# Set-based Multi-Objective Optimization, Indicators, and Deteriorative Cycles


Rudolf Berghammer
Institut für Informatik
Christian-Albrechts-Universität zu Kiel
Kiel, Germany

Tobias Friedrich
Algorithms and Complexity
Max-Planck-Institut für Informatik
Saarbrücken, Germany

Frank Neumann
Algorithms and Complexity
Max-Planck-Institut für Informatik
Saarbrücken, Germany



## ABSTRACT

Evolutionary multi-objective optimization deals with the task of computing a minimal set of search points according to a given set of objective functions. The task has been made explicit in a recent paper by Zitzler et al. [13]. We take an order-theoretic view on this task and examine how the use of indicator functions can help to direct the search towards Pareto optimal sets. Thereby, we point out that evolutionary algorithms for multi-objective optimization working on the dominance relation of search points have to deal with a cyclic behavior that may lead to worsenings with respect to the Pareto-dominance relation defined on sets. Later on, we point out in which situations well-known binary and unary indicators can help to avoid this cyclic behavior.



## Categories and Subject Descriptors

F.2 [**Theory of Computation**]: Analysis of Algorithms and Problem Complexity

## General Terms

Theory, Algorithms, Measurement, Performance

## Keywords

Multiobjective Optimization, Performance Measures, Hypervolume Indicator, Cycles


## 1. INTRODUCTION

Evolutionary computation methods have been widely used for multi-objective optimization problems [3]. Often such problems are hard to tackle as there is no total ordering of the underlying search space. Instead of this, a preorder on the search space is induced by the different objective functions that should be optimized. The minimal elements of such a preorder are called Pareto optimal solutions and the set of minimal elements with respect to the partial order on the objective space is called the Pareto front. The goal of an algorithm for a given multi-objective optimization problem is to compute for each objective vector of the Pareto front a corresponding Pareto optimal search point. Often the size of the Pareto front grows exponentially with respect to the size of the input. Then one is interested in a smaller set of minimal elements that fulfills given user preferences.

Recently, it has been proposed by Zitzler, Thiele, and Bader [12] to extend the preorder on the search space to a preorder on sets of search points. This leads to the term set-based multi-objective optimization which stresses the point that not single solutions should be compared with respect to a preorder but sets of search points have to be compared. Zitzler et al. [12, 13] introduced a preorder on the set of sets of search points which is based on the preorder on the underlying search space. They also use this preorder as a basic relation that may be later refined by special user preferences.

The goal of this paper is to consider the approach of working with a preorder on sets in greater detail. In particular, we discuss the preorder on sets from a theoretical point of view. Evolutionary algorithms work with (multi)-sets of search points. Throughout this paper, we assume that in each iteration a parent population consisting of $\mu$ individuals produces an offspring population consisting of $\lambda$ individuals. Having produced an offspring population, the task of the selection operator is to select a new parent population. We examine how to design a selection operator such that the newly chosen parent population consists of a set of $\mu$ individuals that is minimal among all sets that can be obtained by selecting $\mu$ individuals from the set of parents and children. Later on, we relate it to well known selection methods such as non dominated sorting used in NSGA-II [4] and ranking ideas used in SPEA2 [9].

Having examined how to compute a minimal set of search points in each iteration, we investigate the run of such algorithms from a theoretical point of view with respect to the Pareto-dominance order defined on sets. We show that just working with the dominance relation on sets may lead to a cyclic behavior that can lead to worsenings with respect to the dominance relation. This behavior has already been observed in experimental studies of NSGA-II and SPEA2 (see [7]) and may prevent those algorithms from convergence. We point out conditions that can help to avoid such an undesired behavior. Based on these conditions, we examine how refinements using indicator functions for incomparable sets may help to solve this problem. We show that well-known binary indicators such as the additive and multiplicative $\varepsilon$-indicator



*GECCO'10,* July 7–11, 2010, Portland, Oregon, USA.

can resolve this problem for the case that the parent population is of size 1. On the negative side, we show that even a parent population of size 2 can again run into the non-described cyclic behavior.

On the other hand, we examine the use of unary indicators. Based on our conditions for avoiding cyclic behavior we show that the well-known hypervolume indicator is able to deal with this problem in a successful manner. Our investigations increase the foundation of this indicator and give a further explanation for its usefulness in evolutionary multi-objective optimization.

The outline of the paper is as follows. In Section 2 we recall some basic properties of set-based multi-objective optimization. Section 3 deals with the task of computing a minimal set from the parents and children. The problem of deteriorative cycles and conditions on how to avoid them are pointed out in Section 4. In Section 5 we examine whether it is possible to avoid this cyclic behavior by using binary and unary indicators. The cyclic behavior of the popular $\varepsilon$-indicator is examined in detail in Section 6. Finally, we finish with some conclusions.

## 2. SET-BASED MULTI-OBJECTIVE OPTIMIZATION

A multi-objective optimization problem is given by a vector-valued objective function $f = (f_1, \ldots, f_d) \colon X \to \mathbb{R}^d$ on a search space $X$. W. l. o. g. we assume that each function $f_i$, $1 \leq i \leq d$, should be minimized. We first define a partial order on the objective space. An objective vector $x = (x_1, \ldots, x_d) \in \mathbb{R}^d$ weakly dominates an objective vector $y = (y_1, \ldots, y_d) \in \mathbb{R}^d$ ($x \preceq_{Par} y$) if it is not worse in any objective, i. e.,

$$x \preceq_{Par} y :\Leftrightarrow x_i \leq y_i \text{ for } 1 \leq i \leq d.$$

The objective function $f$ also induces a preorder $\preceq_{Par}$ on the search space $X$. More precisely, we say a search point $a \in X$ weakly dominates a search point $b \in X$ ($a \preceq_{Par} b$) if it is not worse in any of its objective, i. e.,

$$a \preceq_{Par} b :\Leftrightarrow f(a) \preceq_{Par} f(b).$$

Note that we use $\preceq_{Par}$ as a relation on search points as well as a relation on the corresponding objective vectors. $\preceq_{Par}$ is a preorder on the set of search points and a partial order on the set of objective vectors.

Investigating sets of search points in this paper, we assume that these sets are finite. Given a set of search points $A$, we denote by $Min(A, \preceq_{Par})$ the set of minimal elements of $A$ with respect to the preorder $\preceq_{Par}$. Let $f(A)$ be the set of objective vectors of the search points in $A$, i. e.,

$$f(A) = \bigcup_{a \in A} f(A).$$

Then, we denote by $Min(f(A), \preceq_{Par})$ the set of minimal objective vectors in $f(A)$ with respect to the partial order $\preceq_{Par}$ on $f(A)$.

