A Tutorial on the Performance Assessment of Stochastic Multiobjective Optimizers

Joshua D. Knowles, Lothar Thiele, and Eckart Zitzler*

TIK-Report No. 214
Computer Engineering and Networks Laboratory,
ETH Zurich
Gloriastrasse 35, ETH-Zentrum, 8092 Zurich, Switzerland

Revised Version
February 2006

(First Version: July 2005)

Abstract

Identifying or approximating the set of Pareto-optimal solutions is beneficial in many application domains, and with the availability of sufficient computing resources various randomized search heuristics have been proposed for this purpose. Much of the empirical research carried out to this end relies heavily on methods for evaluating and comparing the performance of different stochastic multiobjective optimizers. While visual comparisons were common in the infancy of evolutionary multiobjective optimization, quantitative performance assessment is now becoming more standard. However, no guidelines are available on how to compare the quality of the outcomes generated by several multiobjective optimizers, over several runs, to obtain quantitative and statistically sound inferences. As a consequence, most comparative studies are based on different methodologies and assumptions and therefore the results are difficult to relate to one other.

This paper reviews the current state of the art in statistical performance assessment of stochastic multiobjective optimizers based on the concept of Pareto dominance, derives general recommendations, and demonstrates the implementation of the guidelines suggested in the context of a case study. In addition, corresponding software tools are provided that are free for download.

*Joshua D. Knowles is with the School of Chemistry, University of Manchester, UK (j.knowles@manchester.ac.uk), and Lothar Thiele and Eckart Zitzler are with the Computer Engineering and Networks Laboratory (TIK) at the Swiss Federal Institute of Technology (ETH) Zurich, Switzerland ({thiele, zitzler}@tik.ee.ethz.ch).

1 Motivation

In the last decade, there has been a growing interest in applying randomized search algorithms such as evolutionary algorithms, simulated annealing, and tabu search to multiobjective optimization problems in order to approximate the set of Pareto-optimal solutions. Various methods have been proposed for this purpose, and their usefulness has been demonstrated in several application domains, cf. (Deb 2001; Coello Coello et al. 2002).

With the rapid increase of the number of available techniques, the issue of performance assessment has become more and more important and has developed into an independent research topic. As with single objective optimization, the notion of performance involves both the quality of the solution found and the time to generate such a solution. The difficulty is that in the case of stochastic optimizers the relationship between quality and time is not fixed, but described by a corresponding probability density function. Accordingly, every statement about the performance of a randomized search algorithm is probabilistic in nature. Another difficulty is particular to multiobjective optimizers that aim at approximating the set of Pareto-optimal solutions in a scenario with multiple criteria: the outcome of the optimization process is usually not a single solution but a set of trade-offs. This not only raises the question of how to define quality in this context, but also how to represent the outcomes of multiple runs in terms of a probability density function. At the moment there is no common agreement in the community on how to deal with the latter two issues.

In principle, two basic approaches exist in the literature: the attainment function approach, which models the outcome of a multiobjective optimizer as a probability density function in the objective space, and the indicator approach, which summarizes the outcome of a run on the basis of quantitative performance measures and performs the statistical analysis on the corresponding distribution of performance values. It has remained unclear so far how these approaches are related to each other and what their advantages and disadvantages are. Moreover, there is an ongoing discussion in the community about the design and the choice of appropriate performance measures, and some recent studies show that special care is required in this respect (Knowles and Corne 2002; Okabe et al. 2003; Zitzler et al. 2003). Statistical testing similarly requires special attention because although standard procedures can usually be applied, the multiplicity of ways of summarising the outcome of an optimizer run raises issues about multiple testing and combined inferences. As a result, clear guidelines on how to compare the performance of stochastic multiobjective optimizers systematically, with statistical rigour, have not been available so far.

This paper tries to fill this gap: it summarizes the state of the art in performance assessment of stochastic multiobjective optimizers and derives general recommendations for optimization scenarios that are based on the concept of Pareto dominance. In particular, the two methodological approaches mentioned above will be reviewed in a common framework and related to each other, and in addition a third approach based on dominance-based ranking will be presented. Moreover, a set of software tools has been developed by which the suggested guidelines can be easily implemented in practice; the source code and the binaries for different platforms are available for free download. Note that this review focuses on the aspects of performance assessment that are particular to multiobjective optimization. Many other aspects that also arise in a single-objective context such as setting the parameters of the algorithms, choosing appropriate benchmark problems, integrating the notion of time, etc. will not be covered.

2 General Considerations

Before discussing the comparison methodologies in detail, we will define the basic setting considered in the remainder of this paper and briefly summarize the underlying concepts.

2.1 Basic Terms

2.1.1 Pareto Dominance and Optimality in the Context of Order Theory

A general optimization problem can be considered as a quadruple $(X, Z, \boldsymbol{f}, rel)$ where X denotes the search space or *decision space*, Z represents the *objective space*, $\boldsymbol{f} : X \to Z$ is a function that assigns to each solution or *decision vector* $\boldsymbol{x} \in X$ a corresponding *objective vector* $\boldsymbol{z} = \boldsymbol{f}(\boldsymbol{x}) \in Z$, and *rel* represents a binary relation over Z that defines a partial order of the objective space, which in turn induces a preorder of

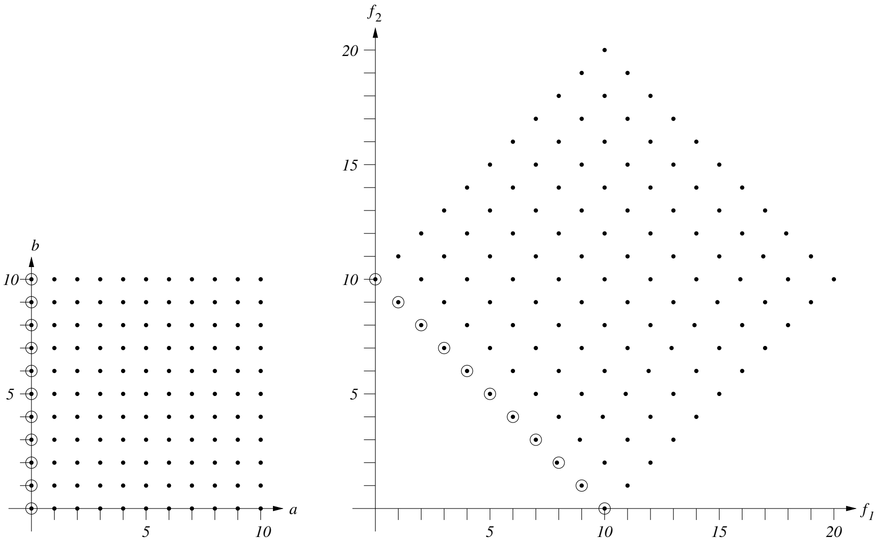


Figure 1: Decision space (left) and objective space (right) for Example 1. The Pareto-optimal decision resp. objective vectors are marked by circles.

the decision space.[1] The goal is to find a solution $\boldsymbol{x}^* \in X$ that is mapped to a minimal element $\boldsymbol{z}^* = \boldsymbol{f}(\boldsymbol{x}^*)$ of $\boldsymbol{f}(X) = \{\boldsymbol{z} \in Z \,|\, \exists \boldsymbol{x} \in X : \boldsymbol{z} = \boldsymbol{f}(\boldsymbol{x})\}$ which is a subset of the partially ordered set (Z, rel).[2]

Usually, $\boldsymbol{f}$ consists of one or several *objective functions* $f_1, f_2, \ldots, f_n$ with $\boldsymbol{f} = (f_1, \ldots, f_n)$, and we here assume without loss of generality that $Z = \mathbb{R}^n$ and that all $f_i : X \to \mathbb{R}$ with $1 \leq i \leq n$ are to be minimized.

