3:  $\mathcal{F}' := \emptyset$ 
4:  $R := A \times A \setminus \preceq_{\mathcal{F}}$ 
5: while  $R \neq \emptyset$  do
6:    $i^* = \operatorname{argmin}_{i \in \mathcal{F} \setminus \mathcal{F}'} \{ |(R \cap \preceq_i) \setminus (\preceq_{\mathcal{F}' \cup \{i\}}^0 \cap \preceq_{\mathcal{F} \setminus (\mathcal{F}' \cup \{i\})}^\delta)| \}$ 
7:    $R := (R \cap \preceq_{i^*}) \setminus (\preceq_{\mathcal{F}' \cup \{i^*\}}^0 \cap \preceq_{\mathcal{F} \setminus (\mathcal{F}' \cup \{i^*\})}^\delta)$ 
8:    $\mathcal{F}' := \mathcal{F}' \cup \{i^*\}$ 
9: end while

```

---

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

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 also be found in [14]. 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  $\delta$ -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  $\delta$ -MOSS.** Algorithm 2, as an approximation algorithm for  $\delta$ -MOSS, computes an objective subset  $\mathcal{F}'$ ,  $\delta$ -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  $\preceq_{\mathcal{F}' \cup \{i\}}^0 \cap \preceq_{\mathcal{F} \setminus (\mathcal{F}' \cup \{i\})}^\delta$  until  $\mathcal{F}'$  is  $\delta$ -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 [14]. Note, that Algorithm 2 not necessarily yields a  $\delta$ -minimal or even  $\delta$ -minimum objective set wrt  $\mathcal{F}$ .

---

**Algorithm 3.** A greedy algorithm for k-EMOSS.

---

```

1: Init:
2:  $\mathcal{F}' := \emptyset$ 
3: while  $|\mathcal{F}'| < k$  do
4:    $\mathcal{F}' := \mathcal{F}' \cup \underset{i \in \mathcal{F} \setminus \mathcal{F}'}{\operatorname{argmin}} \{ \delta_{\min}(\mathcal{F}' \cup \{i\}, \mathcal{F}) \text{ wrt } A \}$ 
5: end while

```

---

**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  $\delta$ . 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  $\delta_{\min}(\mathcal{F}', \mathcal{F}) = 0$ .

## 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 [15] on a linux computer (SunFireV60x with 3060 MHz). Due to space limitations, we refer to [14] for a detailed description of the experimental setups.

**Are All Objectives Necessary?** This issue has been studied for 9 different 0-1-knapsack problem instances [16] and 3 instances for three different continuous test problems, namely DTLZ2, DTLZ5, and DTLZ7 [4]. The Pareto set approximations, generated by IBEA, contain up to 300 solutions for these problems. 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 5-objective-instances and the knapsack instances with 500 items. The results also show that, the more objectives an instance possesses, the more objectives are omissible. 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  $\delta$ -MOSS problem apply for another study, regarding the dominance structure on the whole search space for a small knapsack instance, cf. Fig. 2. 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.

**Table 1.** Sizes (for  $\delta$ -MOSS) and relative errors (for  $k$ -EMOSS) of objective subsets for different problems, computed with the greedy algorithms. For  $\delta$ -MOSS, the  $\delta$  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.

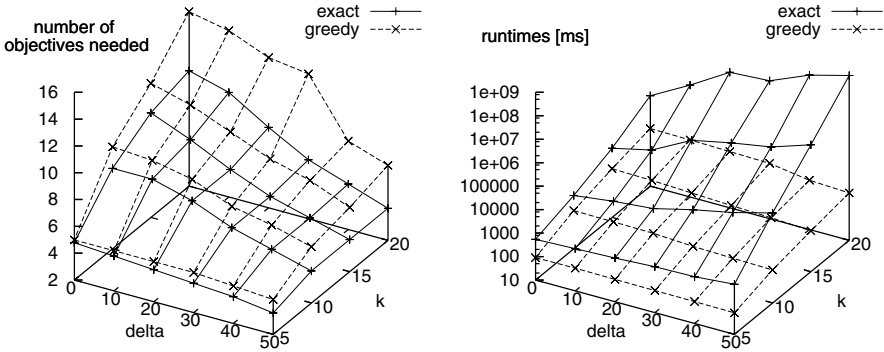
	$\delta$ -MOSS				$k$ -EMOSS		
	0%	10%	20%	40%	30%	60%	90%
knapsack: 100 items, 5 objectives, 100 solutions	5	5	5	5	0.926	0.516	0.486
knapsack, 100 items, 15 objectives, 200 solutions	11	10	10	9	0.818	0.348	0.000
knapsack, 100 items, 25 objectives, 300 solutions	13	13	13	11	0.597	0.000	0.000
knapsack: 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

**Does the Exact Algorithm Outperform the Greedy One?** Fig. 2 shows both the resulting objective set sizes and the running times for the exact and the greedy algorithm on the  $\delta$ -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. For all four choices of the objective set size and all allowed errors  $\delta$ , 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. 2 shows a log scale plot for the running times. Also note, that the running time of the greedy algorithm decreases with higher  $\delta$  which is not self-evident but significant, e.g., in a Wilcoxon rank sum test. Altogether, the results confirm the above observation, the more error is allowed, the more objectives can be omitted. 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 [11] on  $k$ -EMOSS for a knapsack instance with 20 objectives. Table 2 shows the absolute and relative<sup>2</sup>  $\delta$  errors for the objective subsets computed with the method of Deb and Saxena, the exact and the greedy algorithm. With more objectives, the  $\delta$  error gets smaller for all methods. 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.