The goal in multi-objective optimization is to compute a set $X^*$ with $f(X^*) = Min(f(X), \preceq_{Par})$, where $X$ is the considered search space. Often the size of $Min(f(X), \preceq_{Par})$ is large, i. e., exponential with respect to the given input. In this case, it is not possible to compute the whole set of minimal elements of $f(X)$ efficiently and $f(X^*)$ should be a smaller subset of them.

In this paper, we consider evolutionary algorithms for multi-objective optimization. They try to construct a set $X^*$ with $f(X^*) \subseteq Min(f(X), \preceq_{Par})$ in an iterative way by starting with an initial set of search points and producing a new set of search points in each iteration. As evolutionary algorithms work with sets of search points, we want to compare sets of search points against each other. Let $2^X$ be the power-set of $X$, i. e., $2^X := \{R \mid R \subseteq X\}$. Based on a preorder $\preceq_{Par}$ (reflexive and transitive) on single search points, Zitzler et al. [12] have defined the following preorder on sets of search points which conforms to the preorder $\preceq_{Par}$ on the underlying set of search points $X$.

Definition 2.1. *Let $A, B \in 2^X$ then*

$$A \preceq_{dom} B :\Leftrightarrow (\forall b \in B\ \exists a \in A \colon a \preceq_{Par} b).$$

We consider an arbitrary relation $\preceq$ on sets and introduce the following definitions.

Definition 2.2. *Let $A, B \in 2^X$ and $\preceq$ be an arbitrary relation on $2^X$. Then we use*

$$A \prec B :\Leftrightarrow (A \preceq B) \wedge (B \not\preceq A)$$

*to denote that $A$ is strictly better than $B$. We further write*

$$A \equiv B :\Leftrightarrow (A \preceq B) \wedge (B \preceq A)$$

*to denote that $A$ and $B$ are equivalent, and*

$$A \parallel B :\Leftrightarrow (A \not\preceq B) \wedge (B \not\preceq A)$$

*to denote that $A$ and $B$ are incomparable. Further, let*

$$A \succeq B :\Leftrightarrow B \preceq A,$$
$$A \succ B :\Leftrightarrow B \prec A.$$

Evolutionary algorithms often work at each time step with a fixed population size $\mu$. Therefore, we consider subsets of $X$ containing exactly $\mu$ elements. The goal is to obtain a set that is a minimal set with respect to the order $\preceq_{dom}$ among all subsets of $X$ having exactly $\mu$ elements.

We denote the set of minimal elements containing exactly $\mu$ elements of $X$ by $Min_\mu(2^X, \preceq)$, i. e.

$$Min_\mu(2^X, \preceq) := Min\{R \mid R \in 2^X \wedge |R| = \mu\}.$$

Note that such a set is also a minimal set among all subsets having less than $\mu$ elements according to Definition 2.1.

The relation between two sets $A$ and $B$ is determined by the set of their minimal elements. The following lemma relates equivalent sets and the set of their minimal objective vectors.

Lemma 2.3. *If $A$ and $B$ are sets of search points, then*

$$A \equiv_{dom} B \Leftrightarrow Min(f(A), \preceq_{Par}) = Min(f(B), \preceq_{Par}).$$

Proof. "$\Rightarrow$": Let $A, B \in 2^X$ with $A \equiv_{dom} B$. We assume that $Min(f(A), \preceq_{Par}) \neq Min(f(B), \preceq_{Par})$ and show a contradiction. Let $c \in A$ be such that $f(c) \in Min(f(A), \preceq_{Par})$ but $f(c) \notin Min(f(B), \preceq_{Par})$. Since $B \preceq_{dom} A$ and $f(c) \in f(A)$ but $f(c) \notin Min(f(B), \preceq_{Par})$ there exists a $b \in B$ such that $f(b) \in Min(f(B), \preceq_{Par})$ and $b \prec_{Par} c$. From $A \preceq_{dom} B$ we get $a \preceq_{Par} b$ for some $a \in A$. Hence, we have $a \preceq_{Par} c$ and $f(c) \in Min(f(A), \preceq_{Par})$ implies $f(a) = f(c)$. Altogether, we arrive at $f(b) \prec_{Par} f(a)$ and $f(a) \preceq_{Par} f(b)$, i. e., the desired contradiction.

"$\Leftarrow$": Let $f(a) \in f(A)$. Then there exists $f(b) \in Min(f(A), \preceq_{Par})$ such that $f(b) \preceq_{Par} f(a)$. Now $Min(f(A), \preceq_{Par}) = Min(f(B), \preceq_{Par})$ implies $f(b) \in f(B)$. Hence, we get $B \preceq_{dom} A$. Since in the same way $A \preceq_{dom} B$ can be shown, we have $A \equiv_{dom} B$. $\square$

Evolutionary algorithms work in each iteration with a parent population that creates an offspring population by some variation operators such as crossover and mutation. Note, that both the parent and offspring population are multi-sets, i. e., they may contain a search point more than once. Comparing multi-sets with respect to the dominance relation, we ignore duplicates, i. e. we treat a multi-sets as their corresponding sets in $2^X$.

After the set of offspring has been obtained, the goal is to choose a new parent population such that the process can be iterated.

We assume that the parent population has a fixed number of individuals $\mu$ which is very common in evolutionary computation. The size of the offspring population is denoted by $\lambda$ and the goal is to select out of the $\mu + \lambda$ individuals from the parent and offspring a new parent population that is a minimal set among all possible subsets consisting of $\mu$ individuals

To make the setting more precise, we examine Algorithm 1. Our algorithm starts with a population consisting of $\mu$ individuals. In each iteration $\lambda$ offspring are produced. The new parent population is afterwards chosen as a minimal element with respect to the set preference relation $\preceq_{dom}$ among all possible subsets of the parents and offspring that consist of exactly $\mu$ individuals.

Algorithm 1 (Evolutionary Algorithm).

1. *Create an initial population $P$ consisting of $\mu$ individuals.*
2. *Produce from $P$ an offspring population $C$ consisting of $\lambda$ individuals.*
3. *Select a set $P'$ with $P' \in Min_\mu(2^{(P \cup C)}, \preceq_{dom})$.*
4. *Set $P = P'$*
5. *If no termination condition is fullfilled go to step 2*

Consider a set $P' \in Min_\mu(2^{(P \cup C)}, \preceq)$ and compare it to $P$. $P'$ is minimal and $P \in \{R \in 2^{(P \cup C)} \wedge |R| = \mu\}$ which implies that either

$$P' \preceq_{dom} P$$

or

$$P' \parallel_{dom} P$$

holds. $P \prec_{dom} P'$ would contradict the assumption that $P'$ is minimal. If $P' \prec_{dom} P$ we have obtained a strict improvement with respect to the dominance relation on sets. $P' \equiv_{dom} P$ gives us an equivalent set and $P' \parallel_{dom} P$ a set that is incomparable.