In the presence of a single objective function ($n = 1$), the standard relation 'less than or equal' is generally used to define the corresponding minimization problem $(X, \mathbb{R}, f_1, \leq)$. Since $\leq$ is a total order over $Z = \mathbb{R}$, there always exists a unique optimal value $\boldsymbol{z}^* \in \boldsymbol{f}(X)$, i.e., the minimal element of $\boldsymbol{f}(X)$ is unique.[3] Nevertheless, there may be several solutions that are all mapped to $\boldsymbol{z}^*$; therefore, the corresponding relation $\leq$ on the decision space with $\boldsymbol{x}^1 \leq \boldsymbol{x}^2 \Leftrightarrow \boldsymbol{f}(\boldsymbol{x}^1) \leq \boldsymbol{f}(\boldsymbol{x}^2)$ is a preorder, but not necessarily a partial order.

In the case of multiple objective functions, i.e., $n > 1$, usually the relation $\preceq$ with $\boldsymbol{z}^1 \preceq \boldsymbol{z}^2 \Leftrightarrow \forall i \in \{1, \ldots, n\} : \boldsymbol{z}^1_i \leq \boldsymbol{z}^2_i$ is taken; it represents a natural extension of $\leq$ to $\mathbb{R}^n$ and is also known as *weak Pareto dominance*. The associated strict order $\prec$ with $\boldsymbol{z}^1 \prec \boldsymbol{z}^2 \Leftrightarrow \boldsymbol{z}^1 \preceq \boldsymbol{z}^2 \wedge \neg\, \boldsymbol{z}^2 \preceq \boldsymbol{z}^1$ is often denoted as *Pareto dominance*, and instead of $\boldsymbol{z}^1 \prec \boldsymbol{z}^2$ one also says $\boldsymbol{z}^1$ *dominates* $\boldsymbol{z}^2$. In addition to these two relations, we will also use the terms 'strictly dominates', 'incomparable', and 'indifferent' in the remainder of this paper as defined in Table 1. The difference to the single objective case now is based on the fact that multiple minimal elements of $\boldsymbol{f}(X)$ may emerge, each representing a different trade-off between the objectives. The minimal elements are called *Pareto optimal*, and accordingly a decision vector is Pareto optimal iff it is mapped to a Pareto optimal objective vector. The entirety of all Pareto-optimal objective vectors is denoted as *Pareto-optimal front* and the set of all Pareto-optimal solutions constitutes the *Pareto-optimal set.* Again, since several solutions may be mapped to the same objective vector, the Pareto-optimal front does not necessarily contain as many elements as the Pareto-optimal set.

Example 1 *Consider the following minimization problem where the decision space consists of all pairs* $(a, b) \in \{0, \ldots, r\} \times \{0, \ldots, r\}$ *and the objective space is defined by the two functions* $f_1(a, b) = a + b$ *and* $f_2(a, b) = r - b + a$. *Fig. 1 depicts both spaces for* $r = 10$. *The Pareto-optimal set is* $\{(0, b) \mid b \in \{0, \ldots, r\}\}$ *and the Pareto-optimal front is* $\{(a, b) \mid a + b = r \;\wedge\; a, b \in \{0, \ldots, r\}\}$.

[1] A binary relation is called a *preorder* iff it is reflexive and transitive. A preorder which is antisymmetric is denoted as *partial order.*

[2] An element $\boldsymbol{z}^*$ of a subset Z' of a partially ordered set (Z, rel) is denoted as *minimal element* of Z' iff for all $\boldsymbol{z} \in Z'$: $\boldsymbol{z}\ rel\ \boldsymbol{z}^* \Rightarrow \boldsymbol{z} = \boldsymbol{z}^*$.

[3] A partial order *rel* is called a *total order* iff for all pairs $\boldsymbol{z}^1, \boldsymbol{z}^2$ either $\boldsymbol{z}^1\ rel\ \boldsymbol{z}^2$ or $\boldsymbol{z}^2\ rel\ \boldsymbol{z}^1$ (or both).

Table 1: Selected preference relations on objective vectors; the corresponding relations on decision vectors are defined on the basis of the associated objective vectors, i.e., $\boldsymbol{x}^1\ \mathit{rel}\ \boldsymbol{x}^2 \Leftrightarrow \boldsymbol{f}(\boldsymbol{x}^1)\ \mathit{rel}\ \boldsymbol{f}(\boldsymbol{x}^2)$. The relations $\succ$, $\succ\succ$, and $\succeq$ are used accordingly with reversed order of the arguments, e.g., $\boldsymbol{z}^1 \succ \boldsymbol{z}^2$ is equivalent to $\boldsymbol{z}^2 \prec \boldsymbol{z}^1$. Note that the 'indifference' relation actually only makes sense with regard to the decision space; in the objective space, it simply means equality.

relation	interpretation in objective space	
strictly dominates	$\boldsymbol{z}^1 \prec\prec \boldsymbol{z}^2$	$\boldsymbol{z}^1$ is better than $\boldsymbol{z}^2$ in all objectives
dominates	$\boldsymbol{z}^1 \prec \boldsymbol{z}^2$	$\boldsymbol{z}^1$ is not worse than $\boldsymbol{z}^2$ in all objectives and better in at least one objective
weakly dominates	$\boldsymbol{z}^1 \preceq \boldsymbol{z}^2$	$\boldsymbol{z}^1$ is not worse than $\boldsymbol{z}^2$ in all objectives
incomparable	$\boldsymbol{z}^1 \parallel \boldsymbol{z}^2$	neither $\boldsymbol{z}^1 \preceq \boldsymbol{z}^2$ nor $\boldsymbol{z}^2 \preceq \boldsymbol{z}^1$
indifferent	$\boldsymbol{z}^1 \sim \boldsymbol{z}^2$	$\boldsymbol{z}^1$ has the same value as $\boldsymbol{z}^2$ in each objective

2.1.2 Approximation Sets

The formal definition of an optimization problem given above assumes that only a single solution, any of those mapped to a minimal element, is sought. However, in a multiobjective setting one is often interested in the entire Pareto-optimal set rather than in a single, arbitrary Pareto-optimal solution. With many applications, e.g., engineering designs problems, knowledge about the Pareto-optimal set is helpful and provides valuable information about the underlying problem. This leads to a different optimization problem where the search space consists of sets of decision vectors and the objective space of sets of objective vectors. More specifically, the search focuses on sets of mutually incomparable solutions (for any two solutions $\boldsymbol{x}^1, \boldsymbol{x}^2$, neither weakly dominates the other one), which will be here denoted as *Pareto set approximations*; the symbol Ψ stands for the sets of all Pareto set approximations over X. Accordingly, sets of mutually incomparable objective vectors are here called *Pareto front approximations*, and the set of all Pareto front approximations over Z is represented by Ω.

Example 2 *For the problem in Example 1, the set* $\{(1, 10), (5, 5), (7, 2)\}$ *is a Pareto set approximation as none of its elements dominates another element. In contrast, the set* $\{(1, 10), (5, 5), (7, 3)\}$ *is not a Pareto set approximation since the decision vector* $(5, 5)$*, which is mapped to the objective vector* $(10, 10)$*, dominates the solution* $(7, 3)$*, which corresponds to the objective vector* $(10, 14)$.

Now, let $(X, Z, \boldsymbol{f}, \mathit{rel})$ be the original optimization problem. It can be canonically transformed into a corresponding set optimization problem $(\Psi, \Omega, \boldsymbol{f}', \mathit{rel}')$ by extending $\boldsymbol{f}$ and *rel* in the following manner:

- $\boldsymbol{f}'(E) = \{\boldsymbol{z} \in Z \mid \exists \boldsymbol{x} \in E : \boldsymbol{z} = \boldsymbol{f}(\boldsymbol{x})\}$
- $A\ \mathit{rel}' B \Leftrightarrow \forall \boldsymbol{z}^2 \in B\ \exists \boldsymbol{z}^1 \in A : \boldsymbol{z}^1\ \mathit{rel}\ \boldsymbol{z}^2$

If *rel* is $\preceq$, then *rel*$'$ represents the natural extension of weak Pareto dominance to Pareto front approximations. In the same manner, the preference relations on X and Z listed in Table 1 can be extended to Ψ resp. Ω as defined in Table 2; furthermore, an additional relation $\lhd$ is introduced that can be classified in between weak and regular Pareto dominance. In the following, we will use the same symbols as for preference relations on objective vectors and decision vectors also for Pareto front approximations respectively Pareto set approximations—it will become clear from the context, which relation is referred to.

