On Set-Based Multiobjective Optimization

Eckart Zitzler, Lothar Thiele, and Johannes Bader
Computer Engineering and Networks Laboratory, ETH Zurich, 8092 Zurich, Switzerland
{eckart.zitzler, lothar.thiele, johannes.bader}@tik.ee.ethz.ch

TIK-Report No. 300
February 22, 2008

***Abstract*—Assuming that evolutionary multiobjective optimization (EMO) mainly deals with set problems, one can identify three core questions in this area of research: (i) how to formalize what type of Pareto set approximation is sought, (ii) how to use this information within an algorithm to efficiently search for a good Pareto set approximation, and (iii) how to compare the Pareto set approximations generated by different optimizers with respect to the formalized optimization goal. There is a vast amount of studies addressing these issues from different angles, but so far only few studies can be found that consider all questions under one roof.**

This paper is an attempt to summarize recent developments in the EMO field within a unifying theory of set-based multiobjective search. It discusses how preference relations on sets can be formally defined, gives examples for selected user preferences, and proposes a general, preference-independent hill climber for multiobjective optimization with theoretical convergence properties. Furthermore, it shows how to use set preference relations for statistical performance assessment and provides corresponding experimental results. The proposed methodology brings together preference articulation, algorithm design, and performance assessment under one framework and thereby opens up a new perspective on EMO.

I. Introduction

By far most publications within the field of evolutionary multiobjective optimization (EMO) focus on the issue of generating a suitable approximation of the Pareto-optimal set, or Pareto set approximation for short. For instance, the first book on EMO by Kalyanmoy Deb [7] is mainly devoted to techniques of finding multiple trade-off solutions using evolutionary algorithms.

Taking this view, one can state that EMO in general deals with set problems: the search space Ψ consists of all potential Pareto set approximations rather than single solutions, i.e., Ψ is a set of sets. Furthermore, an appropriate order on Ψ is required to fully specifiy the set problem—this order will here be denoted as set preference relation. A set preference relation provides the information on the basis of which the search is carried out; for any two Pareto set approximations, it says whether one set is better or not. Set preference relations can be defined, e.g., via set quality measures, which can be regarded as objective functions for sets, or directly using a binary relation, e.g., extended Pareto-dominance on sets. Transforming the original problem into such a set problem offers many advantages for decision making, but also poses new challenges in terms of search space complexity.

In the light of this dicussion, three core research issues can be identified in the EMO field: (i) how to formalize the optimization goal in the sense of specifying what type of set is sought, (ii) how to effectively search for a suitable set to achieve the formalized optimization goal, and (iii) how to evaluate the outcomes of multiobjective optimizers with respect to the underlying set problem.

The question of what a good set of compromise solutions is depends on the preferences of the decision maker. Suppose, for instance, that a few representative solutions are to be selected from a large Pareto-optimal set; the actual choice can be quite different for different users. Many multiobjective evolutionary algorithms (MOEAs) aim at generating a subset that is uniformly distributed in the objective space [15], [8], [33], others consider the subset maximizing the hypervolume of the dominated objective space as the best one [14], [19]. There are many more possibilites depending on the user and the situation: one may be interested in the extreme regions of the Pareto-optimal set [9], in convex portions of the trade-off surface (knee points) [3], or in regions close to a predefined ideal point [11], to name only a few. These preferences are usually hard-coded in the algorithm, although there have been attemtps to design methods with more flexibility with respect to user preferences [32], [13].

Overall, one can conclude that EMO in general relies on *set preferences*, i.e., preference information that specifies whether one Pareto set approximation is better than another, and that MOEAs usually implement different types of set preferences. There are various studies that focus on the issue of preference articulation in EMO, in particular integrating additional preferences such as priorities, goals, and reference points [16], [4], [6], [2], [11], [25]. However, these studies mainly cover preferences on solutions and not preferences on sets. Furthermore, there is a large amount of publications that deal with the definition and the application of quantitative quality measures for Pareto set approximations [18], [34], [29], [20], [35]. These quality measures or quality indicators reflect set preferences and have been widely employed to compare the outcomes generated by different MOEAs. Moreover, in recent years a trend can be observed to directly use specific measures such as the hypervolume indicator and the epsilon indicator in the search process [24], [21], [32], [10], [14], [19], [1]. Nevertheless, a *general* methodology to formalize set preferences and to use them for optimization is missing.

This paper represents one step towards such an overarching methodology. It proposes

1) a theory of set preference relations that clarifies how user preferences on Pareto set approximations can be formalized on the basis of quality indicators and what criteria such formalizations must fulfill; introduces
2) a general set-preference based search algorithm that can be flexibly adapted to arbitrary types of set preference relations and that can be shown to converge under certain assumptions; and discusses
3) an approach to statistically compare the outcomes of multiple search algorithms with respect to a specific set preference relation.

The novelty of this approach is that it brings all aspects of preference articulation, multiobjective search, and performance assessment under one roof, while achieving a clear separation of concerns. This offers several benefits: (i) it provides flexibility to the decision maker as he can change his preferences without the need to modify the search algorithm, (ii) the search can be better guided which is particularly important in the context of high-dimensional objective spaces, (iii) algorithms designed to meet specific preferences can be compared on a fair basis since the optimization goal can be explicitly formulated in terms of the underlying set preference relation.

In the following, we will first provide the formal basis of set preference relations and introduce fundamental concepts. Afterwards, we will discuss how set preference relations can be designed using quality indicators and present some example relations. A general, preference-based multiobjective search algorithm will be proposed in Section IV, including a discussion of convergence properties. Finally, Section V presents a methodology to compare algorithms with respect to a given set preference relation and provides experimental results for selected preferences.

II. A New Perspective: Set Preference Relations

As has been described in the introduction, multiobjective optimization will be viewed as a preference-based optimization on sets. The purpose of this section is to formally define the notation of optimization and optimality in this context and provides the necessary foundations for the practical algorithms described in the forthcoming sections.

A. Basic Terms

We are considering the optimization of vector-valued objective functions $f = (f_1, ..., f_n) : \mathcal{X} \longrightarrow \mathbb{R}^n$. Here, $\mathcal{X}$ denotes the feasible set of solutions in the decision space, i.e. the set of alternatives of the decision problem. A single alternative $x \in \mathcal{X}$ will be denoted as a decision vector or solution. The image of $\mathcal{X}$ under f is denoted as the feasible set in the objective space $\mathcal{Z} = f(\mathcal{X}) = \{y \in \mathbb{R}^n : y = f(x), x \in \mathcal{X}\}$ where $\mathcal{Z} \subseteq \mathbb{R}^n$. The objectives of a single alternative $y = f(x)$ will be denoted as an objective vector. Fig. 1 illustrates a possible scenario with 7 solutions in the feasible set and a two-dimensional objective space $n = 2$.

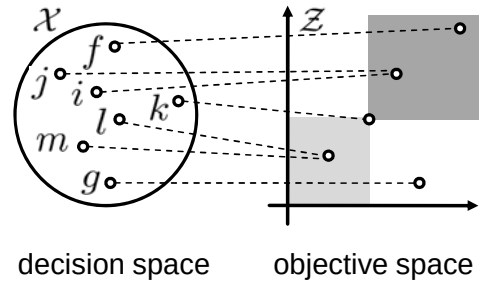


Fig. 1. Optimization scenario with $\mathcal{X} = \{f, g, i, j, k, l, m\}$ and $n = 2$. In the case that the preference relation is the weak Pareto-dominance relation $\preceq_{\text{par}}$, the shaded areas represent locations of solutions that are dominated by k (dark) and that dominate k (light).

The optimization problem can now be formulated as

$$\text{"min" } f(x) \text{ subject to } x \in \mathcal{X} \tag{1}$$

The use of quotation marks indicates that we need to precisely define what minimization means in the context of vector-valued objective functions. It will not be possible in general to find a single decision vector that minimizes all objectives simultaneously.

In order to allow for optimization in such a situation, we need a preference relation $a \preceq b$ on the feasible set in the decision space which denotes that a solution a is at least as good as a solution b. In the context of this paper, we will suppose that all preference relations are preorders[1]. A well known preference relation in the context of multi-objective optimization is the (weak) Pareto-dominance.

Definition 2.1: A solution $a \in \mathcal{X}$ weakly Pareto-dominates a solution $b \in \mathcal{X}$, denoted as $a \preceq_{\text{par}} b$, if it is as least as good in all objectives, i.e. $f_i(a) \leq f_i(b)$ for all $1 \leq i \leq n$.

In Fig. 1, we find for example for solution k that $k \preceq_{\text{par}} i$, $k \preceq_{\text{par}} j$, $k \preceq_{\text{par}} f$, $l \preceq_{\text{par}} k$ and $m \preceq_{\text{par}} k$. The 'area' for all objective vectors that belong to solutions that are weakly Pareto-dominated by k is depicted as dark grey, the 'area' for those who dominate k is colored with light grey.

Note that $\preceq_{\text{par}}$ is only one example of a useful preference relation. The results described in this paper are not restricted to the concept of Pareto-dominance but hold for any preference relation defined on the set of solutions.

The situation that a solution a is at least as good as a solution b will also be denoted as a being weakly preferable to b in this paper. Moreover, we will use the following terms:

- A solution a is strictly better than or preferable to a solution b (denoted as $a \prec b$) if $a \preceq b \wedge b \not\preceq a$. Consequently, in case of the Pareto-dominance, we say that a strictly dominates b if $\forall i : f_i(a) \leq f_i(b)$ and $\exists i : f_i(a) < f_i(b)$. In Fig. 1 we find for example $k \prec_{\text{par}} i$.

[1]A preorder $\preceq$ on a given set S is reflexive and transitive: $a \preceq a$ and $(a \preceq b \wedge b \preceq c) \Rightarrow (a \preceq c)$ holds for all a, b and c in S.

- A solution a is incomparable to a solution b ($a \parallel b$) if $a \not\preceq b \wedge b \not\preceq a$. In Fig. 1 we find for example $k \parallel_{\text{par}} g$.
- A solution a is equivalent or indifferent to a solution b (denoted as $a \equiv b$) if $a \preceq b \wedge b \preceq a$. We say that a set of solutions form an equivalence class, if they are mutually equivalent. In case of the Pareto-dominance, we have $f(a) = f(b)$. In Fig. 1 we find for example $l \equiv_{\text{par}} m$ and therefore, l and m form an equivalence class.
- We denote the set of minimal elements[2] in the ordered set $(S, \preceq)$ as

$$\text{Min}(S, \preceq) = \{x \in S \,:\, \nexists a \in S \text{ such that } a \prec x\}$$

  For the example in Fig. 1, there are three minimal elements: $\text{Min}(S, \preceq_{\text{par}}) = \{l, m, g\}$.

We will call the set

$$\mathcal{X}^* = \text{Min}(\mathcal{X}, \preceq)$$

the optimal set of $\mathcal{X}$ with respect to $\preceq$ and an element $x^* \in \mathcal{X}^*$ is an optimal solution. Optimization according to (1) may now be termed as finding a minimal element in the ordered feasible set $(\mathcal{X}, \preceq)$.

Preference relations can also be depicted graphically. To this end, the elements of the underlying set are represented as nodes of a graph and the relation itself defines the edges: if $a \preceq b$, then there is an edge from a to b. For better readability, usually not all edges are shown; we will omit those that can be inferred from the reflexitiy and transitivity properties of the preference relation. Fig. 2 shows a possible preordered set of solutions $\mathcal{X} = \{a, ..., m\}$. In particular, the preference relations between $\{f, g, i, k, l, m\}$ correspond directly to the scenario shown in Fig. 1.

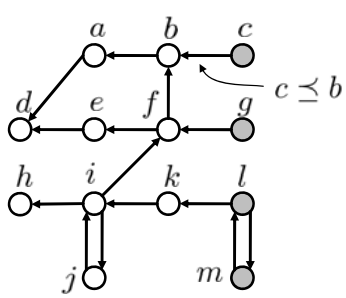


Fig. 2. Representation of a preordered set $(\mathcal{X}, \preceq)$ where $\mathcal{X}$ consists of the solutions $\{a, ..., m\}$. The optimal solutions are $\text{Min}(\mathcal{X}, \preceq) = \{c, g, l, m\}$. $\{i, j\}$ and $\{l, m\}$ form two equivalence classes, i.e. $i \equiv j$ and $l \equiv m$. As $b \not\preceq c$ and $c \preceq b$, we find $c \prec b$.

B. Approximation Of The Pareto-optimal Set

As a preference relation $\preceq$ defined above is usually not a total order on the feasible set[3], we will often have many optimal solutions, i.e. many minimal elements, that reflect the different trade-offs present in the objective function.

[2]A minimal element u of an ordered set $(S, \preceq)$ set with a preorder $\preceq$ satisfies: If $a \preceq u$ for some a in the set, then $u \preceq a$.

[3]A preference relation on a set S is called total, if $(a \preceq b) \vee (b \preceq a)$ holds for all $a, b \in S$.

In particular, this holds for the Pareto preference relation $\preceq_{\text{par}}$. As a result, we may not only be interested in one of these minimal elements but in a carefully selected subset that reflects additional preference information of some decision maker. Traditional evolutionary multiobjective optimization methods attempt to solve this problem by maintaining and improving sets of decision vectors, denoted as populations. The corresponding optimization algorithms are tuned to anticipated preferences of a decision maker.

Thus, the underlying goal of population-based multiobjective optimization can be described as determining a (small-sized) set of alternative solutions

1) that contains as many as possible different decision vectors that are minimal with respect to a preference relation *on the feasible set in the decision space* (for example the weak Pareto-dominance according to Definition 2.1), and
2) whose selection of minimal and non-minimal decision vectors reflects the preferences of the decision maker.

As pointed out in Section I, it is the purpose of the paper to define set-based multiobjective optimization on the basis of these two aspects. In contrast to previous results, the second item as defined above is made formal and treated as a first class citizen in theory and optimization algorithms. This not only leads to a better understanding of classical population-based multiobjective optimization but also allows for defining set-based methods with corresponding convergence results as well as statistical tests to compare different algorithms. Finally, a new set of optimization algorithms can be obtained which directly can take preference information into account.

Therefore, we need to formalize the preferences of a decision maker on the subset of decision vectors in an optimal set of solutions. This will be done by defining a preorder $\preccurlyeq$ on the set of all possible sets of solutions. A set of solutions P is defined as a set of decision vectors, i.e. $x \in P \Rightarrow x \in \mathcal{X}$. A set of all admissible sets, e.g. sets of finite size, is denoted as Ψ, i.e. $P \in \Psi$.

We define set-based multiobjective optimization as finding a minimal element of the ordered set $(\Psi, \preccurlyeq)$ where Ψ is a set of admissible sets of solutions.

We can summarize the elements of a set-based multiobjective optimization problem as follows:

- A set of feasible solutions $\mathcal{X}$,
- a vector-valued objective function $f : \mathcal{X} \longrightarrow \mathbb{R}^n$,
- a set Ψ of all admissible sets P of decision vectors with $x \in P \Rightarrow x \in \mathcal{X}$,
- a preference relation $\preccurlyeq$ on Ψ.

In the light of the above discussion, the preference relation $\preccurlyeq$ needs to satisfy the aforementioned two conditions, whereas the first one guarantees that we actually solve (1) and the second one allows to add preferences of the decision maker. In the next section, we will discuss the necessary properties of suitable preference relations and the concept of refinement.

C. Preference Relations

We will construct $\preccurlyeq$ in two successive steps. At first, a general set-based preference relation (a set preference relation) $\preccurlyeq$ will be defined that is *conforming* to a solution-based preference relation $\preceq$. This set preference relation will then be *refined* by adding preferences of a decision maker in order to possibly obtain a total order. For a *conforming* set preference relation we require that no solution is excluded that could be interesting to a decision maker. In addition, if for each solution $b \in B$ there is some solution $a \in A$ which is at least as good, then we can safely say that A is at least as good as, or weakly preferable to B.

From the above considerations, the definition of a conforming set-based preference relation follows directly; it is in accordance to the formulations used in [18], [35].

Definition 2.2: Let be given a set $\mathcal{X}$ and a set Ψ whose elements are sets of elements $x \in \mathcal{X}$, i.e. sets of solutions. Then the preference relation $\preccurlyeq$ on Ψ conforms to $\preceq$ on $\mathcal{X}$ if for all $A, B \in \Psi$

$$A \preccurlyeq B \;\Leftrightarrow (\forall b \in B : (\exists a \in A : a \preceq b))$$

As an example, Fig. 3 shows three sets of solutions A, B and G. According to the above definition, we find that $B \preccurlyeq A$ and $G \preccurlyeq A$. As sets B and G are incomparable, we have $B \parallel G$.