Evolutionary algorithms in our setting work implicitly on a total relation defined on sets as an algorithm has to make the decision which set to take for the next iteration. Let $\preceq_{Alg}$ be the total relation on $2^X$ that an algorithm $Alg$ uses implicitly. In the case that $P' \parallel_{dom} P$ it is not clear which set to favor over the other. In the following we treat incomparable sets in the same way as indifferent sets. This is very common in evolutionary multi-objective optimization if no additional information is available. Later on, we will examine how additional information based on an indicator function can influence the search.

Algorithm 1 is based on the dominance relation $\preceq_{dom}$ on sets. However, it may also move from a set $P$ to $P'$ iff $P \parallel_{dom} P'$ which is often the case for evolutionary algorithms. The algorithm works implicitly on the total relation $\preceq_{Alg_1}$ on $2^X$ defined as

$$A \preceq_{Alg_1} B \Leftrightarrow (A \preceq_{dom} B) \vee (A \parallel_{dom} B)$$

and may move from $P$ to $P'$ iff $P' \preceq_{Alg_1} P$ holds. Note that $\preceq_{Alg_1}$ is not necessarily a transitive relation.

# 3. COMPUTING MINIMAL SETS

In this section, we examine how to compute a set contained in $Min_\mu(2^{(P \cup C)}, \preceq_{dom})$. We will see that this can be done by using an iterative algorithm that chooses in each iteration a minimal element with respect to the Pareto dominance relation on single points. This is similar to how well known evolutionary algorithms for multi-objective optimization choose their offspring population. In the following, we present the ideas and basic properties that are necessary to compute such a set in a precise way.

We consider the preorder $\preceq_{dom}$ on the subsets of $S := P \cup C$ with exactly $\mu$ elements. To obtain a set $T$ for which $T \in Min_\mu(2^S, \preceq_{dom})$ holds we consider Algorithm 2.

Algorithm 2 ($Min_\mu(2^S, \preceq_{dom})$).

*Input: $S$ with $|S| \geq \mu$.*

1. *$T = \emptyset$.*
2. *while $|T| < \mu$*
   - *Choose an element $x \in Min(S, \preceq_{Par})$*
   - *$T = T \cup \{x\}$, $S = S \setminus \{x\}$.*

Algorithm 2 chooses in each iteration one individual $x$ that is minimal with respect to $S$ and $\preceq_{Par}$. This individual is introduced into $T$ ($T = T \cup \{x\}$) and deleted from $S$ ($S \setminus \{x\}$) until $\mu$ individuals have been chosen in this way. The following theorem shows that Algorithm 2 computes a minimal set among all subsets of $S$ that have exactly $\mu$ elements.

Theorem 3.1. *For the set $T$ produced by Algorithm 2, $T \in Min_\mu(2^S, \preceq_{dom})$ holds.*

Proof. Algorithm 2 selects an element $x$ if $x \in Min(S, \preceq_{Par})$. Hence, $x$ can only be dominated by another $y \in S$ that has been included into $T$ before the selection of $x$ has taken place. This implies that there is no element $z$ in $S \setminus P$ for which $z \preceq_{Par} x$ holds. Therefore, $T \in Min_\mu(2^S, \preceq_{dom})$ holds. $\square$

In the following, we want to modify Algorithm 2 such that user preferences can be incorporated. Often one is not only interested in an arbitrary set $T \in Min_\mu(2^S, \preceq_{dom})$, but a set from $Min_\mu(2^S, \preceq_{dom})$ having additional properties. In the process of construction the set $T$ we use a heuristic function

$$h \colon 2^X \times X \to \mathbb{R}$$

which determines the choice of the next minimal element that should be included into $T$ depending on the already chosen elements and the available minimal elements.

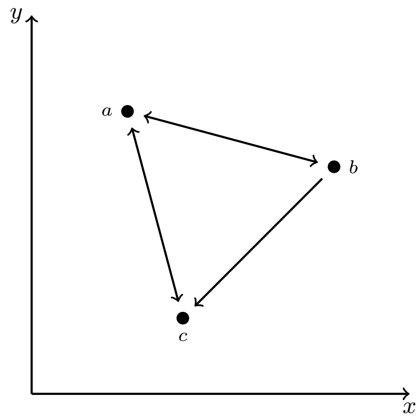


**Figure 1: Objective space for deteriorative cycle based on Pareto-dominance relation.**

ALGORITHM 3 ($Min_\mu(2^S, \preceq_{dom})$ WITH PREFERENCES).

*Input:* $S$ *with* $|S| \geq \mu$.

1. $T = \emptyset$.
2. *while* $|P| < \mu$
   - *Choose an element* $x \in Min(S, \preceq_{Par})$ *for which* $h(P,x)$ *is maximal.*
   - $T = T \cup \{x\}$, $S = S \setminus \{x\}$.

Algorithm 3 computes a minimal set of elements taking into account a heuristic function $h$. This function can be used to incorporate information on how the chosen points should relate to each other. As Algorithm 3 selects in each iteration a minimal element from the remaining set, we can state the following corollary.

COROLLARY 3.2. *For the set* $T$ *produced by Algorithm 3,* $T \in Min_\mu(2^S, \preceq_{dom})$ *holds.*

We have shown two simple algorithms for computing a minimal set. Note that many well-known evolutionary algorithms for multi-objective optimization such as NSGA-II [4] and SPEA2 [9] use a similar approach to compute the next parent population. In fact, it can be shown by similar arguments that they also compute a set in $Min_\mu(2^S, \preceq_{dom})$.

# 4. DETERIORATIVE CYCLES

In this section, we examine algorithms computing always a minimal set among all possible sets consisting of $\mu$ individuals. We point out that using just the preference order on the different subsets of $P \cup C$ may lead to cycles in the optimization process, i. e., the algorithm may return to a set of search points that has obtained already before. Even worse, we show that the algorithm may return to a set of search points that is strictly dominated by another set of search points that has been obtained at an earlier stage of the optimization process.

## 4.1 The Problem of Deteriorative Cycles

We want to point out that using an approach which chooses an *arbitrary* set of $Min_\mu(2^{(P\cup C)}, \preceq_{dom})$ may create cycles. In particular, we show by example that we might get worse during the optimization process according to the preorder order $\preceq_{dom}$. We call such cycles deteriorative and make it precise in using the following definition.