In the remainder of this paper, we assume that $(\Psi, \Omega, \boldsymbol{f}', \preceq)$ is the underlying optimization scenario where $\preceq$ denotes weak Pareto dominance on Pareto front approximations. In this setting, which is commonly used in the evolutionary multiobjective optimization community, the Pareto-optimal front is the unique minimal element of $\boldsymbol{f}'(\Psi)$ with regard to the partially ordered set $(\Omega, \preceq)$. In practice, though, the generation of a set of decision vectors representing the entire Pareto-optimal front is often infeasible due to several reasons: for instance, the number of Pareto optima is too large, or even the determination of a single Pareto optimum is NP hard. Therefore, the aim is usually to identify a satisfactory Pareto set approximation, that means a set as 'close' as possible to the optimum in terms of the preorder defined by the weak Pareto dominance relation. All the following discussions are based on this scenario which means that *we consider only performance*

Table 2: Selected preference relations on Pareto front approximations; the corresponding relations on Pareto set approximations are defined by considering the associated Pareto front approximations. The relations $\succ$, $\succ\succ$, $\succeq$, and $\rhd$ are defined accordingly with reversed order of the arguments, e.g., $A \succ B$ is equivalent to $B \prec A$. Notice that (i) $A \prec\prec B \implies A \prec B \implies A \lhd B$ and (ii) two indifferent Pareto front approximations are identical, while this does not need to hold for two indifferent Pareto set approximations.

relation	interpretation in objective space	
strictly dominates	$A \prec\prec B$	every $\boldsymbol{z}^2 \in B$ is strictly dominated by at least one $\boldsymbol{z}^1 \in A$
dominates	$A \prec B$	every $\boldsymbol{z}^2 \in B$ is dominated by at least one $\boldsymbol{z}^1 \in A$
better	$A \lhd B$	every $\boldsymbol{z}^2 \in B$ is weakly dominated by at least one $\boldsymbol{z}^1 \in A$ and $A \not\sim B$
weakly dominates	$A \preceq B$	every $\boldsymbol{z}^2 \in B$ is weakly dominated by at least one $\boldsymbol{z}^1 \in A$
incomparable	$A \parallel B$	neither $A \preceq B$ nor $B \preceq A$
indifferent	$A \sim B$	$A \preceq B$ and $B \preceq A$

assessment methods that comply with the concept of Pareto dominance. There may be situations where the scenario just outlined does not appropriately reflect the intended optimization goal, e.g., when the decision maker is interested in further aspects such as robustness and diversity in the decision space. In this case, a different optimization problem emerges that may require specific algorithms and performance assessment methods. Such issues are not within the scope of the present paper.

Example 3 *Consider the situation depicted in Fig. 2 where the Pareto-optimal front consists of three elements. If not all decision criteria are quantifyable or included in the optimization model, the decision maker may prefer the approximation B over the Pareto-optimal set A because B contains more alternative solutions that are more diverse in the objective space, while being close to the optimal objective function values. Thereby, the decision maker has more freedom in choosing a final solution with respect to additional criteria.*

However, if such an scenario arises, the optimization problem is different and needs to be formalized appropriately. The search space may still be Ψ, but the corresponding partial order is not (extended) weak Pareto dominance anymore. In particular, the Pareto-optimal front is likely not to be a minimal element of (Ψ, rel), and in this case the Pareto-optimal set is not an optimal outcome—in other words, the goal of the optimization process is not to find a good approximation of the Pareto-optimal set. This has be taken into account in both algorithm design and performance assessment.

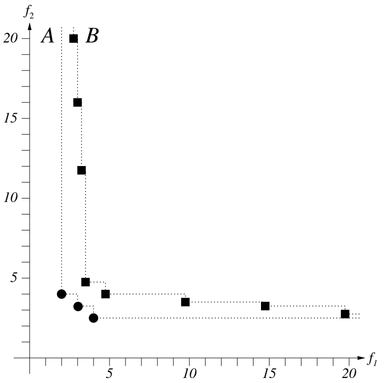


Figure 2: Hypothetical scenario where A represents the Pareto-optimal set and B stands for a Pareto set approximation, plotted in objective space.

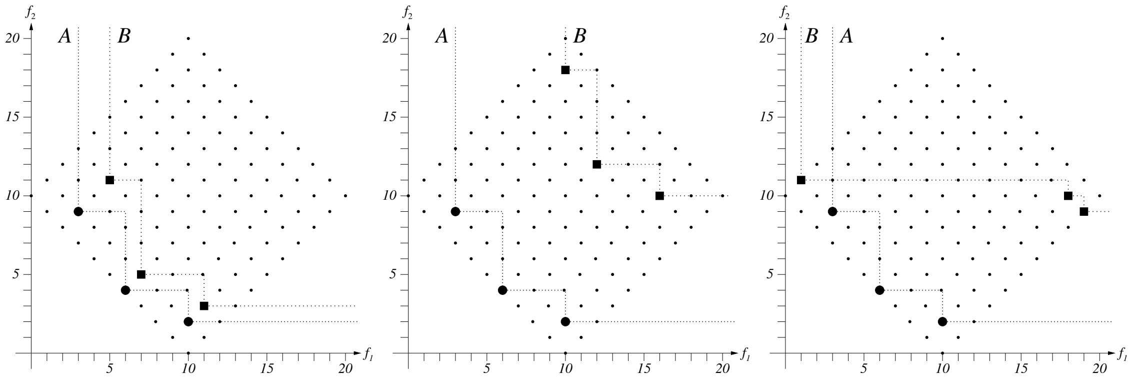


Figure 3: Three examples to illustrate the limitations of statements purely based on weak Pareto dominance. In both the figures on the left, the approximation set A dominates the approximation set B, but in one case the two sets are much closer together than in the other case. On the right, A and B are incomparable, but in most situations A will be more useful to the decision maker than B.

Moreover, we will make the following simplications. First, an algorithm designed for solving the optimization problem $(\Psi, \Omega, \boldsymbol{f}', \preceq)$ is denoted as a *multiobjective optimizer*. Although multiobjective optimizers generate sets of decision vectors, in the remainder of the paper only Pareto front approximations will be considered, similarly to (Hansen and Jaszkiewicz 1998; Zitzler et al. 2003). The reason is that quality assessment is usually done in objective space; the issue of assessing the outcomes of multiobjective optimizers with regard to the decision space will not be addressed. Second, the term *approximation set* will be used as an alias for Pareto front approximation from now on.

2.2 Outperformance

Suppose we would like to assess the performance of two multiobjective optimizers. The question of whether either outperforms the other one involves various aspects such as the quality of the outcome, the computation time required, the parameter settings, etc. This paper focuses on the quality aspect and addresses the issue of how to compare several approximation sets. For the time being, assume that we consider one optimization problem only and that the two algorithms to be compared are deterministic, i.e., with each optimizer exactly one approximation set is associated; the issue of stochasticity will be treated in the next section.

As discussed above, optimization is about searching in an ordered set. The partial order *rel* for an optimization problem $(X, Y, \boldsymbol{f}, rel)$ defines a preference structure on the search space: a solution $\boldsymbol{x}^1$ is preferable to a solution $\boldsymbol{x}^2$ iff $\boldsymbol{f}(\boldsymbol{x}^1)$ *rel* $\boldsymbol{f}(\boldsymbol{x}^2)$ and not $\boldsymbol{f}(\boldsymbol{x}^2)$ *rel* $\boldsymbol{f}(\boldsymbol{x}^1)$. This preference structure is the basis on which the optimization process is performed. Therefore, the most natural way to compare two approximation sets A and B generated by two different multiobjective optimizers is to use the underlying preference structure. In the context of weak Pareto dominance, there can be four situations: (i) A is better than B, (ii) B is better than A, (iii) A and B are incomparable, or (iv) A and B are indifferent, cf. Table 2. These are the types of statements one can make without any additional preference information. Often, though, we are interested in more precise statements that quantify the difference in quality on a continuous scale. For instance, in cases (i) and (ii) we may be interested in knowing how much better the preferable approximation set is, and in case (iii) one may ask whether either set is better than the other in certain aspects not captured by the preference structure—this is illustrated in Fig. 3.