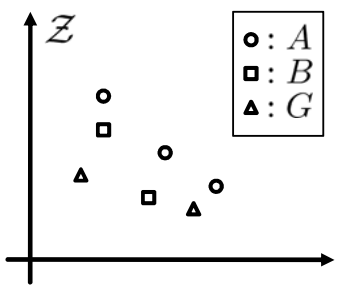


Fig. 3. Representation of a preordered set of sets of solutions $\{A, B, G\} \in \Psi$ where $\preceq_{\text{par}}$ is assumed to be the underlying solution-based preference relation. We find $B \preccurlyeq A$, $G \preccurlyeq A$ and $B \parallel G$, i.e. B and G are incomparable.

Now, it will be shown that the above preference relation is indeed suitable for optimization.

Theorem 2.3: The set preference relation $\preccurlyeq$ as defined in Definition 2.2 is a preorder.

Proof: $A \preccurlyeq A$ is true as $\forall a' \in A : (\exists a \in A : a \preceq a')$ holds with $a' = a$. Now we need to show that $A \preccurlyeq B \wedge B \preccurlyeq C \Rightarrow A \preccurlyeq C$. Given an arbitrary $c^* \in C$. From $B \preccurlyeq C$ we know that $\exists b^* \in B$ such that $b^* \preceq c^*$. From $A \preccurlyeq B$ we know that $\exists a^* \in A$ such that $a^* \preceq b^*$. From the transitivity of $\preceq$ we can conclude that $a^* \preceq c^*$ holds. As a result, $\forall c^* \in C : (\exists a^* \in A : a^* \preceq c^*)$. ■

Therefore, we can also represent the relations between the sets of solutions in form of an acyclic graph whose nodes correspond to sets of solutions and edges correspond to the set preference relation $\preccurlyeq$. As an example, Fig. 4 represents such a diagram. Note that the relation between the sets A, B and G corresponds exactly to the scenario shown in Fig. 3. The minimal sets are $\text{Min}(\Psi, \preccurlyeq) = \{C, D, H\}$.

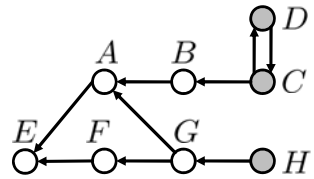


Fig. 4. Representation of a preordered set of sets of solutions $\Psi = \{A, B, C, D, E, F, G, H\}$. The minimal sets are $\text{Min}(\Psi, \preccurlyeq) = \{C, D, H\}$.

Roughly speaking, the above definition of a conforming set-based preference relation ensures, that each set in $\text{Min}(\Psi, \preccurlyeq)$ contains all minimal solutions $\mathcal{X}^* = \text{Min}(X, \preceq)$. More precisely, let us recall that all mutually equivalent elements of $\mathcal{X}^*$ form an equivalence class. Then we can state that each set in $\text{Min}(\Psi, \preccurlyeq)$ contains at least one representative of each equivalence class of $\mathcal{X}^*$. Therefore, no preferences other than that contained in $\preceq$ is included so far.

For practical reasons, one may deal with sets of solutions that have an upper bound m on their size, i.e. if $P \in \Psi_m$ then $|P| \leq m$. In this case, the properties of minimal elements of $(\Psi_m, \preccurlyeq)$ are somewhat more complex as described in the following corollary whose proof is given in the Appendix. The size constraint cuts off part of the preordered set and the set of minimal elements as well as their characterization changes.

Corollary 2.4: Let s denote the number of equivalence classes of $\mathcal{X}^* = \text{Min}(X, \preceq)$, i.e. the maximal number of solutions that are not equivalent. m denotes the number of solutions in any set $P \in \Psi_m$. The set of sets Ψ_m is ordered by $\preccurlyeq$ which conforms to $\preceq$ according to Definition 2.2. Then the following two statements hold:

- Suppose $m \geq s$, i.e. there are at most as many equivalence classes in $\mathcal{X}^*$ as there are solutions in $P \in \Psi_m$. Then each $P^* \in \text{Min}(\Psi_m, \preccurlyeq)$ contains at least one solution for each equivalence class in $\mathcal{X}^*$, i.e. at least one representative for each set of equivalent minimal solutions.
- Suppose $m < s$, i.e. there are more equivalence classes in $\mathcal{X}^*$ than there are solutions in $P \in \Psi_m$. Then a set P is an element of $\text{Min}(\Psi_m, \preccurlyeq)$ (i.e. it is a minimal set of solutions) iff it contains m minimal solutions that are not equivalent.

If we look at Corollary 2.4 and the important case $m < s$ (a small set size), then a minimal set of solutions may contain one representative of each equivalence class only. If we are interested in the solutions themselves, then we may want to have the possibility to determine all solutions that lead to the same value in objective space. They may well be different in the decision space and therefore, be interesting to a decision maker, see Fig. 2. This can be achieved by replacing a given solution-based preference relation $\preceq$ that considers objective values only, e.g. $\preceq_{\text{par}}$. Equivalent solutions that lead to the same objective values should become incomparable. As a result, solutions will be equivalent only if they are identical in the decision space. We denote such a preference relation as $\dot{\preceq}$ with

$$a \dot{\preceq} b \Leftrightarrow (a = b) \vee (a \prec b)$$

This new preference relation is a preorder as it is reflexive ($a \dot{\preceq} a$ clearly holds) and transitive. The latter property can be shown as follows: $a \dot{\preceq} b \wedge b \dot{\preceq} c \Rightarrow ((a = b) \vee (a \prec b)) \wedge ((b = c) \vee (b \prec c)) \Rightarrow ((a = c) \vee (a \prec c)) \Rightarrow a \dot{\preceq} c$. Fig. 5 shows the order graph if $\preceq_{\text{par}}$ is replaced by $\dot{\preceq}_{\text{par}}$ in Fig. 2. As can be seen, the equivalence classes are not present anymore and i, j as well as l, m are now considered as incomparable.

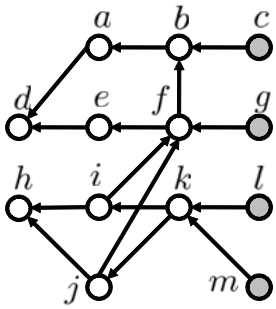


Fig. 5. Representation of the preordered set of solutions from Fig. 2 if $\preceq_{\text{par}}$ is replaced by $\dot{\preceq}_{\text{par}}$.

Now, it is interesting to see how a set-based preference relation looks like that is conforming to $\dot{\preceq}$. Of course, we could directly use Definition 2.2 and state

$$A \dot{\preccurlyeq} B \;\Leftrightarrow (\forall b \in B : (\exists a \in A : a \dot{\preceq} b)) \tag{2}$$

But because of the close relationship between $\preceq$ and $\dot{\preceq}$, we can also give a more intuitive representation in terms of the original preorder $\preceq$:

$$A \dot{\preccurlyeq} B \;\Leftrightarrow \text{Min}(A \cup B, \preceq) \subseteq A$$

In other words, an element $b \in B$ either is in A also ($b \in A$) or otherwise, it is not minimal in $A \cup \{b\}$. In the latter case, there exists an element $a \in A$ which is better ($a \prec b$). This is the same condition as in (2).

The above definition of a preference relation $\dot{\preceq}$ derived from $\preceq$ ensures, that the minimal set of sets $\text{Min}(\Psi, \dot{\preccurlyeq})$ contains sets with all possible combinations of minimal elements $\text{Min}(\mathcal{X}, \preceq)$. If we use the new preference relation in the context of Corollary 2.4, we can state that each minimal set $P^* \in \text{Min}(\Psi_m, \dot{\preccurlyeq})$ now contains all minimal solutions in $\mathcal{X}^* = \text{Min}(\mathcal{X}, \preceq)$ if its size is $m \geq |\mathcal{X}^*|$ and it contains m minimal solutions if $m < |\mathcal{X}^*|$.

D. Refinements

What remains to be done is to define the notion of refinement, as we need to have a possibility to include preference information into a conforming preference relation. This way, we can include preference information of a decision maker and optimize towards a set which contains a preferred subset of all minimal solutions, i.e., nondominated solutions in the case of Pareto-dominance. The goal of such a refinement is twofold: At first, the given preorder should become ”more total”. This way, there are less incomparable sets of solutions which are hard to deal with by any optimization method. Second, the refinement will allow to explicitly take into account preference information of a decision maker.

An example is shown in Fig. 6, where three edges (preference relations) have been added and the resulting ordering is a total preorder with the optimal set of solutions H. Just adding an ordering among incomparable solutions potentially leads to cycles in the ordering as the resulting structure is no longer a preorder. Using such an approach in optimization will prevent convergence in general, see also Fig. 7.

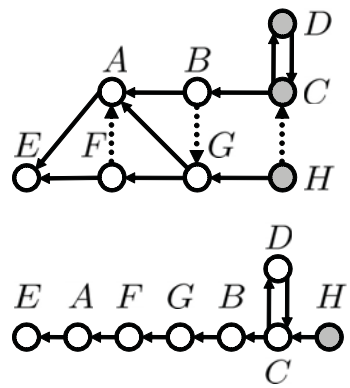


Fig. 6. Including preference information may create a total preorder that can be used for optimization. Here, three preferences $F \preccurlyeq A$, $B \preccurlyeq G$ and $H \preccurlyeq C$ have been added to the preorder shown in Fig. 4. The resulting total preorder is shown at the bottom.

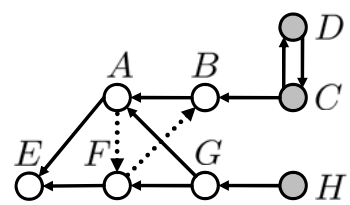


Fig. 7. Including preference information arbitrarily may create cycles in the optimization. Here, two preferences $A \preccurlyeq F$ and $F \preccurlyeq B$ have been added to the preorder shown in Fig. 4.

For the refinement, we require the following properties:

- The refinement should again be a preorder.
- If a set is minimal in the refined order for some subset of Ψ, it should also be minimal in the original order in the same subset. This way, we guarantee that we actually solve (1) with respect to some preference relation, e.g. Pareto-dominance.

As a result of this discussion, we obtain the following definition:

Definition 2.5: Given a set Ψ. Then the preference relation $\preccurlyeq_{\text{ref}}$ refines $\preccurlyeq$ if for all $A, B \in \Psi$ we have

$$(A \preccurlyeq B) \wedge (B \not\preccurlyeq A) \Rightarrow (A \preccurlyeq_{\text{ref}} B) \wedge (B \not\preccurlyeq_{\text{ref}} A)$$

Examples of legal refinements are depicted in Fig. 8. Note, that the refinement still needs to be a preorder.

If we use the notion of strictly better, we can also derive the condition $A \prec B \Rightarrow A \prec_{\text{ref}} B$. In other words, if in the given preference relation a set A is strictly better than a

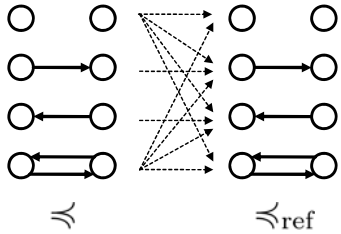


Fig. 8. The left hand side shows the four different possibilities between two nodes of the given preference relation: no edge (incomparable), single edge (one is better than the other) and double edge (equivalent). The right hand side shows the possibility in case of the refined relation. The dotted edges represent all possible changes of edges if $\preccurlyeq$ is refined to $\preccurlyeq_{\text{ref}}$.

set B ($A \prec B$) then it must be strictly better in the refined relation too ($A \prec_{\text{ref}} B$). As can be seen, refining a preference relation maintains existing strict preference relations. If two sets are incomparable, i.e. $A \parallel B = (A \not\preccurlyeq B) \wedge (B \not\preccurlyeq A)$, then additional relations can be added by the refinement. In case of equivalence, i.e. $A \equiv B = (A \preccurlyeq B) \wedge (B \preccurlyeq A)$, relations can be removed.

As will be seen in the next chapter, some of the widely used preference relations are not refinements in the sense of Def. 2.5 but satisfy a weaker condition.

Definition 2.6: Given a set Ψ. Then the set preference relation $\preccurlyeq_{\text{ref}}$ weakly refines $\preccurlyeq$ if for all $A, B \in \Psi$ we have

$$(A \preccurlyeq B) \wedge (B \not\preccurlyeq A) \Rightarrow (A \preccurlyeq_{\text{ref}} B)$$

In other words, if set A is strictly better than B ($A \prec B$), than A weakly dominates B in the refined preference relation, i.e. $A \preccurlyeq_{\text{ref}} B$. Therefore, A could be incomparable to B in the refined preference relation, i.e. $A \parallel_{\text{ref}} B$. In addition, if a preference relation refines another one, it also weakly refines it. The following Fig. 9 depicts all possibilities of a weak refinements. The weak refinement still needs to be a preorder.

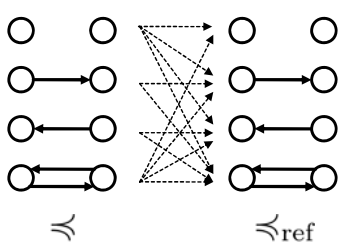


Fig. 9. The left hand side shows the four different possibilities between two nodes of the given preference relation: no edge (incomparable), single edge (one is better than the other) and double edge (equivalent). The right hand side shows the possibility in case of the weakly refined relation. The dotted edges represent all possible changes of edges if $\preccurlyeq$ is weakly refined to $\preccurlyeq_{\text{ref}}$.

The following hierarchical construction of refinement relations will be used in later sections for several purposes. At first, it allows to convert a given weak refinement into a refinement. This way, a larger class of available indicators and preference relations can be used. In addition, it provides a simple method to add decision maker preference information to a given relation by adding an order to equivalent sets. This way, a preorder can be made 'more total'. Finally, it enables to refine a given preorder in a way that helps to speed up the convergence of an optimization algorithm, e.g. by taking into account also solutions that are worse than others in a set. This way, the successful technique of non-dominated sorting can be used in the context of set-based optimization.

The following construction resembles the concept of hierarchy used in [16], but (a) we are dealing with preference relations on sets and (b) the hierarchical construction is different.

Definition 2.7: Given a set Ψ and a sequence S of k preference relations over Ψ with $S = (\preccurlyeq^1, \preccurlyeq^2, \ldots, \preccurlyeq^k)$. Then the preference relation $\preccurlyeq_{\text{S}}$ associated with S is defined as follows: Let $A, B \in \Psi$. Then $A \preccurlyeq_{\text{S}} B$ if and only if $\exists 1 \leq i \leq k$ such that the following two conditions are satisfied:

(i) $(i < k \wedge (A \prec^i B)) \vee (i = k \wedge (A \preccurlyeq^k B))$
(ii) $\forall 1 \leq j < i : (A \preccurlyeq^j B \wedge B \preccurlyeq^j A)$

With this definition, we can derive the following procedure to determine $A \preccurlyeq_{\text{S}} B$ for two sets A and B:

- Start from the first preference relation, i.e. $j = 1$. Repeat the following step: If $A \equiv^j B$ holds (A and B are equivalent), then increase j to point to the next relation in the sequence if it exists.
- If the final j points to the last preference relation ($j = k$), then set $A \preccurlyeq_{\text{S}} B \Leftrightarrow A \preccurlyeq^k B$. Otherwise, set $A \preccurlyeq_{\text{S}} B \Leftrightarrow A \prec^k B$.

At first, we need to show that the sequence of preference relations $\preccurlyeq_{\text{S}}$ is again a preorder. The proof of the theorem is given in the Appendix.

Theorem 2.8: If all preference relations $\preccurlyeq^j$, $1 \leq j \leq k$ in Def. 2.7 are preorders, then $\preccurlyeq_{\text{S}}$ is a preorder.

For optimization purposes, it is desirable that the preference relations used are total preorders. In this case, there are no incomparable sets. The following theorem states a corresponding invariance property of the hierarchical construction in Def. 2.7. Again, the proof is given in the Appendix.

Theorem 2.9: If all preference relations $\preccurlyeq^j$, $1 \leq j \leq k$ in Def. 2.7 are total preorders, then $\preccurlyeq_{\text{S}}$ is a total preorder.

As described above, one of the main reasons to define a sequence of preference relations is to upgrade a given weak refinement to a refinement. In addition, it would be desirable to add arbitrary preorders to the sequence S. As they need not to be refinements of the given order $\preccurlyeq$, a decision make cam freely add his preferences this way. The following theorem states the corresponding result. The proof is provided in the Appendix.

Theorem 2.10: Given a sequence of preference relations according to Def. 2.7. Suppose there is a $k' \leq k$ such that $\preccurlyeq^{k'}$ is a refinement of a given preference relation $\preccurlyeq$. In addition, all relations $\preccurlyeq^j$, $1 \leq j < k'$ are weak refinements of $\preccurlyeq$ and all relations $\preccurlyeq^j$, $k' < j \leq k$ are preorders. Then $\preccurlyeq_{\text{S}}$ is a refinement of $\preccurlyeq$.