DEFINITION 4.1. *A relation* $\preceq$ *on* $2^X$ *contains a deteriorative cycle iff there is a sequence of sets* $A_1, A_2, \ldots A_r \in 2^X$ *with*

$$A_1 \preceq A_2 \preceq \ldots \preceq A_{r-1} \preceq A_r \preceq A_1$$

*and* $A_r \prec_{dom} A_1$.

To illustrate the problem that deteriorative cycles may produce in the optimization process, we consider a simple example given in Figure 1 together with Algorithm 1. For simplicity, we assume $|P| = |C| = 1$ and consider the three sets consisting of exactly one element. We have

$$\begin{array}{rcl} \{a\} \parallel_{dom} \{b\} & \Rightarrow & \{a\} \preceq_{Alg_1} \{b\} \wedge \{b\} \preceq_{Alg_1} \{a\} \\ \{b\} \succ_{dom} \{c\} & \Rightarrow & \{b\} \succ_{Alg_1} \{c\} \\ \{a\} \parallel_{dom} \{c\} & \Rightarrow & \{a\} \preceq_{Alg_1} \{c\} \wedge \{a\} \preceq_{Alg_1} \{c\} \end{array}$$

The relation $\preceq_{Alg_1}$ contains a deteriorative cycle as

$$\{b\} \succeq_{Alg_1} \{c\} \succeq_{Alg_1} \{a\} \succeq_{Alg_1} \{b\}.$$

and

$$\{b\} \succ_{dom} \{c\}.$$

In the following, we point out how Algorithm 1 may produce this cycle during the optimization process. Assume that the algorithm starts with the population $P_1 = \{b\}$ and produces the first offspring $c$. Due to Pareto-dominance $\{c\} \prec_{Alg_1} P_1$ holds, and $P_2 = \{c\}$ becomes the new parent population. The offspring of $P_2$ is $a$ and {a} is incomparable to $P_2$ and therefore $\{a\} \preceq_{Alg_1} P_2$. Hence, $P_3 = \{a\}$ may be the new parent population. Similar, $b$ may be the next offspring and the set $\{b\} = P_1$ is incomparable to $P_3$ ($\{b\} \preceq_{Alg_1} P_3$) such that the algorithm may proceed to $P_4 = \{b\} = P_1$ creating a deteriorative cycle.

## 4.2 Coping with deteriorative cycles

We have seen in the previous section that using just the preference relation on sets and allowing moves between incomparable sets may lead to a deteriorative cyclic behavior. This is due to the fact that evolutionary multi-objective optimization has to deal with incomparable sets. An algorithm just relying on the generalized dominance relation on sets and selecting arbitrarily between incomparable sets may accept a set of search points $A$ that is strongly dominated by another set $B$ that has been obtained before $A$.

In the following, we want to examine how to deal with the cyclic behavior. We want to discuss the properties for $\preceq$ such that it does not contain a deteriorative cycle. The first property is that it is compliant with the Pareto dominance relation.

DEFINITION 4.2. *A relation* $\preceq$ *on* $2^X$ *is Pareto-compliant if*

$$A \prec_{dom} B \Rightarrow A \prec B.$$

*A relation* $\preceq$ *is strictly Pareto-compliant if additionally*

$$A \equiv_{dom} B \Rightarrow A \equiv B.$$

This definition slightly deviates from the use of the terms Pareto-compliant and strictly Pareto-compliant in [6, 10]. Note that if $\preceq$ is strictly Pareto-compliant, then

$$A \preceq_{dom} B \Rightarrow A \preceq B.$$

Definition 4.2 states that the relation $\preceq$ is Pareto-compliant if the indicator can also distinguish between sets that strictly dominate each other. Note that $\preceq$ is Pareto compliant iff it is a refinement of the Pareto dominance relation $\preceq_{dom}$ on sets in terms of Zitzler et al. [13]. We have seen that just the property to have a Pareto-compliant indicator does not avoid cyclic behavior as the Pareto-dominance relation is itself Pareto-compliant. To avoid cyclic behavior we need an additional property.

Definition 4.3. *A relation $\preceq$ on $2^X$ is called transitive if*

$$\big((A \preceq B) \wedge (B \preceq C)\big) \Rightarrow A \preceq C.$$

In the following we show that a relation $\preceq$ that is Pareto compliant and transitive does not contain a deteriorative cycle. We will use this property later on to show which algorithms do not encounter a deteriorative cycle.

Theorem 4.4. *If $\preceq$ is Pareto-compliant and transitive then $\preceq$ does not contain a deteriorative cycle.*

Proof. We prove the theorem by contradiction. Assume that $\preceq$ contains a deteriorative cycle consisting of sets $A_1, A_2, \ldots, A_r \in 2^X$ with

$$A_1 \preceq A_2 \preceq \ldots \preceq A_{r-1} \preceq A_r \preceq A_1$$

and $A_r \prec_{dom} A_1$.

By transitivity of $\preceq$, we get $A_1 \preceq A_r$. As $\preceq$ is also Pareto-compliant, we know from $A_r \prec_{dom} A_1$ that $A_r \prec A_1$ and hence by definition of $\prec$, $A_1 \not\preceq A_r$ which contradicts $A_1 \preceq A_r$. Hence $\preceq$ cannot contain a deteriorative cycle. □

We want to modify Algorithm 1 such that the underlying relation does not contain a deteriorative cycle. Algorithm 4 differs from Algorithm 1 by using an additional relation $\preceq_I$ in the case that $P' \parallel_{dom} P$ holds.

Algorithm 4 (Cycle-free Optimizer).

1. *Choose an initial population $P$ consisting of $\mu$ individuals.*
2. *Produce an offspring population $C$.*
3. *Compute a minimal set $P' \in Min_\mu(2^{(P \cup C)}, \preceq_{dom})$ using Algorithm 2 or 3.*
4. *If $(P' \prec_{dom} P) \vee (P' \equiv_{dom} P) \vee (P' \preceq_I P)$ then $P := P'$.*
5. *If no stopping criteria is fulfilled, go to step 2*

Step 4 should be read like a short-circuit evaluation of Boolean operators in most modern programming languages. That is, if $P' \prec_{dom} P$ or $P' \equiv_{dom} P$, the algorithm can decide to choose $P'$ without calculating the (usually much more expensive) relation $\preceq_I$. This way the algorithm works on the underlying relation $\preceq_{Alg_4}$ given by

$$A \preceq_{Alg_4} B :\Leftrightarrow (A \preceq_{dom} B) \vee ((A \parallel_{dom} B) \wedge (A \preceq_I B)).$$

If $\preceq_I$ is strictly Pareto compliant then $\preceq_{Alg_4} = \preceq_I$. If $\preceq_I$ is in addition transitive, then $\preceq_{Alg_4} = \preceq_I$ does not contain a deteriorative cycle. We state this property in the following corollary.