For this purpose, quantitative performance measures have been introduced. The term performance measure is a little bit unfortunate, as performance usually refers to both quality and time, while the measures proposed in the literature usually capture only the former aspect. Therefore, we will use the term *quality indicator* instead:[4]

[4] Another term which is often used in this context is performance metric or quality metric; however, metric has a clearly defined mathematical meaning that does not apply to most measures proposed in the literature.

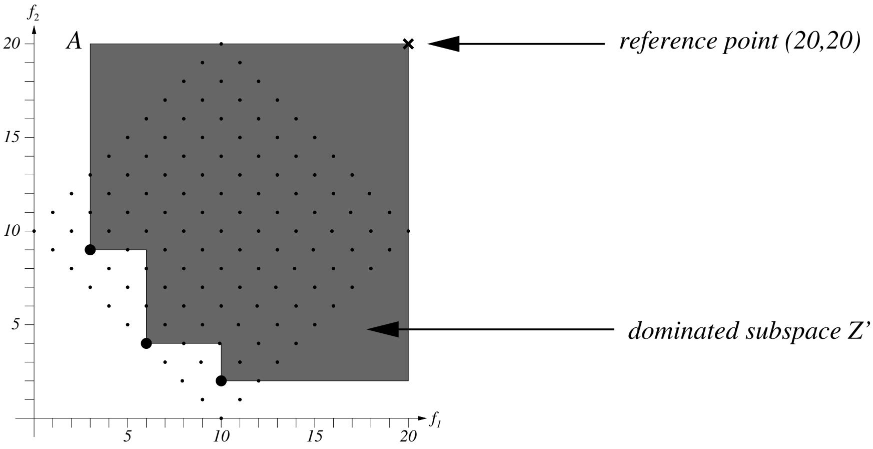


Figure 4: Illustration of the hypervolume indicator. In this example, approximation set A is assigned the indicator value $I_H(A) = 277$; the objective vector $(20, 20)$ is taken as the reference point.

> A (unary) quality indicator is a function $I : \Omega \to \mathbb{R}$ that assigns each approximation set a real number.

In combination with the $\leq$ or $\geq$ relation on $\mathbb{R}$, a quality indicator I defines a total order of Ω and thereby induces a corresponding preference structure: A is preferable to B iff $I(A) > I(B)$, assuming that the indicator values are to be maximized. That means we can compare the outcomes of two multiobjective optimizers by comparing the corresponding indicator values.

Example 4 *Let A be an arbitrary approximation set and consider the subspace $\boldsymbol{Z}'$ of the objective space $\boldsymbol{Z} = \mathbb{R}^n$ that is, roughly speaking, weakly dominated by A. That means any objective vector in $\boldsymbol{Z}'$ is weakly dominated by at least one objective vector in A.*

The hypervolume indicator I_H (Zitzler and Thiele 1999) gives the hypervolume of $\boldsymbol{Z}'$ (see Fig. 4). The greater the indicator value, the better the approximation set. Note that this indicator requires a reference point relatively to which the hypervolume is calculated. This issue will be discussed in detail in Section 3.2.

Considering again Fig. 3, it can be seen that the hypervolume indicator reveals differences in quality that cannot be detected by the dominance relations from Table 2. In the left scenario, $I_H(A) = 277$ and $I(B) = 231$, while for the scenario in the middle, $I_H(A) = 277$ and $I(B) = 76$; in the right scenario, the indicator values are $I_H(A) = 277$ and $I_H(B) = 174$.[5] This advantage, though, comes at the expense of generality, since every quality indicator represents certain assumptions about the decision maker's preferences. Whenever $I_H(A) > I_H(B)$, we can state that A is better than B with respect to the hypervolume indicator; however, the situation could be different for another quality indicator I' that assigns B a better indicator value than A. As a consequence, every comparison of multiobjective optimizers is not only restricted to the selected benchmark problems and parameter settings, but also to the quality indicator(s) under consideration. For instance, if we use the hypervolume indicator in a comparative study, any statement like "optimizer 1 outperforms optimizer 2 in terms of quality of the generated approximation set" needs to be qualified by adding "under the assumption that I_H reflects the decision maker's preferences".

Given the fact that quality indicators express the preferences of the decision maker, one might imagine that *any* function $I : \Omega \to \mathbb{R}$ could be chosen as an indicator. However, this is not so if we wish to maintain consistency with the inherent preference structure of the optimization problem under consideration. In that case, the total order of Ω imposed by the choice of I should not contradict the partial order of Ω that is imposed by the weak Pareto dominance relation. That is, whenever an approximation set A is preferable to B with respect to weak Pareto dominance, the indicator value for A should be at least as good as the indicator value for B; we will call such indicators *Pareto compliant.*

[5] The objective vector $(20, 20)$ is the reference point.

> An indicator $I : \Omega \rightarrow \mathbb{R}$ is *Pareto compliant* iff for all $A, B \in \Omega$: $A \preceq B \Rightarrow I(A) \geq I(B)$, assuming that greater indicator values correspond to higher quality (otherwise $A \preceq B \Rightarrow I(A) \leq I(B)$). In the context of order theory, a Pareto compliant indicator I is an *order-preserving* function from $(\Omega, \preceq)$ to $(\mathbb{R}, \geq)$ (respectively $(\mathbb{R}, \leq)$).

Pareto compliant indicators define refinements of the partial order induced by weak Pareto dominance, and in the remainder of this review we shall limit our discussion of quality indicators to this class of functions.

Note that many of the indicators that have been proposed and are frequently used in the literature are not Pareto compliant. Several popular indicators are designed to assess just one isolated aspect of an approximation set's quality, such as its proximity to the Pareto-optimal front, or its spread in objective space, or the evenness with which the points in it are distributed. These quality indicators, sometimes referred to as 'functionally-independent' indicators tend to be—what we here call—*Pareto non-compliant.*

> Any indicator that can yield for any approximation sets $A, B \in \Omega$ a preference for A over B, when B is preferable to A with respect to weak Pareto dominance ($B \preceq A \wedge \neg\ A \preceq B$, or $B \lhd A$ for short), is *Pareto non-compliant.*

A number of these indicators are classified in Figure 16 in Appendix A. We supplement this with an empirical study in Appendix B, showing that these indicators violate the partial order of weak Pareto dominance quite frequently, not only in some pathological cases. This does not mean that Pareto non-compliant indicators are useless per se for optimization scenarios based on (weak) Pareto dominane; for instance, they may be used to refine the preference structure of a Pareto compliant indicator for approximation sets having identical indicator values. Furthermore, there may other optimization scenarios (Ψ, Ω, h, rel) that are not based on weak Pareto dominance and for which such an indicator is appropriate—provided it does not contradict the partial order defined by *rel*.

Finally, note that the above discussion was restricted to unary quality indicators only, although an indicator can take an arbitrary number of approximation sets as arguments. Several quality indicators have been proposed that assign real numbers to pairs of approximation sets (see (Zitzler et al. 2003) for an overview). For instance, the unary hypervolume indicator can be extended to a binary quality indicator by defining $I_H(A, B)$ as the hypervolume of the subspace of the objective space that is dominated by A but not by B.

2.3 Stochasticity

So far, we have assumed that each algorithm under consideration always generates the same approximation set for a specific problem. However, many multiobjective optimizers are variants of randomized search algorithms and therefore stochastic in nature. If a stochastic multiobjective optimizer is applied several times to the same problem, each time a different approximation set may be returned. In this sense, with each randomized algorithm a random variable is associated whose possible values are approximation sets, i.e., elements of Ω; the underlying probability density function is usually unknown.