<sup>2</sup> The relative error  $\delta_{\text{rel}}$  is the absolute error  $\delta_{\text{abs}}$  divided by the spread of the IBEA population, where we define the maximal spread  $S$  of a population  $P$  as the maximal difference of the solutions’ objective values:  $S = \max_{f_i \in \mathcal{F}} \max_{\mathbf{x}, \mathbf{y} \in P} \{|f_i(\mathbf{x}) - f_i(\mathbf{y})|\}$ .





**Fig. 2.** 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.

**Table 2.** Comparison between the approach of Deb and Saxena [11] with the exact and greedy algorithm for  $k$ -EMOSS on a Pareto set approximation of a knapsack instance with 20 objectives. Due to space limitations, we refer to [14] for details.

# obj	Deb and Saxena		exact algorithm		greedy algorithm	
	$\delta_{abs}$	$\delta_{rel}$	$\delta_{abs}$	$\delta_{rel}$	$\delta_{abs}$	$\delta_{rel}$
1	-	-	552	0.9154	552	0.9154
2	603	1.0000	485	0.8043	508	0.8425
3	546	0.9055	447	0.7413	462	0.7662
4	546	0.9055	363	0.6020	418	0.6932
5	-	-	289	0.4793	369	0.6119
6	-	-	129	0.2139	356	0.5904
7	466	0.7728	0	0.0000	324	0.5373
8	466	0.7728	0	0.0000	287	0.4760
9	357	0.5920	0	0.0000	0	0.0000
$\geq 11$	0	0.0000	0	0.0000	0	0.0000

## 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. Coello Coello, C.A., Van Veldhuizen, D.A., Lamont, G.B.: Evolutionary Algorithms for Solving Multi-Objective Problems. Kluwer Academic Publishers, New York (2002)
2. Paechter, B., Rankin, R.C., Cumming, A., Fogarty, T.C.: Timetabling the classes of an entire university with an evolutionary algorithm. In: PPSN V Proceedings, Springer (1998) 865–874
3. Coello Coello, C.A., Hernández Aguirre, A.: Design of Combinational Logic Circuits through an Evolutionary Multiobjective Optimization Approach. Artificial Intelligence for Engineering, Design, Analysis and Manufacture **16**(1) (2002) 39–53
4. Deb, K., Thiele, L., Laumanns, M., Zitzler, E.: Scalable test problems for evolutionary multi-objective optimization. In Abraham, A., Jain, R., Goldberg, R., eds.: Evolutionary Multiobjective Optimization: Theoretical Advances and Applications. Springer (2005) 105–145
5. Emmerich, M., Beume, N., Naujoks, B.: An emo algorithm using the hypervolume measure as selection criterion. In: EMO 2005 Proceedings, Springer (2005) 62–76
6. Knowles, J., Corne, D.: Properties of an adaptive archiving algorithm for storing nondominated vectors. IEEE Transactions on Evolutionary Computation **7**(2) (2003) 100–116
7. While, L.: A new analysis of the lebmeasure algorithm for calculating hypervolume. In: EMO 2005 Proceedings, Springer (2005) 326–340
8. Deb, K.: Multi-objective optimization using evolutionary algorithms. Wiley, Chichester, UK (2001)
9. Purshouse, R.C., Fleming, P.J.: Conflict, harmony, and independence: Relationships in evolutionary multi-criterion optimisation. In: EMO 2003 Proceedings, Springer, Berlin (2003) 16–30
10. Tan, K.C., Khor, E.F., Lee, T.H.: Multiobjective Evolutionary Algorithms and Applications. Springer, London (2005)
11. Deb, K., Saxena, D.K.: On finding pareto-optimal solutions through dimensionality reduction for certain large-dimensional multi-objective optimization problems. Kangal report no. 2005011, Kanpur Genetic Algorithms Laboratory (KanGAL) (2005)
12. Brockhoff, D., Zitzler, E.: On Objective Conflicts and Objective Reduction in Multiple Criteria Optimization. TIK Report 243, ETH Zurich, Zurich, Switzerland (2006) submitted to Operations Research 2006.
13. Zitzler, E., Thiele, L., Laumanns, M., Foneseca, C.M., Grunert da Fonseca, V.: Performance assessment of multiobjective optimizers: An analysis and review. IEEE Transactions on Evolutionary Computation **7**(2) (2003) 117–132
14. Brockhoff, D., Zitzler, E.: Dimensionality Reduction in Multiobjective Optimization with (Partial) Dominance Structure Preservation: Generalized Minimum Objective Subset Problems. TIK Report 247, ETH Zurich, Zurich, Switzerland (2006)
15. Zitzler, E., Künzli, S.: Indicator-based selection in multiobjective search. In: PPSN VIII Proceedings, Springer (2004) 832–842
16. Zitzler, E., Thiele, L.: Multiobjective evolutionary algorithms: A comparative case study and the strength pareto approach. IEEE Transactions on Evolutionary Computation **3**(4) (1999) 257–271