Corollary 4.5. *If $\preceq_I$ is strictly Pareto compliant and transitive then $\preceq_{Alg_4}$ does not contain a deteriorative cycle.*

# 5. UNARY INDICATORS

We now want to examine the common approach to define refinements via indicator functions (see e. g. [13]). In this section we focus on unary indicator functions while the next section examines binary indicator functions. Unary indicator functions assign each set a real number that somehow reflects their quality, i.e.,

$$I_1 \colon 2^X \to \mathbb{R}.$$

To define a relation based on an indicator function, we use the following definition.

Definition 5.1. *For an unary indicators $I_1$ we set*

$$A \preceq_{I_1} B :\Leftrightarrow I_1(A) \leq I_1(B),$$
$$A \prec_{I_1} B :\Leftrightarrow I_1(A) < I_1(B).$$

Note that the relation $\preceq_{I_1}$ is total and also transitive as the order on real values forms a transitive relation.

Also observe the following simple property which follows directly from Theorem 4.4 and Corollary 4.5.

Lemma 5.2. *Let $I$ be an unary indicator.*

- *If the corresponding relation $\preceq_I$ is Pareto-compliant, then $\preceq_I$ contains no deteriorative cycles.*
- *If the corresponding relation $\preceq_I$ is strictly Pareto-compliant, then $\preceq_{Alg_4} = \preceq_I$ contains no deteriorative cycles.*

This shows that unary indicators can help to avoid in a natural way the problem of cyclic behavior and why the property of Pareto-compliance is especially important for unary indicators. Unfortunately, there is currently only one unary indicator known which is Pareto-compliant. This is the hypervolume indicator. For minimization problems it it measures the volume of the dominated portion of the objective space relative to a fixed reference point $(R_1, R_2, \ldots, R_d) \in \mathbb{R}^d$ which lies above the Pareto front. In our setting where an indicator should be minimized, the hypervolume indicator of a set of solutions $A \in 2^X$ can be defined as

$$I_{\mathrm{HYP}}(A) := -\textsc{vol}\left(\bigcup_{x \in A} [f_1(x), R_1] \times \ldots \times [f_d(x), R_d]\right)$$

with $\textsc{vol}(\cdot)$ being the usual Lebesgue measure. The hypervolume indicator was first introduced for performance assessment in multiobjective optimization by Zitzler and Thiele [8] and hypervolume-based optimizers have become very popular in recent years (see e.g. [1, 5, 11]).

The problem with the hypervolume indicator is that it is computationally expensive, i. e., the runtime for the computation of the hypervolume for a given set of search points grows exponentially with the number of objectives [2]. Compared to this the test whether $A \preceq_{dom} B$ holds can always be done in time polynomial in the size of the given two sets and the number of objectives. The following theorem describes another nice property of the hypervolume indicator.

Theorem 5.3. *Let $A, B \in 2^X$. If $A \equiv_{dom} B$ then $\mathrm{HYP}(A) = \mathrm{HYP}(B)$ holds.*

PROOF. Let $A \equiv_{dom} B$. Then Lemma 2.3 implies $Min(f(A), \preceq_{Par}) = Min(f(B), \preceq_{Par})$. As the hypervolume of a given set of points is only determined by its set of minimal elements in the objective space, this implies $\mathrm{HYP}(A) = \mathrm{HYP}(B)$. $\square$

Note that the above theorem actually not only holds for HYP, but for all unary indicators whose value only depends on the minimal elements in the objective space.

# 6. BINARY INDICATORS

Though the hypervolume indicator is the only Pareto-compliant unary indicator, there are several Pareto-compliant binary indicators. This could give the hope to find a Pareto-compliant indicator which also contains no deteriorative cycles, but is computationally not as expensive as the hypervolume indicator.

Binary indicator functions assign pairs of sets a real number that somehow reflects their relative performance, i.e.,

$$I_2 \colon 2^X \times 2^X \to \mathbb{R}.$$

To define a relation based on an binary indicator function, we use the following definition.

DEFINITION 6.1. *For a binary indicators $I_2$ we set*

$$A \preceq_{I_2} B :\Leftrightarrow I_2(A,B) \leq I_2(B,A),$$
$$A \prec_{I_2} B :\Leftrightarrow I_2(A,B) < I_2(B,A).$$

Zitzler et al. [10] mentiones four Pareto-compliant binary indicators: the multiplicative $\varepsilon$-indicator, the additive $\varepsilon$-indicator, the coverage indicator [8], and the binary hypervolume indicator. However, binary indicators are not transitive in general. Therefore there is no equivalent of Lemma 5.2 for binary indicators.

## 6.1 $\varepsilon$-Indicator

In the following, we focus on the $\varepsilon$-indicators which are very popular in evolutionary multi-objective optimization. We follow the definitions of Zitzler et al. [10] for the multiplicative and additive $\varepsilon$-dominance relation.

DEFINITION 6.2. *A search point $a \in X$ is said to multiplicatively $\varepsilon$-dominate another search point $b \in X$ written as $a \preceq_{\varepsilon*} b$, if and only if*

$$f_i(a) \leq \varepsilon\, f_i(b) \qquad \text{for all } 1 \leq i \leq n.$$

*The binary multiplicative $\varepsilon$-indicator $I_{\varepsilon*}$ on $2^X \times 2^X$ is*

$$I_{\varepsilon*}(A,B) := \max_{b\in B} \min_{a\in A} \max_{1\leq i\leq n} f_i(a)/f_i(b).$$

DEFINITION 6.3. *A search point $a \in X$ is said to additively $\varepsilon$-dominate another search point $b \in X$ written as $a \preceq_{\varepsilon*} b$, if and only if*

$$f_i(a) \leq \varepsilon + f_i(b) \qquad \text{for all } 1 \leq i \leq n.$$

*The binary additive $\varepsilon$-indicator $I_{\varepsilon+}$ on '$2^X \times 2^X$ is*

$$I_{\varepsilon+}(A,B) := \max_{b\in B} \min_{a\in A} \max_{1\leq i\leq n} f_i(a) - f_i(b).$$

We will analyze the corresponding relations $\preceq_{I_{\varepsilon*}}$ and $\preceq_{I_{\varepsilon*}}$ as defined in Definition 6.1. To simplify the notation we set

$$\preceq_{\varepsilon*} := \preceq_{I_{\varepsilon*}} \quad \text{and} \quad \preceq_{\varepsilon+} := \preceq_{I_{\varepsilon+}}.$$

Let further $\prec_{\varepsilon*}$ and $\prec_{\varepsilon+}$ have their obvious meaning. We can show that $\preceq_{\varepsilon*}$ and $\preceq_{\varepsilon+}$ are strictly Pareto-compliant.