One way to estimate this probability density function is by means of theoretical analysis. Since this approach is infeasible for many problems and algorithms used in practice, empirical studies are common in the context of the performance assessment of multiobjective optimizers. By running a specific algorithm several times on the same problem instance, one obtains a sample of approximation sets. Now, comparing two stochastic optimizers basically means comparing the two corresponding approximation set samples. This leads to the issue of statistical hypothesis testing. While in the deterministic case one can state, e.g., that "optimizer 1 achieves a higher hypervolume indicator value than optimizer 2", a corresponding statement in the stochastic case could be that "the expected hypervolume indicator value for algorithm 1 is greater than the expected hypervolume indicator value for algorithm 2 at a significance level of 5%".

In principle, there exist two basic approaches in the literature to analyze two or several samples of Pareto set approximations statistically. The more popular approach first transforms the approximation set samples into samples of real values using quality indicators; then, the resulting samples of indicator values are compared based on standard statistical testing procedures.

Example 5 *Consider two hypothetical stochastic multiobjective optimizers that are applied to the problem in Example 1, and assume that the outcomes of three independent optimization runs are as depicted in Fig. 5.*

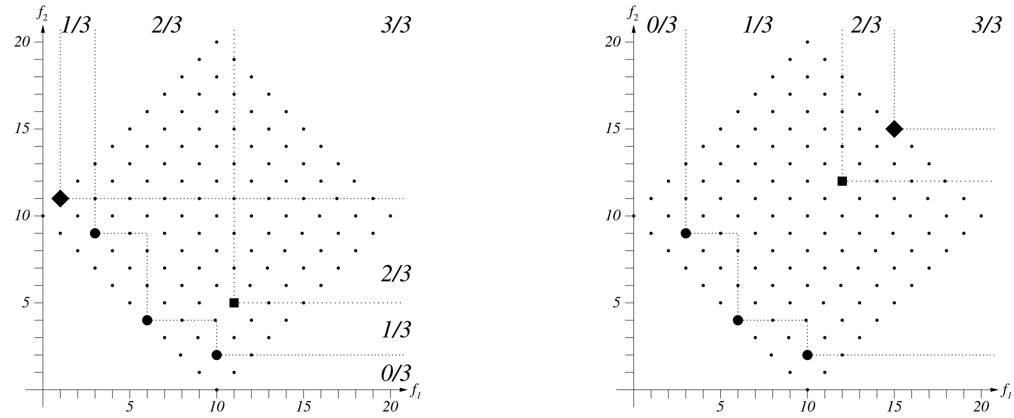


Figure 5: Hypothetical outcomes of three runs for two different stochastic optimizers (left and right). The numbers in the figures give the relative frequencies according to which the distinct regions in the objective space were attained.

If we use the hypervolume indicator with the reference point $(20, 20)$*, we obtain two samples of indicator values:* $(277, 171, 135)$ *and* $(277, 64, 25)$*. These indicator value samples can then be compared and differences can be subjected to statistical testing procedures.*

The alternative approach, the attainment function method, summarizes a sample of approximation sets in terms of a so-called empirical attainment function. To explain the underlying idea, suppose that a certain stochastic multiobjective optimizer is run once on a specific problem. For each objective vector $\boldsymbol{z}$ in the objective space, there is a certain probability p that the resulting approximation set contains an objective vector that weakly dominates $\boldsymbol{z}$. We say p is the probability that $\boldsymbol{z}$ is *attained* by the optimizer. The *attainment function* gives for each objective vector $\boldsymbol{z}$ in the objective space the probability that $\boldsymbol{z}$ is attained in one optimization run of the considered algorithm. As before, the true attainment function is usually unknown, but it can be estimated on the basis of the approximation set samples: one simply counts the number of approximation sets by which each objective vector is attained and normalizes the resulting number with the overall sample size. For the The attainment function is a first order moment measure, meaning that it estimates the probability that $\boldsymbol{z}$ is attained in one optimization run of the considered algorithm *independently* of attaining any other $\boldsymbol{z}$. For the consideration of higher order attainment functions Grunert da Fonseca, Fonseca, and Hall (2001) have developed corresponding statistical testing procedures.

Example 6 *Consider Fig. 5. For the scenario on the right, the three approximation sets cut the objective space into four regions: the upper right region is attained in all of the runs and therefore is assigned a relative frequency of* 1*, the lower left region is attained in none of the runs, and the remaining two regions are assigned relative frequencies of* $1/3$ *and* $2/3$ *because they are attained in one respectively two of the three runs. In the scenario on the left, the objective space is partitioned into six regions; the relative frequencies are determined analogously as shown in the figure.*

A third approach proposed in this paper consists in ranking the obtained approximations by means of the dominance relation, in analogous fashion to the way dominance-based fitness assignment ranks objective vectors in evolutionary multiobjective optimization. First, all approximation sets generated by the different optimizers under consideration are pooled, and then each approximation set is assigned a rank reflecting the number of approximation sets in the pool that are better (cf. Table 2, $A \lhd B$). Thereby, one obtains, for each algorithm, a set of ranks and can statistically verify whether the rank distributions for two algorithms differ significantly or not.

Example 7 *To compare the outcomes of the two hypothetical optimizers depicted in Fig. 5, we check for each pair of the overall six approximation sets whether one is better or not. For the approximation set represented by the diamond on the left hand side, none of the other five approximation sets is better and therefore it is* *assigned the lowest rank* 1. *The approximation set associated with the diamond on the right hand side is worse than all other five approximation sets and accordingly its rank is* $1 + 5$. *Overall, the resulting rank distributions are* $(1, 1, 3)$ *for the algorithm on the left hand side and* $(1, 5, 6)$ *for the algorithm on the right hand side. A statistical test can be used to determine whether the two rank distributions are significantly different.*

3 Sample Transformations

The three comparison methodologies outlined in the previous section have in common that the sample of approximation sets associated with an algorithm is first transformed into another representation—namely a sample of indicator values, an empirical attainment function, or a sample of ranks—before the statistical testing methods are applied. In the following, we will review each of the different types of sample transformations in greater detail (but now considering the dominance ranking first); the issue of statistical testing will be covered in Section 4.

3.1 Dominance Ranking

3.1.1 Principles and Procedure

Suppose that we wish to compare the quality of approximation sets generated by $q \geq 2$ stochastic multiobjective optimizers. For each optimizer $i \in \{1, \ldots, q\}$, a number of runs $r_i \geq 1$ are performed, generating approximations sets $A_1^1, A_2^1, \ldots, A_{r_1}^1, \ldots, A_1^q, \ldots, A_{r_q}^q$.

If one considers the combined collection $\mathbf{C}$ of all the approximation sets, then typically some of the sets will dominate or be better than some of the other ones, while some other pairs of sets will be incomparable to each other. Hence, the relations listed in Table 2 can be used to define a partial order among these approximation sets, and this partial order, in turn, can be exploited to assign a figure of merit or a rank to each approximation set, similarly to the way that dominance-based fitness assignment works on objective vectors in multiobjective evolutionary algorithms.

In principle, there are several ways to assign each approximation set a rank on the basis of a dominance relation, e.g., by counting the number of sets by which a specific approximation set is dominated (Fonseca and Fleming 1993) or by performing a nondominated sorting on $\mathbf{C}$ (Goldberg 1989, p. 201). Here, the former approach in combination with the 'better' relation, cf. Table 2, is preferred as it produces a finer-grained ranking, with fewer ties, than nondominated sorting:

$$rank(C_i) = 1 + |\{C_j \in \mathbf{C} \; : \; C_j \lhd C_i\}|. \tag{1}$$

The lower the rank, the better the corresponding approximation set with respect to the entire collection.

The result of this procedure is that each $C_i \in \mathbf{C}$ of the approximation sets is associated with a figure of merit. Accordingly, the approximation set samples associated with each algorithm have been transformed into samples:

$$\Big(rank(A_1^1), rank(A_2^1), \ldots, rank(A_{r_1}^1)\Big), \ldots, \Big(rank(A_1^q), rank(A_2^q), \ldots, rank(A_{r_q}^q)\Big).$$

A statistical rank test (Conover 1999), cf. Section 4, can then be used to determine whether there is a significant difference in the distribution of these values, in particular whether the ranks for one algorithm are significantly smaller than the ranks assigned to another algorithm.