All preorders $\preccurlyeq^j$, $k' < j \leq k$ can be arbitrary preorders that may reflect additional preferences, see also Fig. 10. Nevertheless, the resulting preference relation $\preccurlyeq_{\text{S}}$ still refines

$\preccurlyeq$. If this relation was conformant to the solution-based preference relation then the results of Corollary 2.4 still hold. The previously described hierarchical construction of refinements will be applied in later sections of the paper to construct preference relations that are useful for set-based multiobjective optimization.

$$S = (\underbrace{\preccurlyeq^1, \ldots, \preccurlyeq^{k'-1}}_{\text{weak refinement}}, \overset{\text{refinement}}{\preccurlyeq^{k'}}, \underbrace{\preccurlyeq^{k'+1}, \ldots, \preccurlyeq^{k}}_{\text{preorder}})$$

Fig. 10. Representation of the hierarchical construction of refinements according to Theorem 2.10.

III. Design of Preference Relations Using Quality Indicators

A. Unary Indicators

Unary quality indicators are a possible means to construct set preference relations that on the one hand are total orders and on the other hand satisfy the refinement property, cf. Definition 2.5. They represent set quality measures that map each set $A \in \Psi$ to a real number $I(A) \in \mathbb{R}$. Given an indicator I, one can define the corresponding preference relation as

$$A \preccurlyeq_{\mathrm{I}} B := (I(A) \leq I(B)) \tag{3}$$

where we assume that smaller indicator values stand for higher quality, in other words, A is as least as good as B if the indicator value of A is not larger than the one of B. By construction, the preference relation $\preccurlyeq_{\mathrm{I}}$ defined above is a preorder as it is reflexive as $I(A) \leq I(A)$ and transitive as $(I(A) \leq I(B)) \wedge (I(B) \leq I(C)) \Rightarrow I(A) \leq I(C)$. Moreover, it is a total preorder because $(I(A) \leq I(B)) \vee (I(B) \leq I(A))$ holds. Note that depending on the choice of the indicator function, there may be still sets that have equal indicator values, i.e., they are indifferent with respect to the corresponding set preference relation $\preccurlyeq_{\mathrm{I}}$. In this case, we may have equivalence classes of sets, each one containing sets with the same indicator value. For multiobjective optimization algorithms that use indicators as their means of defining progress, sets with identical indicator values pose additional difficulties in terms of cyclic behavior and premature convergence. We will later see how these problems can be circumvented by considering hierarchies of indicators.

Clearly, not all possible indicator functions realize a refinement of the orginal preference relation, e.g., weak Pareto-dominance. The following theorem provides sufficient conditions for weak refinements and refinements.

Theorem 3.1: If a unary indicator I satisfies

$$(A \preccurlyeq B) \wedge (B \not\preccurlyeq A) \Rightarrow (I(A) \leq I(B))$$

for all $A, B \in \Psi$, then the corresponding preference relation $\preccurlyeq_{\mathrm{I}}$ according to (3) weakly refines the preference relation $\preccurlyeq$ according to Definition 2.6. If it holds that

$$(A \preccurlyeq B) \wedge (B \not\preccurlyeq A) \Rightarrow (I(A) < I(B))$$

then $\preccurlyeq_{\mathrm{I}}$ refines $\preccurlyeq$ according to Definition 2.5.

Proof: Consider $A, B \in \Omega$ with $(A \preccurlyeq B) \wedge (B \not\preccurlyeq A)$. If $I(A) \leq I(B)$, then also $A \preccurlyeq_{\mathrm{I}} B$ according to (3). If $I(A) < I(B)$, then $I(A) \leq I(B)$, but $I(B) \not\leq I(A)$, which implies that $A \preccurlyeq_{\mathrm{I}} B$ and $B \not\preccurlyeq_{\mathrm{I}} A$. ■

In other words, if A is strictly better than B, i.e. $A \prec B$, then the indicator value of A must be not worse or must be smaller than the one of B in order to achieve a weak refinement or a refinement, respectively. In practice, this global property may be difficult to prove for a specific indicator since one has to argue over all possible sets. Therefore, the following theorem provides sufficient conditions that are only based on the local behavior, i.e., when adding a single element. The proof of the theorem is given in the Appendix.

Theorem 3.2: Let I be a unary indicator and $\preccurlyeq$ a preference relation on populations that itself conforms to a preference relation $\preceq$ on its elements (see Definition 2.2). The relation $\preccurlyeq_{\mathrm{I}}$ according to (3) refines $\preccurlyeq$ if the following two conditions hold for all sets $A \in \Psi$ and solutions b with $\{b\} \in \Psi$:

1) If $A \preccurlyeq \{b\}$ then $I(A \cup \{b\}) = I(A)$.
2) If $A \not\preccurlyeq \{b\}$ then $I(A \cup \{b\}) < I(A)$.

For weak refinement one needs to replace the relation $<$ by $\leq$ in the second condition.

Using these two conditions, it is straight-forward to prove that an indicator induces a refinement of the weak Pareto-dominance relation. The hypervolume indicator [34], [31], for instance, gives the volume of the objective space weakly dominated by a set of solutions with respect to a given set of reference points $R \subset \mathcal{Z}$. We define the corresponding indicator function as the negative volume $I_H(A) = -\lambda(H(A, R))$ where λ denotes the Lebesgue measure and

$$H(A, R) = \{h \mid \exists a \in A \; \exists r \in R \; : \; f(a) \leq h \leq r\}$$

Now, it is easy to see that the volume is not affected whenever a weakly Pareto-dominated solution is added to a set A. Furthermore, any solution b not weakly Pareto-dominated by A covers a part of the objective space not covered by A and therefore the indicator value for $(A \cup \{b\})$ is better (smaller) than the one for A. These properties can easily be verified by looking at the example shown in Fig. 11. Furthermore, there are various other unary indicators which induce weak refinements, e.g., the unary R_2 and R_3 indicators [18] and the epsilon indicator [35]—the above conditions can be used to show this, see also [20], [35] for a more detailed discussion.

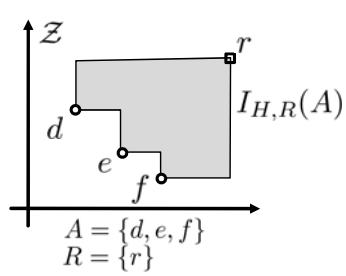


Fig. 11. Graphical representation of the unary hypervolume indicator.

One may also ask what conditions are necessary for an

indicator to be a (weak) refinement. The following corollary shows that the second condition in Theorem 3.2 is also a necessary condition.

Corollary 3.3: Given a unary indicator I and a set preference relation $\preccurlyeq$ that itself conforms to a preference relation $\preceq$ on its elements, then the condition

$$A \not\preccurlyeq \{b\} \Rightarrow I(A \cup \{b\}) < I(A)$$

for all sets $A \in \Psi$ and solutions b with $\{b\} \in \Psi$ is a necessary condition for $\preccurlyeq_{\mathrm{I}}$ being a refinement of $\preccurlyeq$. For a weak refinement one needs to replace $<$ by $\leq$.

Proof: Suppose $A \not\preccurlyeq \{b\}$. According to Definition 2.2, $(A \cup \{b\}) \prec A$ which implies that $(A \cup \{b\}) \preccurlyeq_{\mathrm{I}} A$ (weak refinement) respectively $(A \cup \{b\}) \prec_{\mathrm{I}} A$ (refinement). Hence, $I(A \cup \{b\}) \leq I(A)$ respectively $I(A \cup \{b\}) < I(A)$ according to (3). ■

For instance, most of the diversity indicators proposed in the literature do not fulfill these necessary conditions and therefore do neither weakly refine nor refine the weak Pareto-dominance relation—if used alone. However, nevertheless these indicators can be useful in combination with indicators inducing (weak) refinements as we will show in Section III-D.

B. Binary Indicators

In contrast to unary indicators, binary quality indicators assign a real value to ordered pairs of sets (A, B) with $A, B \in \Psi$. Assuming that smaller indicator values stand for higher quality, we can define for each binary indicator I a corresponding set preference relation as follows:

$$A \preccurlyeq_{\mathrm{I}} B := (I(A, B) \leq I(B, A)) \tag{4}$$

The relation is not necessarily a preorder, and this property needs to be shown for each specific indicator separately. However, if transitivity and reflexitivity are fulfilled, then $\preccurlyeq_{\mathrm{I}}$ is even a total preorder because either $(I(A, B) \leq I(B, A))$ or $(I(B, A) \leq I(A, B))$ (or both). Consider for instance the coverage indicator [34] which gives the portion of solutions in B to which a weakly preferable solution in A exists. This indicator does not induce a preorder as transitivity cannot be ensured [20].

In the following, we present analogous sufficient and necessary conditions for binary indicators I where $\preccurlyeq_{\mathrm{I}}$ represents a preorder.

Theorem 3.4: Let I be a binary indicator that induces a preorder $\preccurlyeq_{\mathrm{I}}$ on Ψ according to 4. If I satisfies

$$(A \preccurlyeq B) \wedge (B \not\preccurlyeq A) \Rightarrow (I(A, B) < I(B, A))$$

for all $A, B \in \Psi$, then the corresponding preference relation $\preccurlyeq_{\mathrm{I}}$ refines $\preccurlyeq$ according to Definition 2.5. If $<$ is replaced by $\leq$, then $\preccurlyeq_{\mathrm{I}}$ weakly refines $\preccurlyeq$ according to Definition 2.6.

Proof: Consider $A, B \in \Psi$ with $(A \preccurlyeq B) \wedge (B \not\preccurlyeq A)$. If $I(A, B) \leq I(B, A)$, then also $A \preccurlyeq_{\mathrm{I}} B$ according to (4). If $I(A, B) < I(B, A)$, then $I(A, B) \leq I(B, A)$, but $I(B, A) \not\leq I(A, B)$, which implies that $A \preccurlyeq_{\mathrm{I}} B$ and $B \not\preccurlyeq_{\mathrm{I}} A$. ■

Similar to Theorems 3.2 and 3.3, one can provide necessary and sufficient conditions that are only based on the local behavior, i.e., when adding a single element. The proof of the following theorem is given in the Appendix.

Theorem 3.5: Let I be a binary indicator that induces a preorder $\preccurlyeq_{\mathrm{I}}$ on Ψ according to (4), and $\preccurlyeq$ a set preference relation that itself conforms to a preference relation $\preceq$ on its elements (see Definition 2.2). The relation $\preccurlyeq_{\mathrm{I}}$ refines $\preccurlyeq$ if the following two conditions hold for all sets $A \in \Psi$ and solutions b with $\{b\} \in \Psi$:

1) If $A \preccurlyeq \{b\}$ then $I(A \cup \{b\}, A) = I(A, A \cup \{b\})$.
2) If $A \not\preccurlyeq \{b\}$ then $I(A \cup \{b\}, A) < I(A, A \cup \{b\})$.

For weak refinement one needs to replace the relation $<$ by $\leq$ in the second condition. The second condition is necessary for $\preccurlyeq_{\mathrm{I}}$ being a refinement (in case of $<$) or weak refinement (in case of $\leq$) of $\preccurlyeq$.

The binary epsilon indicator [35] is an example of an indicator whose corresponding set preference relation $\preccurlyeq_{I_\epsilon}$ according to (4) is a refinement of $\preccurlyeq_{\mathrm{par}}$. It is defined as follows:

$$I_\epsilon(A, B) = \max_{b \in B} \min_{a \in A} F_\epsilon(a, b) \tag{5}$$

where the we use the distance function $F_\epsilon(a, b)$ between two solutions a and b

$$F_\epsilon(a, b) = \max_{1 \leq i \leq n} (f_i(a) - f_i(b))$$

Informally speaking, $I_\epsilon(A, B)$ denotes the minimum amount one needs to improve every objective value $f_i(a)$ of every solution $a \in A$ such that the resulting set is weakly preferable to B. We will not discuss further properties here and instead refer to [18], [20], [35] for detailed reviews of binary indicators.

Preference information of a decision maker can be considered in the above binary epsilon indicator by extending the concepts described in [28] for pairs of solutions to sets. In order to prefer solutions that are close to a set of reference points $r \in R$ in the objective space, we use the concept of an achievement function with respect to a reference point r

$$s(x, r) = \max_{1 \leq i \leq n} (f_i(x) - r_i)$$

It denotes the amount by which one needs to improve each objective of x such that it just weakly dominates the reference point r. Now we normalize this function to positive values

$$\overline{s}(x, r, \delta) = s(x, r) + \delta - \min_{y \in P} \{s(y, r)\},$$

where the specificity $\delta > 0$ determines the minimal value of the normalized function and P denotes the current set of solutions. Using a set R of reference points, then the preference-based distance can be defined as

$$F_P(a, b) = \frac{F_\epsilon(a, b)}{\min_{r \in R} \{\overline{s}(b, r, \delta)\}}$$

If we use $F_P(a, b)$ in (5) instead of F_ϵ, we obtain the preference-based epsilon indicator $I_P(A, B)$ which also leads to a refinement of $\preccurlyeq_{\mathrm{par}}$. By increasing the value of specificity δ from zero gives us a wider set of solutions close to the reference points in R.

Finally, note that there is a close relationship between unary and binary indicators. On the one hand, for every unary indicator I^1 a corresponding binary indicator I^2 can be defined as $I^2(A,B) := I^1(B) - I^1(A)$; it is easy to show that the property of (weak) refinement transfers from the unary indicator to the binary version. On the other hand, every binary indicator I^2 can be transformed into a unary indicator I^1 by using a reference set R: $I^1(A) := I^2(A,R)$[4]. Here, the refinement property is not necessarily preserved, e.g., the unary versions of the binary epsilon indicators induce only weak refinements, while the original binary indicators induce refinements of the weak Pareto-dominance relation.

C. Refinement Through Set Partitioning

The Pareto-dominance relation $\preccurlyeq_{\text{par}}$ on sets is by definition insensitive to dominated solutions in a set, i.e., whether $A \in \Psi$ weakly dominates $B \in \Psi$ only depends on the corresponding minimal sets: $A \preccurlyeq_{\text{par}} B \Leftrightarrow \text{Min}(A, \preceq_{\text{par}}) \preccurlyeq_{\text{par}} \text{Min}(B, \preceq_{\text{par}})$. The same holds for set preference relations induced by the hypervolume indicator and other popular quality indicators. Nevertheless, prefered solutions may be of importance:

- When a Pareto set approximation is evaluated according to additional knowledge and preferences—which may be hard to formalize and therefore may not be included in the search process—, then prefered solutions can become interesting alternatives for a decision maker.
- When a set preference relation is used within a (evolutionary) multiobjective optimizer to guide the search, it is crucial that prefered solutions are taken into account—for reasons of search efficiency.

Accordingly, the question is how to refine a given set preference relation that only depends on its minimal elements such that also non-minimal solutions are considered.

This issue is strongly related to fitness assignment in MOEAs. Pareto-dominance based MOEAs divide the population into dominance classes which are usually hierarchically organized. For instance, with dominance ranking [15] individuals which are dominated by the same number of population members are grouped into one dominance class; with nondominated sorting [17], [27], the minimal elements are grouped into the first dominance class, and the other classes are determined by recursively applying this classification scheme to the remaining population members. The underlying idea can be generalized to arbitrary set preference relations. To this end, we introduce the notion of a set partitioning function.

Definition 3.6: A set partitioning function part is a function $\mathsf{part} : \Psi \times \mathbb{N} \to \Psi$ such for all $A \in \Psi$ a sequence $(A_1, A_2, \ldots, A_l, \ldots)$ of subsets is determined with $A_i := \mathsf{part}(A,i)$ and it holds

1) $\forall 1 \le i \le l : A_i \subseteq A$ and $\forall i > l : A_i = \emptyset$
2) $\forall 1 \le i < j \le l+1 : A_i \supset A_j$

A set partitioning function indirectly creates l non-empty partitions; the ith partition P_i is defined by the set difference between the ith and the $(i+1)$th subset: $P_i = \mathsf{part}(A,i) \setminus \mathsf{part}(A,i+1)$. By construction, the induced partitions are disjoint. For instance, the following two set partitioning functions resemble the concepts of dominance rank and nondominated sorting:

$$\mathsf{rankpart}(A,i) := \{a \in A : |\{b \in A : b \prec a\}| \ge i-1\}$$

$$\mathsf{minpart}(A,i) := \begin{cases} A & \text{if } i = 1 \\ \mathsf{minpart}(A,i-1) \setminus & \\ \text{Min}(\mathsf{minpart}(A,i-1), \preceq) & \text{else} \end{cases}$$

where $\preceq$ is the solution-based preference relation under consideration. The second function, $\mathsf{minpart}$, yields a partitioning such that $P_1 \prec P_2 \prec \ldots \prec P_l$ holds; this is demonstrated in Fig. 12.