LEMMA 6.4. *$\preceq_{\varepsilon*}$ and $\preceq_{\varepsilon+}$ are strictly Pareto-compliant.*

PROOF. Let $A \prec_{dom} B$. Then by Definition 6.2, $I_{\varepsilon*}(A,B) = 1$ and $I_{\varepsilon*}(B,A) > 1$ and therefore $A \prec_{\varepsilon*} B$. Analogously by Definition 6.2, $I_{\varepsilon+}(A,B) = 0$ and $I_{\varepsilon+}(B,A) > 0$ and therefore $A \prec_{\varepsilon+} B$. This shows that $\preceq_{\varepsilon*}$ and $\preceq_{\varepsilon+}$ are Pareto-compliant.

In order to prove that they are also strictly Pareto-compliant, let $A \equiv_{dom} B$. Then by Lemma 2.3, $Min(f(A), \preceq_{Par}) = Min(f(B), \preceq_{Par})$ and therefore $I_{\varepsilon*}(A,B) = I_{\varepsilon*}(B,A) = 1$ and $I_{\varepsilon+}(A,B) = I_{\varepsilon+}(B,A) = 0$, which is equivalent to $A \equiv_{\varepsilon*} B$ and $A \equiv_{\varepsilon+} B$. $\square$

This shows that $\preceq_{\varepsilon*}$ and $\preceq_{\varepsilon+}$ are strictly Pareto-compliant. Therefore by Corollary 4.5 it suffices to show that they are also transitive in order to prove that they contain no deteriorative cycle. Unfortunately, in most cases this does not hold.

In the remainder of this Section 6 we examine for what sets the relations $\preceq_{\varepsilon*}$ and $\preceq_{\varepsilon+}$ contain deteriorative cycles. More precisely, we will prove the following dichotomy.

THEOREM 6.5. *Let $s \geq 1$ and $d \geq 2$. Then the relations $\preceq_{\varepsilon*}$ and $\preceq_{\varepsilon+}$ restricted to sets of size $\leq s$ in $d$ dimensions do* not *contain deteriorative cycles if and only if $s = 1$ and $d = 2$.*

This implies that the relations $\preceq_{\varepsilon*}$ and $\preceq_{\varepsilon+}$ contain no deteriorative cycles only in the simplest case of singleton sets in two dimensions. We prove Theorem 6.5 in the following Lemmas 6.6, 6.7, and 6.8.

## 6.2 Nonexistence of deteriorative cycles for sets of size one in two dimensions

We first give a positive result and show that the relations $\preceq_{\varepsilon*}$ and $\preceq_{\varepsilon+}$ do not contain deteriorative cycles if we restrict it to singleton sets in two dimensions.

LEMMA 6.6. *For sets of size one in two dimensions, the relations $\preceq_{\varepsilon*}$ and $\preceq_{\varepsilon+}$ do not contain deteriorative cycles.*

PROOF. According to Theorem 4.4 and Lemma 6.4, it suffices to show that $\preceq_{\varepsilon*}$ and $\preceq_{\varepsilon+}$ are transitive. Let us consider three arbitrary points $a, b, c \in X$ with $f(a) = (a_x, a_y)$, $f(b) = (b_x, b_y)$ and $f(c) = (c_y, c_y)$. We want to show that

$$\big((\{a\} \preceq_{\varepsilon*} \{b\}) \wedge (\{b\} \preceq_{\varepsilon*} \{c\})\big) \Rightarrow \{a\} \preceq_{\varepsilon*} \{c\}.$$

By definition,

$$\begin{aligned}
&\{a\} \preceq_{\varepsilon*} \{b\}\\
&\Leftrightarrow I_{\varepsilon*}(\{a\},\{b\}) \leq I_{\varepsilon*}(\{b\},\{a\})\\
&\Leftrightarrow \left(\max\left\{\frac{a_x}{b_x}, \frac{a_y}{b_y}\right\} \leq \max\left\{\frac{b_x}{a_x}, \frac{b_y}{a_y}\right\}\right)\\
&\Leftrightarrow \underbrace{\left(\frac{a_x}{b_x} \leq \frac{b_x}{a_x} \wedge \frac{a_y}{b_y} \leq \frac{b_x}{a_x}\right)}_{(1)} \vee \underbrace{\left(\frac{a_x}{b_x} \leq \frac{b_y}{a_y} \wedge \frac{a_y}{b_y} \leq \frac{b_y}{a_y}\right)}_{(2)}.
\end{aligned}$$

Analogously,

$$
\begin{aligned}
&\{b\} \preceq_{\varepsilon*} \{c\}\\
&\Leftrightarrow I_{\varepsilon*}(\{b\},\{c\}) \le I_{\varepsilon*}(\{c\},\{b\})\\
&\Leftrightarrow \left(\max\left\{\frac{b_x}{c_x},\frac{b_y}{c_y}\right\} \le \max\left\{\frac{c_x}{b_x},\frac{c_y}{b_y}\right\}\right)\\
&\Leftrightarrow \underbrace{\left(\frac{b_x}{c_x} \le \frac{c_x}{b_x} \wedge \frac{b_y}{c_y} \le \frac{c_x}{b_x}\right)}_{(3)} \vee \underbrace{\left(\frac{b_x}{c_x} \le \frac{c_y}{b_y} \wedge \frac{b_y}{c_y} \le \frac{c_y}{b_y}\right)}_{(4)}.
\end{aligned}
$$