3.1.2 Discussion

The dominance ranking approach relies on the concept of Pareto dominance and some ranking procedure only, and thus yields quite general statements about the relative performance of the considered optimizers, fairly independently of any preference information. Thus, we recommend this approach to be the first step in any comparison: if one optimizer is found to be significantly better than the other by this procedure, then it is better in a sense consistent with the preference ordering of approximation sets defined in Section 2.1.2. It may be interesting and worthwhile to use either quality indicators or the attainment function to characterize further the differences in the approximation set distributions, but these methods are not needed to conclude which of the stochastic optimizers generates the better sets, if a significant difference can be demonstrated using the ranking of approximation sets alone.

3.2 Quality Indicators

3.2.1 Principles and Procedures

As stated earlier, a *unary* quality indicator I is defined as a mapping from the set of all approximation sets Ω to the set of real numbers:[6]

$$I : \Omega \mapsto \mathbb{R}.$$

The order that I establishes on Ω is supposed to represent the quality of the approximation sets. Thus, given a pair of approximation sets, A and B, the difference between their corresponding indicator values $I(A)$ and $I(B)$ should reveal a difference in the quality of the two sets. This not only holds for the case that either set is better, but also when A and B are incomparable. Note that this type of information goes beyond pure Pareto dominance and represents additonal knowledge; we denote this knowledge as *preference information.*

In the following, three (families of) recommended unary quality indicators are presented, each of which measures 'total information' in a slightly different way. The idea here is to measure the total amount of the objective space that has been 'covered' by the approximation set, or to put it another way, the number of goals that have been attained (as opposed to separating out different aspects such as spread, evenness or proximity). In this sense, the approach is closely related to the attainment function approach detailed in Section 3.3.

The Hypervolume Indicator I_H This indicator, which has been proposed by Zitzler and Thiele (1999), measures the hypervolume of that portion of the objective space that is weakly dominated by an approximation set A, and is to be maximized (see Fig. 6). In order to measure this quantity, the objective space must be bounded—if it is not, then a bounding reference point that is (at least weakly) dominated by all points should be used, as shown in the figure.

Note that one can also consider the hypervolume difference to a reference set R, and we will refer to this indicator as I_H^-. Given an approximation set A, the indicator value is defined as

$$I_H^-(A) = I_H(R) - I_H(A) \tag{2}$$

where smaller values correspond to higher quality—in contrast to the original hypervolume I_H.

The I_H indicator has a desirable property: Whenever $A \lhd B$, then $I_H(A) > I_H(B)$—provided that the bounding point is strictly dominated by all the points in A and B.[7] Therefore, from $I_H(A) < I_H(B)$ one can infer that A cannot be better than B. To our best knowledge, the hypervolume indicator is the only unary indicator that is capable of detecting that A is not better than B for all pairs $B \lhd A$.

Computation of the hypervolume indicator has recently been shown to be exponential in the number of objectives (While 2005; While, Bradstreet, Barone, and Hingston 2005) and is polynomial in the number of points in the approximation set.

The Unary Epsilon Indicators I_ϵ^1 and $I_{\epsilon+}^1$ The epsilon indicator family has been introduced in (Zitzler et al. 2003) and comprises a multiplicative and an additive version—both exist in unary and binary form. The binary multiplicative epsilon indicator, $I_\epsilon(A, B)$, gives the minimum factor ϵ by which each point in B can be multiplied such that the resulting transformed approximation set is weakly dominated by A:

$$I_\epsilon(A, B) = \inf_{\epsilon \in \mathbb{R}} \{\forall \boldsymbol{z}^2 \in B \; \exists \boldsymbol{z}^1 \in A \; : \; \boldsymbol{z}^1 \preceq_\epsilon \boldsymbol{z}^2\}. \tag{3}$$

[6] Indicators that map an ordered pair of approximation sets to a real number, $I : \Omega \times \Omega \mapsto \mathbb{R}$ are called binary indicators (see (Knowles and Corne 2002; Zitzler et al. 2003) for reviews). As demonstrated in (Zitzler et al. 2003), binary indicators are more powerful than unary indicators as, e.g., the binary indicator I_ϵ is able to detect dominance or incomparability between two approximation sets. However, it is beyond the scope of this review to discuss how to make statistical inferences on the basis of binary quality indicators.

[7] When the reference point is only weakly dominated by one or more points, then $A \lhd B$ implies that $I(A) \geq I(B)$.

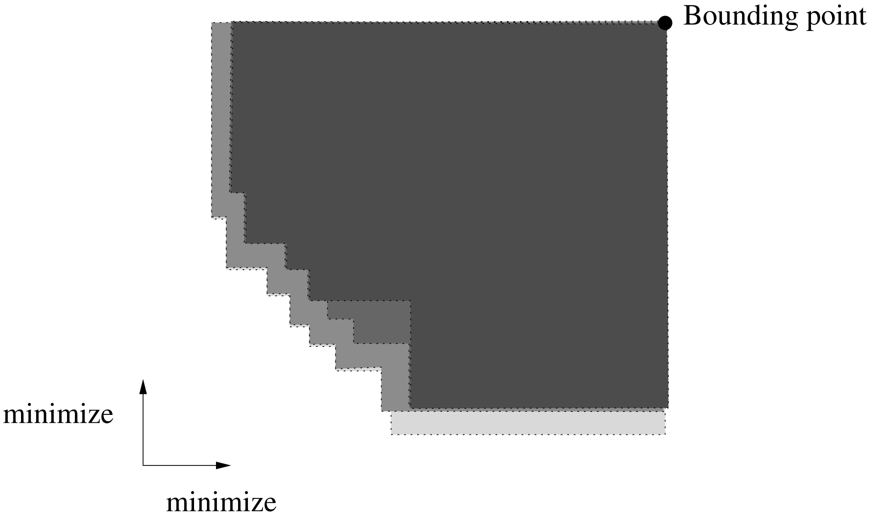


Figure 6: The hypervolume indicator measures the size of the dominated region, bounded by some reference point. Here, four different sets A, B, C, D are shown by increasing order of darkness of the shaded region, and $A \lhd B \lhd C \lhd D$. A is different to B mainly in extent, B is better than C in proximity to the Pareto front, and C is better than D mainly in evenness. Thus, the hypervolume indicator is capable of detecting differences in any of these different aspects.

This indicator relies on the ϵ-dominance relation, $\preceq_\epsilon$, defined as:

$$\boldsymbol{z}^1 \preceq_\epsilon \boldsymbol{z}^2 \iff \forall i \in 1..n \, : \, z_i^1 \leq \epsilon \cdot z_i^2 \tag{4}$$

for a minimization problem, and assuming that all points are positive in all objectives. On this basis, the unary multiplicative epsilon indicator, $I_\epsilon^1(A)$ can then be defined as:

$$I_\epsilon^1(A) = I_\epsilon(A, R), \tag{5}$$

where R is any reference set of points. An equivalent unary additive epsilon indicator $I_{\epsilon+}^1$ is defined analogously, but is based on additive ϵ-dominance:

$$\boldsymbol{z}^1 \preceq_{\epsilon+} \boldsymbol{z}^2 \iff \forall i \in 1..n \, : \, z_i^1 \leq \epsilon + z_i^2. \tag{6}$$

Both unary indicators are to be minimized. An indicator value smaller than 1 (I_ϵ^1) respectively 0 ($I_{\epsilon+}^1$) implies that A strictly dominates the reference set R.