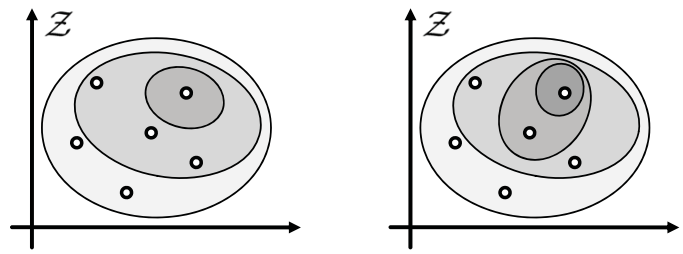


Fig. 12. Illustration of two set partitioning functions, here based on weak Pareto-dominance: $\mathsf{minpart}$ (left) and $\mathsf{rankpart}$ (right). The light-shaded areas stand for the first subsets (A_1) and the darkest areas represent the lth subsets (A_3 left and A_4 right). As to the resulting partitions, on the left holds $P_1 \prec_{\text{par}} P_2 \prec_{\text{par}} P_3$, while on the right $P_1 \prec_{\text{par}} P_i$ for $2 \le i \le 4$, $P_3 \prec_{\text{par}} P_4$, and $P_2 \,||_{\text{par}}\, P_3$ as well as $P_2 \,||_{\text{par}}\, P_4$.

Now, given a set partitioning function part one can construct set preference relations that only refer to specific partitions of two sets $A, B \in \Psi$. By concatenating these relations, one then obtains a sequence of relations that induces a set preference relation according to Def. 2.7.

Definition 3.7: Let $\preccurlyeq$ be a set preference relation and part a set partitioning function. For a fixed l, the partition-based extension of $\preccurlyeq$ is defined as the relation $\preccurlyeq^{\mathsf{part}[l]} := \preccurlyeq_S$ where S is the sequence $(\preccurlyeq^1_{\mathsf{part}}, \preccurlyeq^2_{\mathsf{part}}, \ldots, \preccurlyeq^l_{\mathsf{part}})$ of preference relations with

$$A \preccurlyeq^i_{\mathsf{part}} B \quad :\Leftrightarrow \quad (\mathsf{part}(A,i) \setminus \mathsf{part}(A,i+1)) \preccurlyeq (\mathsf{part}(B,i) \setminus \mathsf{part}(B,i+1))$$

A partition-based extension of a set preference relation $\preccurlyeq$ basically means that $\preccurlyeq$ is successively applied to the hierarchy of partitions defined by the corresponding set partition function. Given $A, B \in \Psi$, first the two first partitions of A and B are compared based on $\preccurlyeq$; if the comparison yields equivalence, then the two second partitions are compared and so forth. This principle reflects the general fitness assignment strategy used in most MOEAs.

One important requirement for such a partition-based extension is that $\preccurlyeq^{\mathsf{part}[l]}$ refines $\preccurlyeq$. Provided that $\preccurlyeq$ only depends on the minimal elements in the sets, both $\mathsf{rankpart}$ and $\mathsf{minpart}$ induce refinements. The argument is simply that $\preccurlyeq^1_{\mathsf{part}}$ is the same as $\preccurlyeq$ because the first partition corresponds for both functions to the set of minimal elements; that means $\preccurlyeq^1_{\mathsf{part}}$

[4]Usually, instead of a reference set of solutions a reference set of objective vectors is given. This requires a slight modication of the indicator.

is a refinement of $\preccurlyeq$. Furthermore, all $\preccurlyeq^i_{\mathsf{part}}$ are preorders. Applying Theorem 2.10 leads to the above statement. For minpart, it can even be shown that $\preccurlyeq^{\mathsf{minpart}[l+1]}$ is a refinement of $\preccurlyeq^{\mathsf{minpart}[l]}$ for all $l > 0$.

In the following, we will mainly consider the set partitioning function minpart and refer to it as minimum elements partitioning (or nondominated sorting in the case of Pareto-dominance). It induces a natural partitioning into sets of minimal elements where the partitions are linearly ordered according to strict preferability. For reasons of simplicity, we may omit the l when refering to the partition-extended relation $\preccurlyeq^{\mathsf{minpart}[l]}$; it is usually set equal to the population size of the evolutionary algorithm used.

D. Combined Preference Relations

The issue of prefered (dominated) solutions in a set $A \in \Psi$ cannot only be addressed by means of set partitioning functions, but also by using multiple indicators in sequence. For instance, one could use the hypervolumen indicator I_H (to assess the minimal elements in A) in combination with a diversity indicator I_D (to assess the non-minimal elements in A); according to Theorem 2.10, the set preference relation $\preccurlyeq_{H,D}$ given by the sequence $(\preccurlyeq_H, \preccurlyeq_D)$ is a proper refinement of weak Pareto-dominance since $\preccurlyeq_H$ is a refinement (see above) and $\preccurlyeq_D$ is a preorder. Thus, such combinations can be useful to increase the sensitivity of a relation (meaning that the number of unidirectional edges $A \prec B$ is increased); however, there are several further reasons for combining indicators:

- To embed indicators which only induce weak refinements or not refinement at all into set preference relations representing refinements.
- To reduce computation effort: for instance, the hypervolume indicator is expensive to compute; by using it only occasionally at the end of a sequence of indicators, a considerable amount of computation time may be saved.
- To include heuristic information to guide the search: a set preference relation that reflects the user preferences not necessarily provides information for an effective search; therefore, it may be augmented by optimization-related aspects like diversity through additional dedicated indicators.

In the following, we present some examples for combined set preference relations that illustrate different application scenarios. All of these relations are refinements of the set preference relation $\preccurlyeq_{\text{par}}$.

1) The first combination is based on the unary epsilon indicator $I_{\epsilon 1}$ with a reference set R in objective space which is defined as $I_{\epsilon 1}(A) = E(A, R)$ with

$$E(A, R) = \max_{r \in R} \min_{a \in A} \epsilon(a, r)$$

where

$$\epsilon(a, r) = \max\{f_i(a) - r_i \mid 1 \leq i \leq n\}$$

and r_i is the ith component of the objective vector r. Since this indicator induces only a weak refinement of the weak Pareto-dominance relation $\preccurlyeq_{\text{par}}$, we will use the hypervolumen indicator to distinguish between sets indifferent with respect $I_{\epsilon 1}$. The resulting set preference relation is denoted as $\preccurlyeq_{\epsilon 1, H}$; it is a refinement of $\preccurlyeq_{\text{par}}$.

2) The second combination uses the R_2 indicator proposed in [18] for which the following definition is used here:

$$I_{R2}(A) = R_2(A, R) = \frac{\sum_{\lambda \in \Lambda} u^*(\lambda, R) - u^*(\lambda, f(A))}{|\Lambda|}$$

where the function u^* is a utility function based on the weighted Tchebycheff function

$$u^*(\lambda, T) = -\min_{z \in T} \max_{1 \leq j \leq n} \lambda_j |z^*_j - z_j|$$

and Λ is a set of weight vectors $\lambda \in \mathbb{R}^n$, $R \subset \mathcal{Z}$ is a reference set, and $z^* \in \mathcal{Z}$ is a reference point. In this paper, we will set $R = \{z^*\}$. Also the R_2 indicator provides only a weak refinement; as before, the hypervolume indicator is added in order to achieve a refinement. This set preference relation will be denoted as $\preccurlyeq_{R2,H}$.

3) The previous two indicator combinations will be used together with a set partitioning function. To demonstrate that the partitioning can also be accomlished by indicators, we propose the following sequence of indicators $S = (I_H, I_C, I_D)$ where I_C measures the largest distance of a solution to the closest minimal element in a set and I_D reflects the diversity of the solutions in the objective space. The latter two indicators, which both do not induce weak refinements of $\preccurlyeq_{\text{par}}$, are defined as follows:

$$I_C(A) = \max_{a \in A} \min_{b \in \mathsf{Min}(A, \preceq)} dist(f(a), f(b))$$

and

$$I_D(A) = \max_{a \in A} \left(\frac{1}{nn_1(a, A \setminus \{a\})} + \frac{1}{nn_2(a, A \setminus \{a\})} \right)$$

with

$$nn_1(a, B) = \min_{b \in B} dist(f(a), f(b))$$

$$nn_2(a, B) = \max_{c \in B} \min_{b \in B \setminus \{c\}} dist(f(a), f(b))$$

where $nn_1(a, B)$ gives the smallest and $nn_2(a, B)$ the second smallest distance of a to any solution in B. For the distance function $dist(z^1, z^2)$, Euclidean distance is used here, i.e., $dist(z^1, z^2) = \sqrt{\sum_{1 \leq i \leq n} (z^1_i - z^2_i)^2}$. The I_C indicator resembles the generational distance measure proposed in [30] and I_D resembles the nearest neighbor niching mechanism in SPEA2 [33]. We will refer to the overall set preference relation as $\preccurlyeq_{H,C,D}$. According to Theorem 2.10, $\preccurlyeq_{H,C,D}$ is a refinement of $\preccurlyeq_{\text{par}}$.

It is worth mentioning that it is also possible to combine a non-total preorder such as $\preccurlyeq_{\text{par}}$ with total orders differently to the principle suggested in Def. 2.7. As has been pointed out, see e.g. Fig. 7, convergence may fail if an optimization is not based on a preorder or if the underlying preorder is not

a refinement. The following example illustrates why density-based MOEAs such as NSGA-II and SPEA2 show cyclic behavior, see [24], in particular, when the population mainly contains incomparable solutions, e.g., when being close to the trade-off surface.

For instance, let I be a unary indicator, then one may define a set preference relation $\preccurlyeq_{\text{par},I}$ as follows with $A, B \in \Psi$:

$$A \preccurlyeq_{\text{par},I} B \;:\Leftrightarrow (A \preccurlyeq_{\text{par}} B) \vee ((A \,||_{\text{par}}\, B) \wedge (A \preccurlyeq_I B))$$

Now, consider a unary diversity indicator, e.g., I_D as defined above; this type of indicator usually does not induce a weak refinement. The resulting set preference relation $\preccurlyeq_{\text{par},I}$ is not a proper preorder as Fig. 13 demonstrates: transitivity is violated, i.e., $A \preccurlyeq_{\text{par},I} B$ and $B \preccurlyeq_{\text{par},I} C$, but $A \not\preccurlyeq_{\text{par},I} C$. The relation graph of $\preccurlyeq_{\text{par},I}$ contains cycles. However, if I stands for the hypervolumen indicator I_H, then $\preccurlyeq_{\text{par},I}$ is a set preference relation refining $\preccurlyeq_{\text{par}}$; this combination could be useful to reduce computation effort.

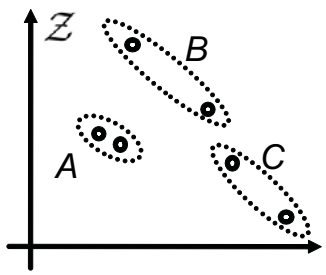


Fig. 13. Three sets are shown in the objective space where $A \preccurlyeq_{\text{par}} B$, $A \,||_{\text{par}}\, C$ and $B \,||_{\text{par}}\, C$. Using a combination of Pareto-dominance and diversity results in a cyclic relation $\preccurlyeq_{\text{par},I}$ with $A \prec_{\text{par},I} B$, $B \prec_{\text{par},I} C$, and $C \prec_{\text{par},I} A$.

IV. Multiobjective Optimization Using Set Preference Relations

In the following, we will present a generalized multiobjective optimizer that can search according to arbitrary set preference relations. Section IV-A describes the algorithm, while Section IV-B discusses theoretical aspects of the algorithm, in particular convergence properties.

A. *SPAM - Set Preference Algorithm for Multiobjective Optimization*

The main part of the Set Preference Algorithm for Multiobjective Optimization (SPAM) is given by Algorithm 1. It resembles a standard hill climber with the difference that two new elements of the search space Ψ are created using two types of mutation operators.

Starting with a randomly chosen set $P \in \Psi_m$ of size m, first a random mutation operator is applied to generate another set P'. This operator should be designed such that every element in Ψ could be possibly generated, i.e., the neighborhood is in principle the entire search space. In practice, the operator will usually have little effect on the optimization process; however, its property of exhaustivness is important from a theoretical perspective, in particular to show convergence, cf. Section IV-B.

Second, a heuristic mutation operator is employed. This operator mimics the mating selection, variation, and environmental selection steps as used in most MOEAs. The goal of this operator is to create a third set $P'' \in \Psi$ that is better than P in the context of a predefined set preference relation $\preccurlyeq$. However, since it is heuristic it cannot guarantee to improve P; there may be situations where it is not able to escape local optima of the landscape of the underlying set problem. Finally, P is replaced by P'', if the latter is weakly preferable to the former; otherwise, P is either replaced by P' (if $P' \preccurlyeq P$) or remains unchanged. Note that in the last step, weak preferability ($\preccurlyeq$) and not preferability ($\prec$) needs to be considered in order to allow the algorithm to cross landscape plateaus, cf. [5].

Algorithm 1 SPAM Main Loop

Require: set preference relation $\preccurlyeq$
1: generate initial set P of size m, i.e., randomly choose $A \in \Psi_m$ and set $P \leftarrow A$
2: **while** termination criterion not fulfilled **do**
3: $\quad P' \leftarrow$ *randomSetMutation*(P)
4: $\quad P'' \leftarrow$ *heuristicSetMutation*(P)
5: $\quad$ **if** $P'' \preccurlyeq P$ **then**
6: $\quad\quad P \leftarrow P''$
7: $\quad$ **else if** $P' \preccurlyeq P$ **then**
8: $\quad\quad P \leftarrow P'$
9: **return** P

For the mutation operators, we propose Algorithms 2 and 3. Algorithm 2 (random set mutation) randomly chooses k decision vectors from $\mathcal{X}$ and uses them to replace k elements in P.[5] Algorithm 3 (heuristic set mutation) generalizes the iterative truncation procedures used in NSGA-II [8], SPEA2 [33], and others. First, k new solutions are created based on P; this corresponds to mating selection plus variation in a standard MOEA. While the variation is problem-specific, for mating selection either uniform random selection (used in the following) or fitness-based selection can be used (using the fitness values computed by Algorithm 4 resp. 5). Then, these k solutions are added to P, and finally the resulting set of size $m + k$ is iteratively truncated to size m by removing the solution with the worst fitness values in each step. Here, the fitness value of $a \in P$ reflects the loss in quality for the entire set P if a is deleted: the lower the fitness, the larger the loss.

Algorithm 2 Random Set Mutation

1: **procedure** *randomSetMutation*(P)
2: $\quad$ randomly choose $r_1, \ldots, r_k \in \mathcal{X}$ with $r_i \neq r_j$
3: $\quad$ randomly select $p_1, \ldots, p_k$ from P with $p_i \neq p_j$
4: $\quad P' \leftarrow P \setminus \{p_1, \ldots, p_k\} \cup \{r_1, \ldots, r_k\}$
5: $\quad$ **return** P'

[5]Note that for both mutation operators the same k is used here, although they can be chosen independently. The safe version ($k = m$) for the random mutation operator means that a random walk is carried out on Ψ.

Algorithm 3 Heuristic Set Mutation

1: **procedure** *heuristicSetMutation*(P)
2: generate $r_1, \ldots, r_k \in \mathcal{X}$ based on P
3: $P'' \leftarrow P \cup \{r_1, \ldots, r_k\}$
4: **while** $|P''| > m$ **do**
5: **for all** $a \in P''$ **do**
6: $\delta_a \leftarrow$ *fitnessAssignment(a, P")*
7: choose $p \in P''$ with $\delta_p = \min_{a \in P''} \delta_a$
8: $P'' \leftarrow P'' \setminus \{p\}$
9: **return** P''

To estimate how useful a particular solution $a \in P$ is, Algorithm 4 compares all sets $A_i \subset P$ with $|A_i| = |P| - 1$ to $P \setminus \{a\}$ using the predefined set preference relation $\preccurlyeq$. The fewer sets A_i are weakly preferable to $P \setminus \{a\}$, the better the set $P \setminus \{a\}$ and the less important is a. This procedure has a runtime complexity of $\mathcal{O}((m+k)g)$, where g stands for the runtime needed to compute the preference relation comparisons which usually depends on $m+k$ and the number of objective functions. If unary indicators are used, then fitness assignment can be done faster. Algorithm 5 gives an example where $\preccurlyeq$ is defined by a single unary indicator I; the loss in quality is simply the difference in the indicator value caused by the removal of a. The last scheme is used, e.g., in [21], [14], [31], [19] in combination with the hypervolume indicator; Fig. 14 illustrates the working principles.

Algorithm 4 Fitness Assignment (General Version)

procedure *fitnessAssignment*(a, P'')
$\delta_a \leftarrow 0$
for all $b \in P''$ **do**
if $P'' \setminus \{b\} \preccurlyeq P'' \setminus \{a\}$ **then**
$\delta_a \leftarrow \delta_a + 1$
return δ_a

Algorithm 5 Fitness Assignment (Unary Indicator Version)

procedure *fitnessAssigment*(a, P'')
$\delta_a \leftarrow I(P'' \setminus \{a\}) - I(P'')$
return δ_a

Finally, note that Algorithm 3 is heuristic in nature, in particular the truncation method realized by the while loop at Step 4. Before entering the loop at Step 4, P'' contains exactly $m+k$ solutions. Ideally, that subset $A \subseteq P''$ with $|A| = m$ would be chosen that is weakly prefered by the lowest number of m-element subsets $A_i \subseteq P''$. However, computing the optimal subset A is usually computationally expensive and therefore the iterative truncation procedure represents a greedy strategy to obtain a good approximation of A in reasonable time.