Plugging both together yields,

$$
\begin{aligned}
&(\{a\} \preceq_{\varepsilon*} \{b\}) \wedge (\{b\} \preceq_{\varepsilon*} \{c\})\\
&\Leftrightarrow \big((1)\wedge(3)\big) \vee \big((1)\wedge(4)\big) \vee \big((2)\wedge(3)\big) \vee \big((2)\wedge(4)\big)\\
&\Rightarrow \big((1)\wedge(3)\big) \vee \big((2)\wedge(4)\big)\\
&\Leftrightarrow \left(\frac{a_x}{b_x} \le \frac{b_x}{a_x} \wedge \frac{a_y}{b_y} \le \frac{b_x}{a_x} \wedge \frac{b_x}{c_x} \le \frac{c_x}{b_x} \wedge \frac{b_y}{c_y} \le \frac{c_x}{b_x}\right) \vee\\
&\quad \left(\frac{a_x}{b_x} \le \frac{b_y}{a_y} \wedge \frac{a_y}{b_y} \le \frac{b_y}{a_y} \wedge \frac{b_x}{c_x} \le \frac{c_y}{b_y} \wedge \frac{b_y}{c_y} \le \frac{c_y}{b_y}\right)\\
&\Rightarrow \left(\frac{a_x}{c_x} \le \frac{c_x}{a_x} \wedge \frac{a_y}{c_y} \le \frac{c_x}{a_x}\right) \vee \left(\frac{a_x}{c_x} \le \frac{c_y}{a_y} \wedge \frac{a_y}{c_y} \le \frac{c_y}{a_y}\right)\\
&\Leftrightarrow \left(\max\left\{\frac{a_x}{c_x},\frac{a_y}{c_y}\right\} \le \max\left\{\frac{c_x}{a_x},\frac{c_y}{a_y}\right\}\right)\\
&\Leftrightarrow I_{\varepsilon*}(\{a\},\{c\}) \le I_{\varepsilon*}(\{c\},\{a\})\\
&\Leftrightarrow \{a\} \preceq_{\varepsilon*} \{c\}.
\end{aligned}
$$

The proof for the additive $\varepsilon$-relation is equivalent and can be obtained by replacing all fractions $\frac{\alpha}{\beta}$ in the proof above with subtractions $\alpha - \beta$. $\square$

## 6.3 Existence of deteriorative cycles for sets of size one in three dimensions

In Lemma 6.6 we showed that for sets of size one in two dimensions the relations $\preceq_{\varepsilon*}$ and $\preceq_{\varepsilon+}$ do not contain deteriorative cycles. We now prove that already for three dimensions this does not hold anymore.

Lemma 6.7. *For sets of size one in more than two dimensions the relations $\preceq_{\varepsilon*}$ and $x \preceq_{\varepsilon+}$ contain deteriorative cycles.*

Proof. We choose four points $a, b, c, d \in X$ with $f(a) := (1, 2, 4)$, $f(b) := (1, 1, 4)$, $f(c) := (3, 3, 1)$, $f(d) := (4, 1, 2)$ and show that these four points build a deteriorative cycle for both $\varepsilon$-dominance relation. Let us first look at the multiplicative $\varepsilon$-dominance relation. As we are looking at minimization problems, $b$ clearly dominates $a$, that is, $\{b\} \prec_{dom} \{c\}$. On the other hand, $\{c\} \prec_{\varepsilon*} \{b\}$ as

$$I_{\varepsilon*}(\{c\},\{b\}) = 3 < 4 = I_{\varepsilon*}(\{b\},\{c\}).$$

Also $\{d\} \prec_{\varepsilon*} \{c\}$ as

$$I_{\varepsilon*}(\{d\},\{c\}) = 2 < 3 = I_{\varepsilon*}(\{c\},\{d\}).$$

The deteriorative cycle is then closed by $\{a\} \prec_{\varepsilon*} \{d\}$ as

$$I_{\varepsilon*}(\{a\},\{d\}) = 2 < 4 = I_{\varepsilon*}(\{d\},\{a\}).$$

Overall, $\{a\} \prec_{\varepsilon*} \{d\} \prec_{\varepsilon*} \{c\} \prec_{\varepsilon*} \{b\} \prec_{dom} \{a\}$.

It remains to show that the same points form a deteriorative cycle for the additive $\varepsilon$-dominance relation. It is easy to that

$$
\begin{aligned}
I_{\varepsilon+}(\{c\},\{b\}) &= 2 < 3 = I_{\varepsilon+}(\{b\},\{c\}),\\
I_{\varepsilon+}(\{d\},\{c\}) &= 1 < 2 = I_{\varepsilon+}(\{c\},\{d\}),\\
I_{\varepsilon+}(\{a\},\{d\}) &= 2 < 3 = I_{\varepsilon+}(\{d\},\{a\}),
\end{aligned}
$$

and therefore $\{a\} \prec_{\varepsilon+} \{d\} \prec_{\varepsilon+} \{c\} \prec_{\varepsilon+} \{b\} \prec_{dom} \{a\}$, which finishes to proof. $\square$

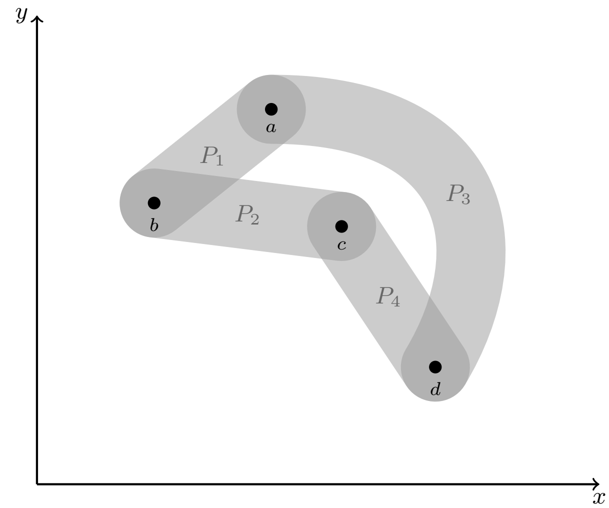


**Figure 2: Illustration of the sets used in the proof of Lemma 6.8.**

## 6.4 Existence of deteriorative cycles for sets of size two in two dimensions

To complete the dichotomy of the relations $\preceq_{\varepsilon*}$ and $\preceq_{\varepsilon+}$, it remains to prove that also in two dimensions sets of size more than one give deteriorative cycles.

Lemma 6.8. *For sets of size more than one in two dimensions, the relations $\preceq_{\varepsilon*}$ and $x \preceq_{\varepsilon+}$ contain deteriorative cycles.*

Proof. We choose $a, b, c, d \in X$ such that

$$
\begin{aligned}
P_1 &:= \{a, b\}, \quad \text{with} & f(a) &:= (10, 16),\\
P_2 &:= \{c, b\}, & f(b) &:= (5, 12),\\
P_3 &:= \{a, d\}, & f(c) &:= (13, 11),\\
P_4 &:= \{d, c\}, & f(d) &:= (17, 5).
\end{aligned}
$$

We want to show that these four sets $P_1$, $P_2$, $P_3$, $P_4$ form a deteriorative cycle. It is easy to see that $P_1 \prec_{dom} P_2$. We first examine the multiplicative $\varepsilon$-dominance relation and prove that $P_1 \prec_{dom} P_2 \prec_{\varepsilon*} P_3 \prec_{\varepsilon*} P_4 \prec_{\varepsilon*} P_1$.