For the unary epsilon indicators, it holds that whenever $A \lhd B$, then $I_\epsilon^1(A) \leq I_\epsilon^1(B)$ respectively $I_{\epsilon+}^1(A) \leq I_{\epsilon+}^1(B)$. Accordingly, from $I_\epsilon^1(A) > I_\epsilon^1(B)$ respectively $I_{\epsilon+}^1(A) > I_{\epsilon+}^1(B)$, one can deduce that A is not better than B. However, the unary epsilon indicators work on a different principle than the hypervolume indicator. Therefore, in some cases, the hypervolume and epsilon indicators may return opposite preference orderings for a pair of approximation sets A and B. From such a result, it logically follows that the two sets A and B are incomparable. An example of such a case is given in Figure 7.

For any finite approximation set A, and any finite reference set R, the unary epsilon indicator is cheap to compute; the runtime complexity is of order $\mathcal{O}(n \cdot |A| \cdot |R|)$, where n is the number of objectives.

The I_{R2}^1 and I_{R3}^1 Indicators The R indicators proposed in (Hansen and Jaszkiewicz 1998) can be used to assess and compare approximation sets on the basis of a set of utility functions. Here, a utility function u is defined as a mapping from the set Z of n-dimensional objective vectors to the set of real numbers:

$$u : Z \mapsto \mathbb{R}.$$

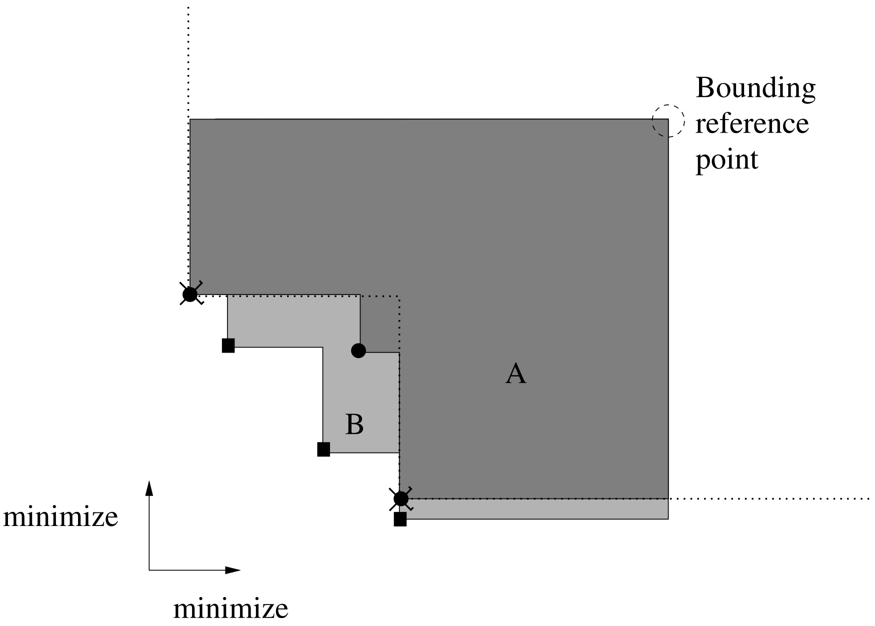


Figure 7: Two imcomparable approximation sets A and B. Under the hypervolume indicator, B is the better set, but under the epsilon indicator A is better with respect to the reference set (the two X points connected by the dashed line), since $I_{\epsilon 1}(A)$ is one, while $I_{\epsilon 1}(B)$ is greater than one. This discrepancy indicates that the two sets must be incomparable.

In this sense, it represents the counterpart to a quality indicator with the difference that the domain is Z and not Ω.

Suppose that the decision maker's preferences are given in terms of a parameterized utility function u_{lambda} and a corresponding set Λ of parameters. For instance, $u_{\boldsymbol{\lambda}}$ could represent a weighted sum of the objective values, where $\boldsymbol{\lambda} = (\lambda_1, \dots \lambda_n) \in \Lambda$ stands for a particular weight vector. Hansen and Jaszkiewicz (1998) propose several ways to transform such a family of utility functions into a quality indicator; in particular, the binary I_{R2} and I_{R3} indicators are defined as:[8]

$$I_{R2}(A, B) = \frac{\sum_{\boldsymbol{\lambda} \in \Lambda} u^*(\boldsymbol{\lambda}, A) - u^*(\boldsymbol{\lambda}, B)}{|\Lambda|}, \tag{7}$$

$$I_{R3}(A, B) = \frac{\sum_{\boldsymbol{\lambda} \in \Lambda} [u^*(\boldsymbol{\lambda}, B) - u^*(\boldsymbol{\lambda}, A)] / u^*(\boldsymbol{\lambda}, B)}{|\Lambda|}. \tag{8}$$

where u^* is the maximum value reached by the utility function $u_{\boldsymbol{\lambda}}$ with weight vector $\boldsymbol{\lambda}$ on an approximation set A, i.e., $u^*(\boldsymbol{\lambda}, A) = \max_{z \in A} u_{\boldsymbol{\lambda}}(z)$. Similarly to the epsilon indicators, the unary R indicators are defined on the basis of the binary versions by replacing B by an arbitrary, but fixed reference set: $I^1_{R2}(A) = I_{R2}(R, A)$ and $I^1_{R3}(A) = I_{R3}(A, R)$.

With respect to the choice of the parameterized utility function $u_{\boldsymbol{\lambda}}$, there are various possibilities. A first utility function u that can be used in the above is a weighted linear function

$$u_{\lambda}(z) = -\sum_{j \in 1..n} \lambda_j |z^*_j - z_j|, \tag{9}$$

where $\boldsymbol{z}^*$ is the ideal point, if known, or any point that weakly dominates all points in the approximation set. (When comparing approximation sets, the same $\boldsymbol{z}^*$ must be used each time).

[8] The full formalism described in (Hansen and Jaszkiewicz 1998) also considers arbitrary sets of utility functions in combination with a corresponding probability distribution over the utility functions. The interested reader is referred to the original paper for further information.

A disadvantage of the use of a weighted linear function means that points not on the convex hull of the approximation set are not rewarded. Therefore, it is often preferable to use a nonlinear function such as the weighted Tchebycheff function,

$$u_{\boldsymbol{\lambda}}(\boldsymbol{z}) = -\max_{j\in 1..n} \lambda_j |z_j^* - z_j|. \tag{10}$$

In this case, however, the utility of a point and one which weakly dominates it might be the same. To avoid this, it is possible to use the combination of linear and nonlinear functions: the augmented Tchebycheff function,

$$u_{\boldsymbol{\lambda}}(\boldsymbol{z}) = -\left(\max_{j\in 1..n} \lambda_j |z_j^* - z_j| + \rho \sum_{j\in 1..n} |z_j^* - z_j|\right), \tag{11}$$

where ρ is a sufficiently small positive real number. In all cases, the set Λ of weight vectors should contain a sufficiently large number of uniformly dispersed normalized weight combinations $\boldsymbol{\lambda}$ with $\forall i \in 1..n : \lambda_n \geq 0 \wedge \sum_{j=1..n} \lambda_j = 1$.

Provided that any of the above utility functions is used, the R indicators guarantee that, in the case of minimization, the indicator value for an approximation set A is less than or equal to the indicator value associated with B, whenever $A \lhd B$. Vice versa, from $I^1_{R2}(A) > I^1_{R2}(B)$ respectively $I^1_{R3}(A) > I^1_{R3}(B)$ one can infer that A is not better than B.

The runtime complexity for computing the indicator values of order $\mathcal{O}(n \cdot |\Lambda| \cdot |A| \cdot |R|)$, where n is the number of objectives.

3.2.2 Discussion

Using unary quality indicators in a comparative study is attractive as it transforms a sample of approximation sets into a sample of reals for which standard statistical testing procedures exist, cf. Section 4. In contrast to the dominance ranking approach, it is also possible to make quantitative statements about the differences in quality, even for incomparable approximation sets. However, this comes at the cost of generality: every unary quality indicator represents specific preference information. Accordingly, any statement of the type 'algorithm A outperforms algorithm B' needs to be qualified in the sense of 'with respect to quality indicator I'—the situation may be different for another indicator.

In the following, we discuss important aspects that need to be considered when quality indicators are to be used for sample transformation.