B. Convergence Results

A fundamental question in stochastic optimization is whether and under what conditions a particular search algorithm converges to the optimum given infinite running time resources. Although convergence alone does not necessarily imply that the algorithm is fast and efficient in practice, it is an important and desirable property; in the EMO field, only few convergence results are available to date, e.g., [26], [24], [23].

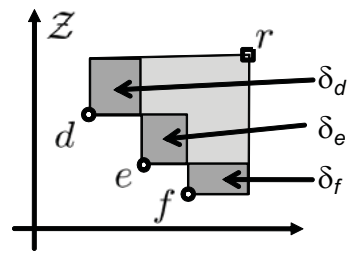


Fig. 14. The figure depicts the fitness values computed according to Algorithms 4 and 5 when using the hypervolume indicator. The light shaded area stands for the volume of the entire three-elements set, the dark shaded areas reflects the volume contributions (fitness values) of the distinct solutions.

Here, we are interested in the properties of the set mutation operators that guarantee convergence of SPAM; in particular, we will investigate the influence of the parameter k that determines the number of newly created solutions per set mutation. Convergence in this paper refers to the limit behavior of an optimization algorithm.

In order to simplify the discussion, let us suppose that the set of solutions $\mathcal{X}$ is finite and that only random set mutation is used, i.e., *heuristicSetMutation*(P) always returns P for the time being. In addition, we have available a preference relation $\preccurlyeq$ on the set Ψ_m of sets containing not more than m solutions, see Section II.

Then, SPAM is called convergent, if the probability that the resulting set P is not minimal, i.e. $P \notin \text{Min}(\Psi_m, \preccurlyeq)$, approaches 0 for $N \to \infty$ where N denotes the number of iterations of the while loop in Algorithm 1. If we can now guarantee the convergence of SPAM, then this simple hill-climbing strategy would be able to solve any multiobjective optimization problem as stated in Section II-B in case of finite but unbounded computation time.

Given Ψ_m, let us consider the underlying set preference relation $\preccurlyeq$ and in particular the corresponding relation graph where there is an edge (A, B) for each pair $A, B \in \Psi_m$ with $A \preccurlyeq B$. As $\preccurlyeq$ is a preorder, there is a directed path from any minimal element $P^* \in \text{Min}(\Psi_m, \preccurlyeq)$ to any node P. In order to be able to reach such an optimal set using SPAM, the chosen preference relation as well as the set mutation operator need to satisfy certain constraints. They both must enable such paths under the condition that each set mutation replaces at most k elements from the current set P in order to generate P'. Note that the optimizer traverses a path in reverse direction as there is an edge from A to B if $A \preccurlyeq B$.

Let $(P^1, P^2, ..., P^i)$ denote the sequence of sets generated by SPAM where P^i denotes the contents of P at the beginning of the ith iteration of the while loop in Algorithm 1. In addition, r^i_j and p^i_j, $1 \leq j \leq k$ denote the elements randomly created resp. selected for removal in Algorithm 2. In particular, P^i is mutated by removing $p^i_1, ..., p^i_k$ and adding $r^i_1, ..., r^i_k$.

Definition 4.1: Let us now suppose that for any initial set P^1 there exists a number N of iterations and p_j^i and r_j^i, $1 \leq j \leq k$, $1 \leq i \leq N$ such that (i) $P^N \in \text{Min}(\Psi_m, \preccurlyeq)$ and (ii) $P^{i+1} \preccurlyeq P^i$ for all $1 \leq i \leq N-1$. Then we say that the given set preference relation $\preccurlyeq$ is k-greedy.

The above property states that if the given set preference relation $\preccurlyeq$ is k-greedy, then there exists a sequence of sets with cardinality not larger than m that starts from any set in Ψ_m and ends at an optimal set in Ψ where at most k elements differ from one set to the next. Therefore, this path could be found by SPAM, if the *randomSetMutation* operator does exactly the requested replacements of elements. In order to make this argument more formal, we need to define properties of the set mutation operator. For example, if never elements of some optimal set P^* are generated, this set could never be reached.

Definition 4.2: Given any k different solutions $s_j \in \mathcal{X}$ and any k different solutions $\hat{p}_j$ in P where $1 \leq j \leq k$. If for an execution of the *randomSetMutation* operator the probability is larger than zero that (i) the selected elements satisfy $p_j = \hat{p}_j$ and (ii) the randomly generated elements satisfy $r_j = s_j$ for all $1 \leq j \leq k$, then the *randomSetMutation* operator is called exhaustive.

In other words, an exhaustive *randomSetMutation* operator replaces any k elements of the current set P by any k elements from $\mathcal{X}$ with a finite probability. Therefore, if the set mutation operator is concatenated sufficiently often for an arbitrary initial set (e.g., *randomSetMutation*(*randomSetMutation*(P))), then any possible set in Ψ_m can be generated.

Theorem 4.3: If the given set preference relation $\preccurlyeq$ is k-greedy and the given *randomSetMutation* operator is exhaustive, then the SPAM optimization algorithm converges.

Proof: Consider a graph whose nodes correspond to all sets $Q \subseteq \mathcal{X}$ with size $|Q| = m$ and the edges of which correspond to all possible executions of the *randomSetMutation* operator. If we remove all edges (i, j) with $Q^j \not\preccurlyeq Q^i$ then each path in the graph corresponds to a feasible execution of SPAM. If the set preference relation $\preccurlyeq$ is k-greedy and the *randomSetMutation* operator is exhaustive, then there exists a path with finite probability from any node to a node with $Q \in \text{Min}(\Psi_m, \preccurlyeq)$.

Because of the finiteness of $\mathcal{X}$, SPAM will not converge iff for some execution we have $P^i \equiv P^{i+1}$ and $P^* \prec P^{i+1}$ for all $i \geq K$ with some constant K and for some minimal element $P^* \in \text{Min}(\Psi_m, \preccurlyeq)$. During such an execution, a subset of nodes is visited infinitely often. Let us now suppose that SPAM does not converge. Then none of the infinitely visited nodes has an outgoing edge (i, j) with $Q^j \prec Q^i$. This contradicts the assumption that there exists a path with finite probability from any node to an optimal node Q with $Q \in \text{Min}(\Psi_m, \preccurlyeq)$. ■

Theorem 4.4: Every set preference relation $\preccurlyeq$ is m-greedy where m denotes the maximum size of the sets of solutions, i.e., Ψ_m is the universe of the sets under consideration.

Proof: The set preference relation $\preccurlyeq$ is a preorder. Therefore, there exists a path from any initial set P^1 to an optimal set P^N in Def. 4.1. The intermediate populations $P^2, ..., P^N$ can be generated by exchanging at most all m solutions from one set to the next. As the set preference relation is m-greedy, the elements p_j^i and r_j^i can be chosen correspondingly. ■

The last theorem leads to the fact that SPAM can be used with any set preference relation resp. quality indicator inducing a preorder if we use $k = m$. Whereas this statement may be of theoretical value, SPAM will be of no practical use for large m if only the *randomSetMutation* operator is applied; the reason is that the probability to determine a better set by exchanging randomly a large number of elements will be very small. Therefore, the *heursticSetMutation* operator is used in order to obtain improved sets with a high probability; however, this operator *in general* is not exhaustive, i.e., it may not possible to follow a path to an optimal set $P^* \in \text{Min}(\Psi_m, \preccurlyeq)$. Therefore, if the used *randomSetMutation* operator is exhaustive, i.e. it generates any possible k-neighbor with a finite probability, then SPAM converges to an optimal solution.

Finally, one may ask whether smaller values for k with $k < m$ are possible while still guaranteeing convergence. This may be desirable from a practical perspective since the generation of new solutions in $\mathcal{X}$ would thereby be tighter linked to the update of P; several indicator-based MOEAs basically use set mutation operators where k equals 1 [14], [19]. The answer clearly depends on the set preference relation under consideration. It can be shown that the set preference relations induced by the epsilon indicator and the hypervolume indicator are in general not 1-greedy; the proofs can be found in the appendix.

Corollary 4.5: The set preference relation $\preccurlyeq_{\epsilon 1}$ induced by the unary epsilon indicator $I_{\epsilon 1}$ is not 1-greedy.

Corollary 4.6: The set preference relation $\preccurlyeq_H$ induced by the hypervolume indicator I_H is not 1-greedy.

That means whenever theses indicators are used within a set preference relation incorporated in SPAM, then there are cases where SPAM will not convergence if $k = 1$; this also holds for any other MOEA that is replaces only a single solution in the current population and is based on these indicators. Whether other values for k with $1 < k < m$ are sufficient to guarantee convergence for these indicators is an open research issue.

V. Experimental Validation

This section investigates the practicability of the proposed approach. The main questions are: (i) can different user preferences be expressed in terms of set preference relations, (ii) is it feasible to use a general search algorithm for arbitrary set preference relations, i.e., is SPAM effective in finding appropriate sets, and (iii) how well are set preference relations suited to guide the optimization process? However, the purpose is not to carry out a performance comparison of SPAM to existing MOEAs; the separation of user preferences and search algorithm is the focus of our study.

TABLE I
OVERVIEW OF THE SET PREFERENCE RELATIONS USED IN THE EXPERIMENTAL STUDIES; FOR DETAILS, SEE SECTION III.

$\preccurlyeq_H^{\text{minpart}}$	hypervolume indicator I_H with reference point $(12, 12)$ resp. $(12, 12, 12, 12, 12)$ and minimum elements partitioning
$\preccurlyeq_{P,H}^{\text{minpart}}$	preference-based quality indicator I_P with with specificity $\delta = 0.001$ and two reference points $g^1 = (0.2, 0.8)$ resp. $(0.2, 0.8, 0.8, 0.8, 0.8)$, $g^2 = (0.7, 0.4)$ resp. $(0.7, 0.4, 0.4, 0.4, 0.4)$, followed by the hypervolume indicator I_H with reference point $(12, 12)$ resp. $(12, 12, 12, 12, 12)$; in addition, minimum elements partitioning is used
$\preccurlyeq_{R2,H}^{\text{minpart}}$	R_2 indicator I_{R2} with reference set $B = \{(0, 0)\}$ and $\Lambda = \{(0, 1)$,$(0.01, 0.99)$,$\ldots$,$(0.3, 0.7)$,$(0.7, 0.3)$,$(0.71, 0.29)$,$\ldots$,$(1, 0)\}$ in the case of two objectives[6] ($\|\Lambda\| = 62$), followed by hypervolume indicator I_H with reference point $(12, 12)$ resp. $(12, 12, 12, 12, 12)$; in addition, minimum elements partitioning is used
$\preccurlyeq_{\epsilon 1,H}^{\text{minpart}}$	unary (additive) epsilon indicator $I_{\epsilon 1}$ with reference set $B = \{(k \cdot 0.002, 0.8 - k \cdot 0.004)\,;\, k \in \{0, 1, \ldots, 100\}\}$ resp. $B = \{(k \cdot 0.002, 0.8 - k \cdot 0.004, 0.8 - k \cdot 0.004, 0.8 - k \cdot 0.004, 0.8 - k \cdot 0.004)\,;\, k \in \{0, 1, \ldots, 100\}\}$, followed by the hypervolume indicator I_H with reference point $(12, 12)$ resp. $(12, 12, 12, 12, 12)$; in addition, minimum elements partitioning is used
$\preccurlyeq_{\epsilon}^{\text{minpart}}$	binary (additive) ϵ-indicator I_ϵ combined with minimum elements partitioning
$\preccurlyeq_{H,C,D}$	unary hypervolume indicator I_H with reference point $(12, 12)$ resp. $(12, 12, 12, 12, 12)$ followed by the distance-to-front indicator I_C (maximum distance of any solution to the closest front member) and the diversity indicator I_D (kth-nearest neighbor approach)
$\preccurlyeq_{D}^{\text{minpart}}$	diversity indicator I_D (kth-nearest neighbor approach) combined with minimum elements partitioning

A. Comparison Methodology

In the following, we consider different set preference relations for integration in SPAM; they have been discussed in Section III and are listed in Table I. All of them except of the last one are refinements of the set dominance relation $\preccurlyeq_{\text{par}}$; the relation $\preccurlyeq_D^{\text{minpart}}$ is just used for the purpose of mimicking the behavior of dominance and density based MOEAs such as NSGA-II and SPEA2. As reference algorithms, NSGA-II [8] and IBEA[7] [32] are used; in the visual comparisons also SPEA2 [33] is included.

In order to make statements about the effectiveness of the algorithms considered, one needs to assess the generated Pareto set approximations with regard to the set preference relation under consideration. We suggest the use of the Mann-Whitney U test to compare multiple outcomes of one algorithm with multiple outcomes of another algorithm. This is possible since all set preference relations considered in this paper are total preorders; otherwise, the approach proposed in [22] can be applied. Thereby, one can obtain statements about whether either algorithm yields significantly better results for a specified set preference relation.

In detail, the statistical testing is carried as follows. Assuming two optimizers OA and OB, first all Pareto-set approximations generated by OA are pairwisely compared to all Pareto-set approximations generated by OB. If, e.g., 30 runs have been performed for each algorithm, then overall 900 comparisons are made. Now, let A and B be two Pareto-set approximations resulting from OA respectively OB; then, we consider set A as better than set B with respect to the set preference relation $\preccurlyeq$, if $A \prec B$ holds. By counting the number of comparisons where the set of OA is better than the corresponding set of OB, one obtains the test statistics U; doing the same for OB gives U' which reflects the number of cases where OB yields a better outcome. The bigger U is compared to U', the better algorithm OA is geared towards the test relation $\preccurlyeq$ regarding OB.

[6]The weight combinations used for the five-objective problem instances are provided in the appendix; in this case, the considered reference set was $B = \{(0, 0, 0, 0, 0)\}$

[7]With parameters $\kappa = 0.05$ and $\rho = 1.1$.

As long as the entirety of the considered sets can be regarded as a large sample (e.g., 30 runs per algorithm), one can use the one-tailed normal approximation to calculate the significance of the test statistics U, correcting the variance for ties. Furthermore, multiple testing issues need to be taken into account when comparing multiple algorithms with each other; here, the significance levels are Bonferroni corrected.

B. Results

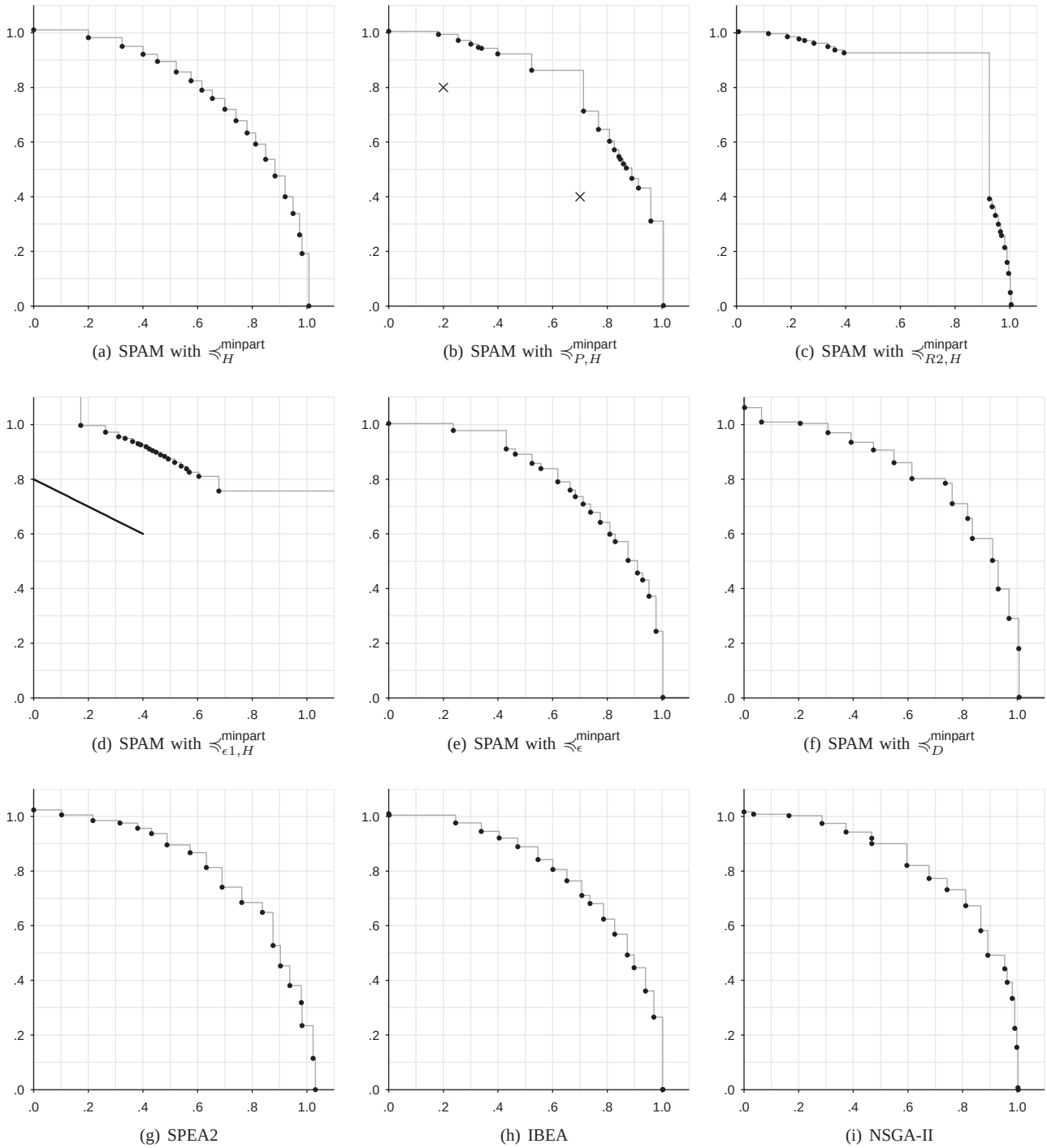
This section provides experimental results for two test problems, namely DTLZ2 and DTLZ5 [12] with 20 decision variables for 2 and 5 objectives. One the one hand, we will provide visual comparisons in order to verify to which extent the formalized user preferences have been achieved. On the other hand, statistical tests are applied to investigate which search strategy is best suited to optimize which user preferences; for each optimizer, 30 have been carried out. The general parameters used in the optimization algorithms are given in Table II.