Observe that $P_2 \prec_{\varepsilon*} P_3$ holds as

$$
\begin{aligned}
I_{\varepsilon*}(P_3, P_2) &= \max\{17/13, 2\} = 2\\
&< 2.2 = \max\{3/4, 11/5\} = I_{\varepsilon*}(P_2, P_3),
\end{aligned}
$$

and $P_3 \prec_{\varepsilon*} P_4$ holds as

$$
\begin{aligned}
I_{\varepsilon*}(P_4, P_3) &= \max\{13/10, 1\} = 1.3\\
&< 1.308... = \max\{1, 17/13\} = I_{\varepsilon*}(P_3, P_4).
\end{aligned}
$$

It remains to show $P_4 \prec_{\varepsilon*} P_1$, which holds since

$$I_{\varepsilon*}(P_1, P_4) = \max\{12/5, 12/11\} = 2.4$$
$$< 2.6 = \max\{13/10, 13/5\} = I_{\varepsilon*}(P_4, P_1).$$

This shows the claim for the multiplicative $\varepsilon$-dominance relation. We now prove that the same sets also contains a deteriorative cycle in the case of the additive $\varepsilon$-dominance relation. We have $P_2 \prec_{\varepsilon*} P_3$ as

$$I_{\varepsilon+}(P_3, P_2) = \max\{4, 5\} = 5$$
$$< 6 = \max\{-4, 6\} = I_{\varepsilon+}(P_2, P_3),$$

and $P_3 \prec_{\varepsilon*} P_4$ as

$$I_{\varepsilon+}(P_4, P_3) = \max\{3, 0\} = 3$$
$$< 4 = \max\{0, 4\} = I_{\varepsilon+}(P_3, P_4),$$

and $P_4 \prec_{\varepsilon*} P_1$ as

$$I_{\varepsilon+}(P_1, P_4) = \max\{7, 1\} = 7$$
$$< 8 = \max\{3, 8\} = I_{\varepsilon+}(P_4, P_1).$$

This shows $P_1 \prec_{dom} P_2 \prec_{\varepsilon+} P_3 \prec_{\varepsilon+} P_4 \prec_{\varepsilon+} P_1$ and finishes the proof. □

# 7. CONCLUSIONS

Evolutionary algorithms for multi-objective optimization search for a set of search points that is minimal with respect to the Pareto dominance relation on sets. This optimization goal has been made explicit recently in Zitzler et al. [13]. With this paper, we have contributed to the theoretical understanding of this optimization process by investigating the underlying relation that an evolutionary algorithm uses for optimization. First, we have shown how to choose a minimal set among the parents and children to build the next parent population. Our algorithms are similar to the method used in NSGA-II and SPEA2 and allow to incorporate preferences into the computation of a minimal set.

Later on, we have pointed out that algorithms which are solely based on the Pareto dominance relation may encounter deteriorative cycles if they can move between incomparable sets. This is due to the fact that the Pareto dominance relation is not a total relation. We have examined how such cycles can be avoided by using indicator functions on incomparable sets. Our studies show that if the total relation on which an algorithm works is Pareto compliant and transitive then the relation does not contain a deteriorative cycle. Investigating the binary $\varepsilon$-indicator which is Pareto compliant, we have shown that it is transitive only for very restricted cases and may lead to deteriorative cycles in general. Unary indicators are in a natural way transitive and therefore each unary indicator that is Pareto compliant fulfills our conditions. An indicator matching the desired properties is the hypervolume indicator. Therefore, our studies give a further justification for using this indicator that has become very popular in evolutionary multi-objective optimization. In remains an open problem to construct further unary indicators that are Pareto compliant and transitive as such indicator help to guide the search with respect to the Pareto dominance relation on sets.

# References


[1] N. Beume, B. Naujoks, and M. T. M. Emmerich. SMS-EMOA: Multiobjective selection based on dominated hypervolume. *European Journal of Operational Research*, 181(3):1653–1669, 2007.

[2] K. Bringmann and T. Friedrich. Approximating the volume of unions and intersections of high-dimensional geometric objects. In *Proc. 19th International Symposium on Algorithms and Computation (ISAAC '08)*, volume 5369 of *LNCS*, pages 436–447, 2008.

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

[4] K. Deb, S. Agrawal, A. Pratap, and T. Meyarivan. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Trans. Evolutionary Computation*, 6 (2):182–197, 2002.

[5] C. Igel, N. Hansen, and S. Roth. Covariance matrix adaptation for multi-objective optimization. *Evolutionary Computation*, 15(1):1–28, 2007.

[6] 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, Feb. 2006.

[7] M. Laumanns, L. Thiele, K. Deb, and E. Zitzler. Combining convergence and diversity in evolutionary multiobjective optimization. *Evolutionary Computation*, 10(3):263–282, 2003.

[8] E. Zitzler and L. Thiele. Multiobjective evolutionary algorithms: a comparative case study and the strength Pareto approach. *IEEE Trans. Evolutionary Computation*, 3(4):257–271, 1999.

[9] E. Zitzler, M. Laumanns, and L. Thiele. SPEA2: Improving the strength Pareto evolutionary algorithm for multiobjective optimization. In *Proc. of EUROGEN 2001*, pages 95–100. International Center for Numerical Methods in Engineering (CIMNE), 2002.

[10] E. Zitzler, L. Thiele, M. Laumanns, C. M. Fonseca, and V. Grunert da Fonseca. Performance assessment of multiobjective optimizers: an analysis and review. *IEEE Trans. Evolutionary Computation*, 7(2):117–132, 2003.

[11] E. Zitzler, D. Brockhoff, and L. Thiele. The hypervolume indicator revisited: On the design of Pareto-compliant indicators via weighted integration. In *Proc. Fourth International Conference on Evolutionary Multi-Criterion Optimization (EMO '07)*, volume 4403 of *LNCS*, pages 862–876. Springer, 2007.

[12] E. Zitzler, L. Thiele, and J. Bader. SPAM: Set preference algorithm for multiobjective optimization. In *Proc. of 10th International Conference on Parallel Problem Solving from Nature (PPSN X)*, volume 5199 of *LNCS*, pages 847–858. Springer, 2008.

[13] E. Zitzler, L. Thiele, and J. Bader. On Set-Based Multiobjective Optimization. *IEEE Transactions on Evolutionary Computation*, 2010+. To appear.