TABLE II
PARAMETER SETTINGS USED IN SECTION V-B

Parameter	Value
set size / population size m	20*,50**
newly created solutions / offspring individuals k	20*,50**
number of iterations / generations	1000
mutation probability	1
swap probability	0.5
recombination probability	1
η-mutation	20
η-recombination	20
symmetric recombination	false
scaling	false
tournament size	2
mating selection	uniform

* visual comparision, ** statistical testing

1) Visual comparisons: Figure 15 shows the Pareto-set approximations generated by SPAM with the aforementioned set preference relations and by the reference algorithms for the biobjective DTLZ2 problem. The plots well reflect the chosen user preferences: (a) a set maximizing hypervolume, (b) a divided set close to two reference points, (c) focus on the extremes using corresponding weight combinations, (d)



(a) SPAM with $\preccurlyeq_{H}^{\text{minpart}}$ (b) SPAM with $\preccurlyeq_{P,H}^{\text{minpart}}$ (c) SPAM with $\preccurlyeq_{R2,H}^{\text{minpart}}$

(d) SPAM with $\preccurlyeq_{\epsilon 1,H}^{\text{minpart}}$ (e) SPAM with $\preccurlyeq_{\epsilon}^{\text{minpart}}$ (f) SPAM with $\preccurlyeq_{D}^{\text{minpart}}$

(g) SPEA2 (h) IBEA (i) NSGA-II

Fig. 15. Pareto-set approximations found after 1000 generations on a biobjective DTLZ2 problem for a set size / population size of $m = 20$. All algorithms were started with the same initial set / population.

TABLE III

PAIRWISE STATISTICAL COMPARISON OF 30 RUNS PER ALGORITHM AFTER 1000 GENERATIONS. IN THE NOTATION $U : U'$, U (RESP. U') STANDS FOR THE NUMBER OF TIMES A SET GENERATED BY ALGORITHM A (RESP. B) BEATS A SET OF ALGORITHM B (RESP. A) WITH REGARD TO THE TEST RELATION ASSOCIATED WITH THE CORRESPONDING ROW. A STAR NEXT TO THESE NUMBERS INDICATES A SIGNIFICANT DIFFERENCE, THE FEW CASES WHERE THIS WAS NOT THE CASE ARE SHOWN IN BOLD.

(a) 2-dimensional DTLZ2

alg. A \ alg B.		SPAM with set preference relation …							IBEA	NSGA-II	test relation
		$\preccurlyeq^{\text{minpart}}_{P,H}$	$\preccurlyeq^{\text{minpart}}_{H}$	$\preccurlyeq^{\text{minpart}}_{R2,H}$	$\preccurlyeq^{\text{minpart}}_{\epsilon 1,H}$	$\preccurlyeq^{\text{minpart}}_{\epsilon}$	$\preccurlyeq_{H,C,D}$	$\preccurlyeq^{\text{minpart}}_{D}$			
SPAM with set preference relation …	$\preccurlyeq^{\text{minpart}}_{P,H}$	-	869: 31*	900: 0*	900: 0*	869: 31*	879: 21*	900: 0*	878: 22*	900: 0*	$\preccurlyeq_{P,H}$
	$\preccurlyeq^{\text{minpart}}_{H}$	900: 0*	-	900: 0*	900: 0*	900: 0*	**456:444**	900: 0*	900: 0*	900: 0*	$\preccurlyeq_{H}$
	$\preccurlyeq^{\text{minpart}}_{R2,H}$	900: 0*	900: 0*	-	900: 0*	900: 0*	900: 0*	900: 0*	900: 0*	900: 0*	$\preccurlyeq_{R2,H}$
	$\preccurlyeq^{\text{minpart}}_{\epsilon 1,H}$	900: 0*	900: 0*	900: 0*	-	900: 0*	889: 1*	900: 0*	900: 0*	900: 0*	$\preccurlyeq_{\epsilon 1,H}$
	$\preccurlyeq^{\text{minpart}}_{\epsilon}$	900: 0*	**63:837**	900: 0*	900: 0*	-	**81:819**	899: 1*	**274:626**	896: 4*	$\preccurlyeq_{\epsilon}$
	$\preccurlyeq_{H,C,D}$	900: 0*	**444:456**	900: 0*	900: 0*	853: 47*	-	900: 0*	820: 80*	900: 0*	$\preccurlyeq_{H,C,D}$

* preference is significant at the 0.001 level (1-tailed, Bonferroni-adjusted)

(b) 2-dimensional DTLZ5

alg. A \ alg B.		SPAM with set preference relation …							IBEA	NSGA-II	test relation
		$\preccurlyeq^{\text{minpart}}_{P,H}$	$\preccurlyeq^{\text{minpart}}_{H}$	$\preccurlyeq^{\text{minpart}}_{R2,H}$	$\preccurlyeq^{\text{minpart}}_{\epsilon 1,H}$	$\preccurlyeq^{\text{minpart}}_{\epsilon}$	$\preccurlyeq_{H,C,D}$	$\preccurlyeq^{\text{minpart}}_{D}$			
SPAM with set preference relation …	$\preccurlyeq^{\text{minpart}}_{P,H}$	-	897: 3*	900: 0*	900: 0*	893: 7*	891: 9*	900: 0*	887: 13*	900: 0*	$\preccurlyeq_{P,H}$
	$\preccurlyeq^{\text{minpart}}_{H}$	900: 0*	-	900: 0*	900: 0*	900: 0*	**445:455**	900: 0*	900: 0*	900: 0*	$\preccurlyeq_{H}$
	$\preccurlyeq^{\text{minpart}}_{R2,H}$	900: 0*	900: 0*	-	900: 0*	900: 0*	900: 0*	900: 0*	900: 0*	900: 0*	$\preccurlyeq_{R2,H}$
	$\preccurlyeq^{\text{minpart}}_{\epsilon 1,H}$	900: 0*	891: 9*	900: 0*	-	900: 0*	897: 3*	900: 0*	900: 0*	900: 0*	$\preccurlyeq_{\epsilon 1,H}$
	$\preccurlyeq^{\text{minpart}}_{\epsilon}$	896: 4*	**60:840**	900: 0*	900: 0*	-	**57:843**	900: 0*	**349:551**	889: 11*	$\preccurlyeq_{\epsilon}$
	$\preccurlyeq_{H,C,D}$	900: 0*	**455:445**	900: 0*	900: 0*	900: 0*	-	900: 0*	900: 0*	900: 0*	$\preccurlyeq_{H,C,D}$

* preference is significant at the 0.001 level (1-tailed, Bonferroni-adjusted)

(c) 5-dimensional DTLZ2

alg. A \ alg B.		SPAM with set preference relation …							IBEA	NSGA-II	test relation
		$\preccurlyeq^{\text{minpart}}_{P,H}$	$\preccurlyeq^{\text{minpart}}_{H}$	$\preccurlyeq^{\text{minpart}}_{R2,H}$	$\preccurlyeq^{\text{minpart}}_{\epsilon 1,H}$	$\preccurlyeq^{\text{minpart}}_{\epsilon}$	$\preccurlyeq_{H,C,D}$	$\preccurlyeq^{\text{minpart}}_{D}$			
SPAM with set preference relation …	$\preccurlyeq^{\text{minpart}}_{P,H}$	-	817: 83*	900: 0*	900: 0*	896: 4*	864: 36*	900: 0*	900: 0*	900: 0*	$\preccurlyeq_{P,H}$
	$\preccurlyeq^{\text{minpart}}_{H}$	900: 0*	-	900: 0*	900: 0*	900: 0*	**404:496**	900: 0*	895: 5*	900: 0*	$\preccurlyeq_{H}$
	$\preccurlyeq^{\text{minpart}}_{R2,H}$	900: 0*	900: 0*	-	900: 0*	900: 0*	900: 0*	900: 0*	900: 0*	900: 0*	$\preccurlyeq_{R2,H}$
	$\preccurlyeq^{\text{minpart}}_{\epsilon 1,H}$	900: 0*	895: 5*	870: 30*	-	894: 6*	895: 5*	900: 0*	895: 5*	900: 0*	$\preccurlyeq_{\epsilon 1,H}$
	$\preccurlyeq^{\text{minpart}}_{\epsilon}$	900: 0*	900: 0*	900: 0*	899: 1*	-	900: 0*	900: 0*	**454:446**	900: 0*	$\preccurlyeq_{\epsilon}$
	$\preccurlyeq_{H,C,D}$	900: 0*	**496:404**	900: 0*	900: 0*	900: 0*	-	900: 0*	900: 0*	900: 0*	$\preccurlyeq_{H,C,D}$

* preference is significant at the 0.001 level (1-tailed, Bonferroni-adjusted)

(d) 5-dimensional DTLZ5

alg. A \ alg B.		SPAM with set preference relation …							IBEA	NSGA-II	test relation
		$\preccurlyeq^{\text{minpart}}_{P,H}$	$\preccurlyeq^{\text{minpart}}_{H}$	$\preccurlyeq^{\text{minpart}}_{R2,H}$	$\preccurlyeq^{\text{minpart}}_{\epsilon 1,H}$	$\preccurlyeq^{\text{minpart}}_{\epsilon}$	$\preccurlyeq_{H,C,D}$	$\preccurlyeq^{\text{minpart}}_{D}$			
SPAM with set preference relation …	$\preccurlyeq^{\text{minpart}}_{P,H}$	-	866: 34*	900: 0*	900: 0*	898: 2*	858: 42*	900: 0*	900: 0*	900: 0*	$\preccurlyeq_{P,H}$
	$\preccurlyeq^{\text{minpart}}_{H}$	900: 0*	-	900: 0*	900: 0*	900: 0*	**455:445**	900: 0*	900: 0*	900: 0*	$\preccurlyeq_{H}$
	$\preccurlyeq^{\text{minpart}}_{R2,H}$	900: 0*	900: 0*	-	900: 0*	900: 0*	900: 0*	900: 0*	900: 0*	900: 0*	$\preccurlyeq_{R2,H}$
	$\preccurlyeq^{\text{minpart}}_{\epsilon 1,H}$	879: 21*	**618:282**	900: 0*	-	856: 44*	**626:274**	900: 0*	893: 7*	900: 0*	$\preccurlyeq_{\epsilon 1,H}$
	$\preccurlyeq^{\text{minpart}}_{\epsilon}$	900: 0*	824: 76*	900: 0*	900: 0*	-	782:118*	895: 5*	**146:754**	900: 0*	$\preccurlyeq_{\epsilon}$
	$\preccurlyeq_{H,C,D}$	900: 0*	**445:455**	900: 0*	900: 0*	900: 0*	-	900: 0*	900: 0*	900: 0*	$\preccurlyeq_{H,C,D}$

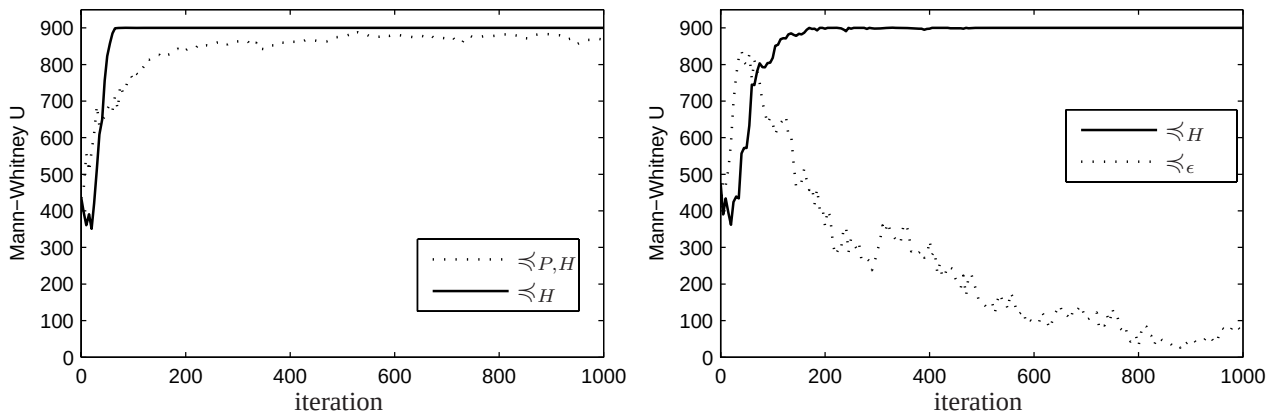
* preference is significant at the 0.001 level (1-tailed, Bonferroni-adjusted)

closeness to a given reference set, (e) a set minimizing the epsilon value with regard to the Pareto-optimal set, and (f) a uniformly distributed set of solutions. This demonstrates that SPAM is in principle capable of optimizing towards the user preferences that are encoded in the corresponding set preference relation.

It can also be seen that the density-based approaches by NSGA-II and SPEA2 can be imitated by using a corresponding diversity indicator—although this is not the goal of this study. Furthermore, the results obtained by IBEA and SPAM with $\preccurlyeq^{\text{minpart}}_{\epsilon}$ are similar, yet there are some differences; we will discuss this further below.

2) Usefulness for Search: After having seen the proof-of-principle results for single runs, we now investigate the question of how effective SPAM is in optimizing a given set preference relation $\preccurlyeq$, i.e., how specific the optimization process is. The hypothesis is that SPAM used in combination with a specific $\preccurlyeq_A$ (let us say SPAM-A) yields better Pareto set approximations than if used with any other set preference relation $\preccurlyeq_B$ (let us say SPAM-B)—better here means with respect to $\preccurlyeq_A$. Ideally, for every set A generated by SPAM-A and every set B generated by SPAM-B, it would hold $A \preccurlyeq_A B$ or even $A \prec_A B$. Clearly, this describes an ideal situtation. A set preference relation that is well suited for

Fig. 16. Each figures compares two SPAM variants for the biobjective DTLZ2 problem: SPAM with $\preccurlyeq_H^{\text{minpart}}$ versus SPAM with $\preccurlyeq_{P,H}^{\text{minpart}}$ (left) and SPAM with $\preccurlyeq_H^{\text{minpart}}$ versus SPAM with $\preccurlyeq_\epsilon^{\text{minpart}}$. Per figure, both algorithms are compared with respect to both set preference relations (without set partitioning). The solid line on the left gives for each iteration, the number of cases (30 runs versus 30 runs) where SPAM with $\preccurlyeq_H^{\text{minpart}}$ is better than SPAM with $\preccurlyeq_{P,H}^{\text{minpart}}$ with respect to $\preccurlyeq_H$. The dotted line on the left shows how often SPAM with $\preccurlyeq_{P,H}^{\text{minpart}}$ is better than SPAM with $\preccurlyeq_H^{\text{minpart}}$ regarding $\preccurlyeq_{P,H}$ over time. The right figure provides the same where $\preccurlyeq_{P,H}$ is replaced by $\preccurlyeq_\epsilon$.



(a) SPAM using $\preccurlyeq_{P,H}^{\text{minpart}}$ and $\preccurlyeq_H^{\text{minpart}}$

(b) SPAM using $\preccurlyeq_\epsilon^{\text{minpart}}$ and $\preccurlyeq_H^{\text{minpart}}$

representing certain preferences may not be well suited for search per se, cf. Section III-D; for instance, when using a single indicator such as the hypervolume indicator refinement through set partitioning is important for effective search.

To this end, we statistically compare all algorithmic variants with each other with respect to the six refinements listed in Table I. Note that set partitioning is only used for search, not for the comparisons. The outcomes of the pairwise comparisons after Bonferroni correction are given in Table III. With only few exceptions, the above hypothesis is confirmed: using $\preccurlyeq_A$ in SPAM yields the best Pareto-set approximations with regard to $\preccurlyeq_A$, independently of the problem and the number of objectives under consideration. These results are highly significant at a significance level of 0.001.

Concerning the exceptions, first it can be noticed that there is no significant difference between $\preccurlyeq_H^{\text{minpart}}$ and $\preccurlyeq_{H,C,D}$ when used in SPAM—both times, the hypervolume indicator value is optimized. This actually confirms the assumption that set partitioning can be replaced by a corresponding sequence of quality indicators. Second, the algorithm based on the set preference relation $\preccurlyeq_\epsilon$ using the binary epsilon indicator performs slighlty worse than IBEA with respect to $\preccurlyeq_\epsilon$. This is not suprising since IBEA has been designed mainly for the epsilon indicator and exploits certain characteristics; for instance, all population members are compared to each other and not only those in the current front. In addition, SPAM with the binary epsilon indicator performs significantly worse than SPAM with any of the two hypervolume-based relations $\preccurlyeq_H^{\text{minpart}}$ and $\preccurlyeq_{H,C,D}$ in the case of two objectives. This may indicate that the binary epsilon indicator is not sensitive enough to differentiate between small improvements. That means that in Step 7 of Algorithm 3 too many solutions may achieve the minimum δ-value, and therefore a choice needs to be done at random.

To see how the differences develop over the course of time, Fig. 16 compares selected SPAM variants with each other and provides the test statistics for each iteration. As can be seen on the left, already after 150 iterations the differences become highly significant when comparing SPAM with $\preccurlyeq_H^{\text{minpart}}$ and SPAM with $\preccurlyeq_{P,H}^{\text{minpart}}$. In Fig. 16 on the right, it can be observed that SPAM with the binary epsilon indicator is—after a competetive starting phase—soon overtaken by SPAM with the hypervolume indicator. As already mentioned, this may reflect certain insensitivities of the epsilon indicator; however, these results only hold for the biobjective case here and further investigations are needed to make a profound statement.

3) Running Time Issues: Last, we investigate the running time of SPAM, particularly in the light of the fitness assignment procedure chosen (Alg. 4 versus Alg. 5). Clearly, the running time of SPAM heavily depends on the set preference relation used. For instance, the hypervolume-based set preference relations induce considerably more computation time—especially in higher dimensions—than the relation based on the epsilon indicator. This is reflected by the figures provided in Table IV. However, the influence of the fitness assignment algorithms is even stronger: the use of Alg. 4 slows down the search by a factor of 100 in comparison to Alg. 5.

TABLE IV
AVERAGED RUNNING TIME IN SECONDS PER ITERATION FOR SPAM USING THE GENERAL FITNESS ASSIGNMENT PROCEDURE (ALG. 4) AS WELL AS THE PROCEDURE DEDICATED TO UNARY INDICATORS (ALG. 5). THE COMPARISON ARE PROVIDED FOR THE TWO SET PREFERENCE RELATIONS $\preccurlyeq_{\epsilon 1}^{\text{MINPART}}$ AND $\preccurlyeq_H^{\text{MINPART}}$.

algorithm	2d	5d
SPAM using Alg. 4 and $\preccurlyeq_{\epsilon 1}^{\text{minpart}}$	0.0601s	24.8660s
SPAM using Alg. 5 and $\preccurlyeq_{\epsilon 1}^{\text{minpart}}$	0.0055s	0.2128s
SPAM using Alg. 4 and $\preccurlyeq_H^{\text{minpart}}$	0.0170s	253.8699s
SPAM using Alg. 5 and $\preccurlyeq_H^{\text{minpart}}$	0.0097s	1.9722s

VI. Conclusions

In this paper, we have discussed EMO from a single-objective perspective that is centered around set preference relations and based on the following three observations:

1) The result of an MOEA run is usually a set of trade-off solutions representing a Pareto set approximation;
2) Most existing MOEAs can be regarded as hill climbers—usually (1,1)-strategies—on set problems;
3) Most existing MOEAs are (implicitly) based on set preference information.

When applying an evolutionary algorithm to the problem of approximating the Pareto-optimal set, the population itself can be regarded as the current Pareto set approximation. The subsequent application of mating selection, variation, and environmental selection heuristically produces a new Pareto set approximation that—in the ideal case—is better than the previous one. In the light of the underlying set problem, the population represents a single element of the search space which is in each iteration replaced by another element of the search space. Consequently, selection and variation can be regarded as a mutation operator on populations resp. sets. Somewhat simplified, one may say that a classical MOEA used to approximate the Pareto-optimal set is a $(1,1)$-strategy on a set problem. Furthermore, MOEAs are usually not preference-free. The main advantage of generating methods such as MOEAs is that the objectives do not need to be aggregated or ranked a priori; but nevertheless preference information is required to guide the search, although it is usually weaker and less stringent. In the environmental selection step, for instance, an MOEA has to choose a subset of individuals from the parents and the offspring which constitutes the next Pareto set approximation. To this end, the algorithm needs to know the criteria according to which the subset should be selected, in particular when all parents and children are incomparable, i.e., mutually nondominating. That means the generation of a new population usually relies on set preference information.

We here have tried to consequently take this line of thought further and to study how set preference information can be formalized such that a total order on the set of Pareto set approximations results. To this end, we have shown how to construct set preference relations on the basis of quality indicators and provided various examples. Moreover, we have presented a general set preference algorithm for multiobjective optimization (SPAM), which is basically a hill climber and generalizes the concepts found in most modern MOEAs. SPAM can be used in combination with any type of set preference relation and thereby offers full flexibility for the decision maker; furthermore, it can be shown to converge under general conditions. As the experimental results indicate, set preference relations can be used to effectively guide the search, but also to evaluate the outcomes of multiobjective optimizers.

The new perspective that this paper offers is, roughly speaking, considering EMO as using evolutionary algorithms for single-objective set problems in the context of multiobjective optimization. Clearly, there are many open issues, e.g., the design of both specific and arbitrary set preference relations to integrate particular user preferences, especially local preferences. Furthermore, the design of fast search algorithms dedicated to particular set preference relations is of high interest; SPAM provides flexibility, but is rather a baseline algorithm that naturally cannot achieve maximum possible efficiency. And finally, one may think of using a true evolutionary algorithm for set-based multiobjective optimization, one that operates on populations of Pareto set approximations—whether this approach can be beneficial is an open research issue.

References

[1] J. Bader, K. Deb, and E. Zitzler. Faster Hypervolume-based Search using Monte Carlo Sampling. In *Conference on Multiple Criteria Decision Making (MCDM 2008)*. Springer, 2008. to appear.

[2] J. Branke and K. Deb. Integrating User Preferences into Evolutionary Multi-Objective Optimization. Technical Report 2004004, Indian Institute of Technology, Kanput, India, 2004. also published as book chapter in Y. Jin, editor: *Knowledge Incorporation in Evolutionary Computation*, pages 461–477, Springer, 2004.

[3] J. Branke, K. Deb, H. Dierolf, and M. Osswald. Finding Knees in Multi-objective Optimization. In T. P. Runarsson et al., editors, *Conference on Parallel Problem Solving from Nature (PPSN VIII)*, volume 3242 of *LNCS*, pages 722–731. Springer, 2004.

[4] J. Branke, T. Kaußler, and H. Schmeck. Guidance in Evolutionary Multi-Objective Optimization. *Advances in Engineering Software*, 32:499–507, 2001.

[5] D. Brockhoff, T. Friedrich, N. Hebbinghaus, C. Klein, F. Neumann, and E. Zitzler. Do Additional Objectives Make a Problem Harder? In D. Thierens et al., editors, *Genetic and Evolutionary Computation Conference (GECCO 2007)*, pages 765–772, New York, NY, USA, 2007. ACM Press.

[6] D. Cvetković and I. C. Parmee. Preferences and their Application in Evolutionary Multiobjective Optimisation. *IEEE Transactions on Evolutionary Computation*, 6(1):42–57, February 2002.

[7] K. Deb. *Multi-objective optimization using evolutionary algorithms*. Wiley, Chichester, UK, 2001.

[8] K. Deb, S. Agrawal, A. Pratap, and T. Meyarivan. A Fast Elitist Non-Dominated Sorting Genetic Algorithm for Multi-Objective Optimization: NSGA-II. In M. Schoenauer et al., editors, *Conference on Parallel Problem Solving from Nature (PPSN VI)*, volume 1917 of *LNCS*, pages 849–858. Springer, 2000.

[9] K. Deb, S. Chaudhuri, and K. Miettinen. Towards Estimating Nadir Objective Vector Using Evolutionary Approaches. In Maarten Keijzer et al., editors, *Conference on Genetic and Evolutionary Computation (GECCO 2006)*, pages 643–650. ACM Press, 2006.

[10] K. Deb, M. Mohan, and S. Mishra. Evaluating the ϵ-Domination Based Multi-Objective Evolutionary Algorithm for a Quick Computation of Pareto-Optimal Solutions. *Evolutionary Computation*, 13(4):501–525, Winter 2005.

[11] K. Deb and J. Sundar. Reference Point Based Multi-Objective Optimization Using Evolutionary Algorithms. In Maarten Keijzer et al., editors, *Conference on Genetic and Evolutionary Computation (GECCO 2006)*, pages 635–642. ACM Press, 2006.

[12] K. Deb, L. Thiele, M. Laumanns, and E. Zitzler. Scalable Multi-Objective Optimization Test Problems. In *Congress on Evolutionary Computation (CEC 2002)*, pages 825–830. IEEE Press, 2002.

[13] K. Deb and S. Tiwari. Omni-optimizer: A generic evolutionary algorithm for single and multi-objective optimization. *European Journal of Operational Research*, 185(3):1062–1087, 2008.

[14] M. Emmerich, N. Beume, and B. Naujoks. An EMO Algorithm Using the Hypervolume Measure as Selection Criterion. In *Conference on Evolutionary Multi-Criterion Optimization (EMO 2005)*, volume 3410 of *LNCS*, pages 62–76. Springer, 2005.

[15] C. M. Fonseca and Peter J. Fleming. Genetic Algorithms for Multiobjective Optimization: Formulation, Discussion and Generalization. In Stephanie Forrest, editor, *Conference on Genetic Algorithms*, pages 416–423, San Mateo, California, 1993. Morgan Kaufmann.

[16] C. M. Fonseca and Peter J. Fleming. Multiobjective Optimization and Multiple Constraint Handling with Evolutionary Algorithms—Part I: A Unified Formulation. *IEEE Transactions on Systems, Man, and Cybernetics*, 28(1):26–37, 1998.

[17] D. E. Goldberg. *Genetic Algorithms in Search, Optimization, and Machine Learning*. Addison-Wesley, Reading, Massachusetts, 1989.

[18] M. P. Hansen and A. Jaszkiewicz. Evaluating the quality of approximations of the non-dominated set. Technical report, Institute of Mathematical Modeling, Technical University of Denmark, 1998. IMM Technical Report IMM-REP-1998-7.

[19] C. Igel, N. Hansen, and S. Roth. Covariance Matrix Adaptation for Multi-objective Optimization. *Evolutionary Computation*, 15(1):1–28, 2007.

[20] J. Knowles and D. Corne. On Metrics for Comparing Non-Dominated Sets. In *Congress on Evolutionary Computation (CEC 2002)*, pages 711–716, Piscataway, NJ, 2002. IEEE Press.

[21] J. Knowles and D. Corne. Properties of an Adaptive Archiving Algorithm for Storing Nondominated Vectors. *IEEE Transactions on Evolutionary Computation*, 7(2):100–116, 2003.

[22] J. Knowles, L. Thiele, and E. Zitzler. A Tutorial on the Performance Assessment of Stochastic Multiobjective Optimizers. TIK Report 214, Computer Engineering and Networks Laboratory (TIK), ETH Zurich, February 2006.

[23] J. D. Knowles and D. W. Corne. Instance Generators and Test Suites for the Multiobjective Quadratic Assignment Problem. In C. M. Fonseca, P. J. Fleming, E. Zitzler, K. Deb, and L. Thiele, editors, *Evolutionary Multi-Criterion Optimization (EMO 2003)*, volume 2632 of *LNCS*, pages 295–310, Berlin, 2003. Springer.

[24] M. Laumanns, L. Thiele, K. Deb, and E. Zitzler. Combining Convergence and Diversity in Evolutionary Multiobjective Optimization. *Evolutionary Computation*, 10(3):263–282, 2002.

[25] L. Rachmawati and D. Srinivasan. Preference Incorporation in Multiobjective Evolutionary Algorithms: A Survey. In *2006 IEEE Congress on Evolutionary Computation (CEC'2006)*, pages 3385–3391, Vancouver, BC, Canada, July 2006. IEEE Press.

[26] G. Rudolph and A. Agapie. Convergence Properties of Some Multi-Objective Evolutionary Algorithms. In A. Zalzala and R. Eberhart, editors, *Congress on Evolutionary Computation (CEC 2000)*, volume 2, pages 1010–1016. IEEE Press, 2000.

[27] N. Srinivas and Kalyanmoy Deb. Multiobjective optimization using nondominated sorting in genetic algorithms. *Evolutionary Computation*, 2(3):221–248, 1994.

[28] L. Thiele, K. Miettinen, P. Korhonen, and J. Molina. A preference-based interactive evolutionary algorithm for multiobjective optimization. Technical Report W-412, Helsinki School of Economics, Finland, 2007.

[29] D. A. Van Veldhuizen and G. B. Lamont. On Measuring Multiobjective Evolutionary Algorithm Performance. In *Congress on Evolutionary Computation (CEC 2000)*, volume 1, pages 204–211. IEEE Service Center, 2000.

[30] David A. Van Veldhuizen and Gary B. Lamont. Multiobjective Evolutionary Algorithms: Analyzing the State-of-the-Art. *Evolutionary Computation*, 8(2):125–147, 2000.

[31] E. Zitzler, D. Brockhoff, and L. Thiele. The Hypervolume Indicator Revisited: On the Design of Pareto-compliant Indicators Via Weighted Integration. In S. Obayashi et al., editors, *Conference on Evolutionary Multi-Criterion Optimization (EMO 2007)*, volume 4403 of *LNCS*, pages 862–876, Berlin, 2007. Springer.

[32] E. Zitzler and S. Künzli. Indicator-Based Selection in Multiobjective Search. In *Conference on Parallel Problem Solving from Nature (PPSN VIII)*, volume 3242 of *LNCS*, pages 832–842. Springer, 2004.

[33] E. Zitzler, M. Laumanns, and L. Thiele. SPEA2: Improving the Strength Pareto Evolutionary Algorithm for Multiobjective Optimization. In K.C. Giannakoglou et al., editors, *Evolutionary Methods for Design, Optimisation and Control with Application to Industrial Problems (EUROGEN 2001)*, pages 95–100. International Center for Numerical Methods in Engineering (CIMNE), 2002.

[34] E. Zitzler and L. Thiele. Multiobjective Evolutionary Algorithms: A Comparative Case Study and the Strength Pareto Approach. *IEEE Transactions on Evolutionary Computation*, 3(4):257–271, 1999.

[35] E. Zitzler, L. Thiele, M. Laumanns, C. M. Foneseca, and V. Grunert da Fonseca. Performance Assessment of Multiobjective Optimizers: An Analysis and Review. *IEEE Transactions on Evolutionary Computation*, 7(2):117–132, 2003.

APPENDIX

Proof of Corollary 2.4: Given $m \geq s$ and a population P with $|P| = m$. Let us suppose that P does not contain a representative of some equivalence class in $\mathcal{X}^*$ and denote a representative as a^*. As $m \geq s$, there are either non-minimal elements in P or several representatives of some (other) equivalence class. If we replace such an element by a^*, the new population P^* will be better than P and P was not minimal.

Given $m < s$ and a population P with $|P| = m$.

- Let us suppose that P contains a non-minimal solution a. Therefore, there exists a minimal solution a^* in $\mathcal{X}^*$ that is better than a. If we replace a by a^* in P, the new set is better than P and P was not minimal.
- Let us suppose that P contains two equivalent minimal solutions $a \equiv b$. Then there exists a minimal solution c^* in $\mathcal{X}^*$ that is not equivalent to any other solution in P (as $m < s$). If we replace a by c^* then the new set is better than P and P was not minimal.
- Let us suppose that P is not minimal, i.e. not an element of $\text{Min}(\Psi_m, \preccurlyeq)$. Then there exists a set P^* that is better than P, i.e. $P^* \preccurlyeq P$ and $P \not\preccurlyeq P^*$. Then to each solution in P there is one in P^* that is either equivalent or strictly better. In the latter case, P did not only contain minimal solutions. In the former case, there is a solution in P^* which is incomparable to all solutions in P as $P \not\preccurlyeq P^*$. Therefore, P contained at least one pair of equivalent solutions.

■

Proof of Theorem 2.8: Reflexivity: As $A \preccurlyeq^i A$ holds for all $1 \leq i \leq k$ (since all $\preccurlyeq^i$ are preorders), we have $i = k$ in Def. 2.7 (i). Therefore, $(A \preccurlyeq_{\text{S}} A) \Leftrightarrow (A \preccurlyeq^i A)$ and the reflexivity holds. Transitivity: We proof transitivity by induction. We first need to show that transitivity holds for $k = 1$. In this case, we have $A \preccurlyeq_{\text{S}} B \Leftrightarrow A \preccurlyeq^1 B$ as $i = k$ in Def. 2.7 (i). Transitivity holds as $\preccurlyeq^1$ is a preorder. Now we have to show that transitivity holds for k if it holds for $k-1$. Let us define the sequence of length $k-1$ as S'. Then we can reformulate Def. 2.7 as follows:

$$(A \preccurlyeq_{\text{S}} B) \Leftrightarrow ((A \equiv_{\text{S}'} B) \wedge (A \preccurlyeq^k B)) \vee (A \prec_{\text{S}'} B) \quad (6)$$

Now, we can show that transitivity holds:

$$
\begin{aligned}
&(A \preccurlyeq_{\mathrm{S}} B) \wedge (B \preccurlyeq_{\mathrm{S}} C) \Rightarrow \\
&\quad \Rightarrow [((A \equiv_{\mathrm{S}'} B) \wedge (A \preccurlyeq^k B)) \vee (A \prec_{\mathrm{S}'} B)] \wedge \\
&\qquad [((B \equiv_{\mathrm{S}'} C) \wedge (B \preccurlyeq^k C)) \vee (B \prec_{\mathrm{S}'} C)] \Rightarrow \\
&\quad \Rightarrow ((A \equiv_{\mathrm{S}'} B) \wedge (B \equiv_{\mathrm{S}'} C) \wedge (A \preccurlyeq^k B) \wedge (\preccurlyeq^k C)) \vee \\
&\qquad ((A \prec_{\mathrm{S}'} B) \wedge (B \prec_{\mathrm{S}'} C)) \Rightarrow \\
&\quad \Rightarrow ((A \equiv_{\mathrm{S}'} C) \wedge (A \preccurlyeq^k C)) \vee (A \prec_{\mathrm{S}'} C) \Rightarrow A \preccurlyeq_{\mathrm{S}} C
\end{aligned}
$$

■

Proof of Theorem 2.9: A preorder $\preccurlyeq$ is called total if $(A \preccurlyeq B) \vee (B \preccurlyeq A)$ holds for all $A, B \in \Psi$. Using the same induction principle as in the proof of 2.8 we can notice that for $k = 1$ we have $(A \preccurlyeq_{\mathrm{S}} B) \Leftrightarrow (A \preccurlyeq^1 B)$ and therefore, $\preccurlyeq_{\mathrm{S}}$ is total. For the induction we know that (6) holds. Therefore, we can conclude:

$$
\begin{aligned}
&(A \preccurlyeq_{\mathrm{S}} B) \vee (B \preccurlyeq_{\mathrm{S}} A) \Leftrightarrow \\
&\quad \Leftrightarrow ((A \equiv_{\mathrm{S}'} B) \wedge (A \preccurlyeq^k B)) \vee ((B \equiv_{\mathrm{S}'} A) \wedge (B \preccurlyeq^k A)) \vee \\
&\qquad (A \prec_{\mathrm{S}'} B) \vee (B \prec_{\mathrm{S}'} A) \Leftrightarrow \\
&\quad \Leftrightarrow (A \equiv_{\mathrm{S}'} B) \vee (A \prec_{\mathrm{S}'} B) \vee (B \prec_{\mathrm{S}'} A) \Leftrightarrow \mathrm{true}
\end{aligned}
$$

■

In order to proof Theorem 2.10 we first need to state a lemma that states that $\preccurlyeq_{\mathrm{S}}$ refines a given preference relation $\preccurlyeq$ if the last preference relation is a refinement.

Lemma A.1: If $\preccurlyeq^k$ in Def. 2.7 is a refinement of a given preference relation $\preccurlyeq$ and all relations $\preccurlyeq^j$, $1 \leq j < k$ are weak refinements of $\preccurlyeq$, then $\preccurlyeq_{\mathrm{S}}$ is a refinement of $\preccurlyeq$.

Proof of A.1: Let us suppose that $A \prec B$ holds for some $A, B \in \Psi$. We have to show that $A \prec_{\mathrm{S}'} B$ holds. At first note, the $A \preccurlyeq^j B$ holds for all $1 \leq j < k$ as $\preccurlyeq^j$ are weak refinements and $A \prec^k B$ holds as $\preccurlyeq^k$ is a refinement. Let us now consider the sequence S' of length $k-1$. Because all $\preccurlyeq^j$ are weak refinements, either $A \equiv^j B$ or $A \prec^j B$ holds. Taking into account the construction of S' according to Def. 2.7 we can easily see that $A \preccurlyeq_{\mathrm{S}'} B$ holds. Based on the fact that $\preccurlyeq_{\mathrm{S}'}$ is a weak refinement we will show that $A \prec_{\mathrm{S}} B$ holds, i.e. $\prec_{\mathrm{S}}$ is a refinement. To this end, we again use (6) to derive

$$
\begin{aligned}
&(A \preccurlyeq_{\mathrm{S}} B) \wedge (B \not\preccurlyeq_{\mathrm{S}} A) \Leftrightarrow \\
&\quad \Leftrightarrow [((A \equiv_{\mathrm{S}'} B) \wedge (A \preccurlyeq^k B)) \vee (A \prec_{\mathrm{S}'} B)] \wedge \\
&\qquad ((B \not\equiv_{\mathrm{S}'} A) \vee (B \not\preccurlyeq^k A)) \wedge (A \not\prec_{\mathrm{S}'} B)
\end{aligned} \tag{7}
$$

As $\preccurlyeq_{\mathrm{S}'}$ is a weak refinement, we need to consider two cases. If $A \equiv_{\mathrm{S}'} B$ holds, then $A \not\prec_{\mathrm{S}'} B$ holds as well as $B \not\prec_{\mathrm{S}'} A$. In this case, the expression becomes $(A \preccurlyeq^k B) \wedge (B \not\preccurlyeq^k A)$ which yields true. If $A \prec_{\mathrm{S}'} B$ holds, then $A \not\equiv_{\mathrm{S}'} B$, $B \not\equiv_{\mathrm{S}'} A$ and $B \not\prec_{\mathrm{S}'} A$ hold. The expression above becomes now $(A \prec_{\mathrm{S}'} B) \wedge (B \not\prec_{\mathrm{S}'} A)$ which also yields true. ■

Now we can give the proof of Theorem 2.10.

Proof of Theorem 2.10: Because of Lemma A.1, we know that the sequence $S' = (\preccurlyeq^1, \preccurlyeq^2, \ldots, \preccurlyeq^{k'})$ leads to a refinement of $\preccurlyeq$. We just need to show that additional preference relations $\preccurlyeq^j$, $k' < j \leq k$ in the sequence do not destroy this property. We again use the same induction principle as in the previous proofs. Let us suppose that S' yields a refinement (as shown above) and S has one additional relation $\preccurlyeq^{k'+1}$, i.e. $k = k' + 1$. Using again (6) we can derive the expression for $A \prec_{\mathrm{S}} B$ as in (7). If we suppose that $A \prec B$ holds in the given preorder, and $\preccurlyeq_{\mathrm{S}'}$ is a refinement, the relations $A \not\equiv_{\mathrm{S}'} B$, $B \not\equiv_{\mathrm{S}'} A$, $A \prec_{\mathrm{S}'} A$ and $B \not\prec_{\mathrm{S}'} A$ hold. For the expression in (7) we get $(A \preccurlyeq^k B) \wedge (B \not\preccurlyeq^k A)$ which yields true. ■

Proof of Theorem 3.2: Suppose conditions 1 and 2 hold, and let $A, B \in \Psi$ be two arbitrary sets with $A \prec B$, i.e. $(A \preccurlyeq B) \wedge (B \not\preccurlyeq A)$. For the proof, we will apply the two local transformations in order to gradually change B to A and show that at each step the indicator value does not increase and there is at least one step where it decreases. First, we successively add the elements of B to A; since for each $b \in B$ it holds $A \preccurlyeq \{b\}$, according to condition 1 the indicator value remains constant after each step, i.e., $I(A) = I(A \cup B)$. Now, we successively add the elements of A to B; since $A \prec B$, there exists an element $a \in A$ such that $B \not\preccurlyeq \{a\}$ according to the conformance of $\preccurlyeq$ with $\preceq$. That means when adding the elements of A to B the indicator value either remains unchanged (condition 1) or decreases (and it will decrease at least once, namely for a, according to condition 2), and therefore $I(A \cup B) < I(B)$. Combining the two intermediate results, one obtains $I(A) = I(A \cup B) < I(B)$ which implies $A \preccurlyeq_{\mathrm{I}} B$ and $B \not\preccurlyeq_{\mathrm{I}} A$. Hence, $\preccurlyeq_{\mathrm{I}}$ refines $\preccurlyeq$. For weak refinement, the proof is analogous. ■

Proof of Theorem 3.5: Suppose conditions 1 and 2 hold, and let $A, B \in \Psi$ be two arbitrary sets with $A \prec B$, i.e. $(A \preccurlyeq B) \wedge (B \not\preccurlyeq A)$. First, we note that due to condition 1 adding a solution b to the set A with $A \preccurlyeq \{b\}$ does not change its preferability, i.e., $A \preccurlyeq_{\mathrm{I}} A \cup \{b\}$ and $A \cup \{b\} \preccurlyeq_{\mathrm{I}} A$. Therefore, by adding the elements of B one by one to A never changes the preferability and overall it follows that $A \preccurlyeq_{\mathrm{I}} A \cup B$ and $A \cup B \preccurlyeq_{\mathrm{I}} A$. Second, we note that due to conditions 1 and 2 adding an arbitrary solution a to the set B makes the new set at least as preferable as B, i.e., $\{a\} \cup B \preccurlyeq_{\mathrm{I}} B$. Now, we add the elements $\{a_1, a_2, \ldots, a_t\}$ of A one by one to B; the union of both sets becomes at least as preferable as B, i.e., $A \cup B \preccurlyeq_{\mathrm{I}} B$. Let a_i be the first element in the sequence $(a_1, a_2, \ldots, a_t)$ for which holds $B \not\preccurlyeq \{a_i\}$. After adding a_i, it holds $\{a_1, \ldots, a_{i-1}, a_i\} \cup B \prec_{\mathrm{I}} \{a_1, \ldots a_{i-1}\} \cup B$. From the transitivity of $\preccurlyeq_{\mathrm{I}}$ and $\prec_{\mathrm{I}}$, it then follows that $A \cup B \prec_{\mathrm{I}} B$ and overall $A \prec_{\mathrm{I}} B$. For weak refinement, the proof is analogous.

To show the correctness of the necessary condition, we suppose $A \not\preccurlyeq \{b\}$. According to Definition 2.2, $(A \cup \{b\}) \prec A$ which implies that $(A \cup \{b\}) \preccurlyeq_{\mathrm{I}} A$ (weak refinement) or $(A \cup \{b\}) \prec_{\mathrm{I}} A$ (refinement), respectively. Hence, $I(A \cup \{b\}, A) \leq I(A, A \cup \{b\})$ respectively $I(A \cup \{b\}, A) < I(A, A \cup \{b\})$ according to (4). ■

Proof of Corollary 4.5: We will show that in the case of a 3-dimensional objective space $\mathcal{Z} \subset \mathbb{R}^3$, the epsilon indicator $I_{\epsilon 1}(A) = I_\epsilon(A, R)$ is not 1-greedy.

The proof is done by providing a counterexample. We will give a scenario consisting of an objective space $\mathcal{Z} \subset \mathbb{R}^3$, reference set $R \subset \mathcal{Z}$, a maximum set size m and an initial set

$P \in \Psi_m$. We will show that there is no path from the initial set P to an optimal set P^* such that every time only one element of the set is exchanged. In particular, we have $\mathcal{X} = \{x^1, x^2, x^3, x^4\}$, $R = \{r^1, r^2, r^3, r^4\}$, $m = 2$, $P = \{x^1, x^2\}$, $P^* = \{x^3, x^4\}$, and $f : \mathcal{X} \to \mathcal{Z}$ according to Table V. With

TABLE V
COUNTEREXAMPLE FOR 1-GREEDYNESS OF THE UNARY EPSILON INDICATOR.

	r^1	r^2	r^3	r^4	$f(x^1)$	$f(x^2)$	$f(x^3)$	$f(x^4)$
1	4	0	6	3	6	2	0	3
2	0	4	3	6	2	6	0	3
3	6	6	2	2	4	4	6	2

the values $\epsilon(z, r)$ shown in Table VI we obtain the epsilon indicator value for all possible sets as $I_\epsilon(\{x^1, x^2\}, R) = 2$, $I_\epsilon(\{x^1, x^3\}, R) = 3$, $I_\epsilon(\{x^1, x^4\}, R) = 3$, $I_\epsilon(\{x^2, x^3\}, R) = 3$, $I_\epsilon(\{x^2, x^4\}, R) = 3$ and $I_\epsilon(\{x^3, x^4\}, R) = 0$. Therefore,

TABLE VI
VALUES $\epsilon(z, r)$ FOR THE COUNTEREXAMPLE.

	$f(x^1)$	$f(x^2)$	$f(x^3)$	$f(x^4)$
r^1	2	6	0	3
r^2	6	2	0	3
r^3	2	3	4	0
r^4	3	2	4	0

from the initial set $\{x^1, x^2\}$, every set mutation operator that exchanges one element only leads to a worse indicator value.

Finally, note that it can be shown, that in $\mathbb{R}^2$, the epsilon indicator is 1-greedy. ■

Proof of Corollary 4.6:

TABLE VII
COUNTEREXAMPLE FOR 1-GREEDYNESS OF THE HYPERVOLUME INDICATOR.

	r	$f(x^1)$	$f(x^2)$	$f(x^3)$	$f(x^4)$
1	10	1	6	5	7
2	7	6	2	3	1

We will show that in the case of a 2-dimensional objective space $\mathcal{Z} \subset \mathbb{R}^2$, the hypervolume indicator I_H is not 1-greedy.

The proof is done by providing a counterexample similar to the proof of the previous Corollary. Here we use again a decision space $\mathcal{X} = \{x^1, x^2, x^3, x^4\}$, a reference set $R = \{r\}$, an initial set $P = \{x^1, x^2\}$, an optimal set $P^* = \{x^3, x^4\}$, and $f : \mathcal{X} \to \mathcal{Z}$ according to Table VII. Using this scenario, we obtain the hypervolume indicators for all possible populations as $I_H(\{x^1, x^2\}, R) = -25$, $I_H(\{x^1, x^3\}, R) = -24$, $I_H(\{x^1, x^4\}, R) = -24$, $I_H(\{x^2, x^3\}, R) = -24$, $I_H(\{x^2, x^4\}, R) = -24$ and $I_H(\{x^3, x^4\}, R) = -26$. Therefore, from the initial set $\{x^1, x^2\}$, every set mutation operator that exchanges one element only leads to a worse indicator value. ■

Weight combinations for $\preccurlyeq^{\text{minpart}}_{R2,H}$: In the case of five objectives, overall $32 \cdot 5$ weight combinations are used for the set preference relation $\preccurlyeq^{\text{minpart}}_{R2,H}$, cf. Table I. In detail, Λ is defined as follows:

$$
\begin{aligned}
\Lambda \quad = \quad & \{ \\
& (0,0,0,0,1), \\
& (0.01/4, 0.01/4, 0.01/4, 0.01/4, 0.99), \\
& \ldots, \\
& (0.3/4, 0.3/4, 0.3/4, 0.3/4, 0.7) \\
& \} \\
\cup \quad & \\
& \{ \\
& (0,0,0,1,0), \\
& (0.01/4, 0.01/4, 0.01/4, 0.99, 0.01/4), \\
& \ldots, \\
& (0.3/4, 0.3/4, 0.3/4, 0.7, 0.3/4) \\
& \} \\
\cup \quad & \\
\ldots \quad & \\
\cup \quad & \\
& \{ \\
& (1,0,0,0,0), \\
& (0.99, 0.01/4, 0.01/4, 0.01/4, 0.01/4), \\
& \ldots, \\
& (0.7, 0.3/4, 0.3/4, 0.3/4, 0.3/4) \\
& \}
\end{aligned}
$$

■