Combination of Quality Indicators Many comparative studies in the literature evaluate algorithms with respect to several different quality indicators. This is a sound approach which provides quality assessments with respect to slightly different decision-maker preferences—but only provided the considered indicators are Pareto compliant (cf. Section 2). A combination of Pareto compliant indicators can also yield interpretations that are more powerful than can be made by a single indicator alone. In particular, if two Pareto compliant indicators contradict one another on the preference ordering of two approximations sets, then this implies that the two sets are incomparable.

The situation is different, though, if the indicators used are Pareto non-compliant. Many of the *Pareto non-compliant* indicators detailed in Fig. 16 in Appendix A are frequently used in combination with each other. A common approach is to assess isolated aspects of a decision maker's preferences with respect to approximation sets, e.g., their proximity to the Pareto front, diversity, evenness, and cardinality, in terms of distinct quality indicators. However, it is possible that all of the indicators judge that A is preferable to B, when in fact B is better than A according to Pareto dominance; this is demonstrated empirically in Appendix B. Thus, the combination of several indicators does not nullify or minimize the impact of their Pareto non-compliance; on the contrary, it can give an unjustified sense of 'security' to the interpretations made.

Scaling and Normalization In principle, the notion of Pareto dominance is completely scale- and normalization-independent. When using quality indicators, though, scaling and normalization can be necessary in order to allow the different objectives to contribute approximately equally to indicator values, for indicators such as I_H, and $I^1_{\epsilon+}$.

A standard, linear normalization procedure will apply the following transformation to each objective dimension's value:

$$z_i' = \frac{z_i - z_i^{(min)}}{z_i^{(max)} - z_i^{(min)}} \tag{12}$$

where $z_i^{(max)}$ and $z_i^{(min)}$ are some known or estimated maximum and minimum values, respectively that the ith objective can take. In the case of indicators, such as I_ϵ^1, which rely on all objective values being strictly positive, $z_i^{(min)}$ should be an unattainable value; alternatively, we can add 1 to the transformed values z_i' such that all objective values lie in the interval $[1, 2]$.

Note also that sometimes it may be appropriate to use the original objective function values. For example, in the multiobjective knapsack problems, each objective is measuring the same kind of thing using the same scale (benefit or profit). Therefore, in this case, it may be best to simply use absolute objective values.

Reference Points and Sets The reference point $\boldsymbol{z}^+$ bounding the dominated region in the hypervolume indicator, cf. Fig. 6, must be set in such a way that the objective vectors contained in the approximation sets $A_1, A_2, \ldots, A_r$ under consideration are dimension-wise smaller than the reference point:

$$\forall i = 1 \ldots r \;\forall \boldsymbol{z} \in A_i \;\forall j = 1 \ldots n \,:\, z_j < z_j^+ \tag{13}$$

If the image $f(X)$ of the search space X is bounded, then $\boldsymbol{z}^+$ can simply be set to any objective vector that is worse in all dimensions than any element in $f(X) \subset Z$.

Similar comments apply to the choice of reference point $\boldsymbol{z}^*$ for the I_{R2}^1 and I_{R3}^1 indicators, except that here the reference point should weakly dominate any objective vector in the approximation sets:

$$\forall i = 1 \ldots r \;\forall \boldsymbol{z} \in A_i \;\forall j = 1 \ldots n \,:\, z_j^* \leq z_j. \tag{14}$$

As to the reference approximation set in the context of the epsilon and R indicators, the Pareto front is the ideal reference because it can yield more power to some indicators, e.g., the I_ϵ^1 indicator. However, in most cases the Pareto front is unknown and cannot be computed in reasonable time. We recommend two alternative approaches to circumvent this problem:

- First, all approximation sets generated by the algorithms under consideration are combined, and then the dominated objective vectors are removed from this union. The remaining points, which are not dominated by any of the approximation sets, form the reference set.
- Another possibility is to use a reference set that dominates 50% of solutions in the search space — a kind of median reference set. To this end, a certain number, e.g., 1000, points are randomly created, each one representing the outcome of one run of a random search strategy, and then the 50% attainment surface of these 1000 artificial runs is taken as the reference set. More details on attainment surfaces are given in Section 3.3.

The advantage of the first approach is that the reference set weakly dominates all approximation sets under consideration; however, whenever additional approximation sets are included in the comparison, the reference set needs to be re-computed. The second approach avoids this problem, but also gives a slightly different picture: it measures the quality of an approximation set with respect to a reference set independent of any algorithm; it is also 'nearly' independent of how many points are sampled, so that 1000 should prove enough for its location to have converged.

When communicating the results of a study, it is strongly advised to communicate the reference point and reference sets used, in addition to the indicator values. This means that others can compare their indicator values, computed using the same reference point and reference set, directly with the ones reported in the study, without access to the approximation sets.

3.3 Empirical Attainment Function

3.3.1 Principles and Procedures

The central concept in this approach is the notion of an *attainment function*. Since the multiobjective optimizers that we consider are stochastic, the result of running the optimizer can be described by a distribution.

Because the optimizer may return more than one objective vector in any given run, the distribution is described by a random *set* $\mathcal{Z}$ of random objective vectors $\check{\boldsymbol{z}}^j$, with the cardinality of the set, $\check{m}$, also random, as follows:

$$\mathcal{Z} = \{\check{\boldsymbol{z}}^j \in \mathbb{R}^n, j = 1, \ldots, \check{m}\}, \tag{15}$$

where n is the number of objectives of the problem, as usual. The attainment function is a description of this distribution based on the notion of goal-attainment: A goal, here meaning an objective vector, is attained whenever it is weakly dominated by the approximation set returned by the optimizer. It is defined by the function $\alpha_{\mathcal{Z}}(.) : \mathbb{R}^n \mapsto [0, 1]$ with

$$\begin{aligned} \alpha_{\mathcal{Z}}(z) &= P(\check{\boldsymbol{z}}^1 \preceq \boldsymbol{z} \vee \check{\boldsymbol{z}}^2 \preceq \boldsymbol{z} \vee \ldots \vee \check{\boldsymbol{z}}^{\check{m}} \preceq \boldsymbol{z}) && (16)\\ &= P(\mathcal{Z} \preceq \{\boldsymbol{z}\}) && (17)\\ &= P(\text{that the optimizer attains goal } \boldsymbol{z} \text{ in a single run}), && (18) \end{aligned}$$

where $P(.)$ is the probability density function. The attainment function is a first order moment measure, and can be seen as a mean-measure for the set $\mathcal{Z}$. Thus, it describes the location of the approximation set distribution; higher order moments are needed if the variability across runs is to be assessed, and to assess dependencies between the probabilities of attaining two or more goals *in the same run* (see (Fonseca et al. 2005)).

The attainment function can be estimated from a sample of r independent runs of an optimizer via the *empirical attainment function* (EAF) defined as

$$\alpha_r(\boldsymbol{z}) = \frac{1}{r} \sum_{i=1}^{r} \boldsymbol{I}(A^i \preceq \{\boldsymbol{z}\}), \tag{19}$$

where A^i is the ith approximation set (run) of the optimizer and $\boldsymbol{I}(.)$ is the indicator function, which evaluates to one if its argument is true and zero if its argument is false. In other words, the EAF gives for each objective vector in the objective space the relative frequency that it was attained, i.e., weakly dominated by an approximation set, with respect to the r runs.

The outcomes of two optimizers can be compared by performing a corresponding statistical test on the resulting two EAFs, as will be explained in Section 4.4. In addition, EAFs can also be used for visualizing the outcomes of multiple runs of an optimizer. For instance, one may be interested in plotting all the goals that have been attained (independently) in 50% of the runs. This is defined in terms of a *k%-approximation set*:

> An approximation set A is called the *k%-approximation set* of an EAF $\alpha_r(\boldsymbol{z})$, iff it weakly dominates exactly those objective vectors that have been attained in at least k percent of the r runs. Formally,
>
> $$\forall \boldsymbol{z} \in Z \ : \ \alpha_r(\boldsymbol{z}) \geq k/100 \Leftrightarrow A \preceq \{\boldsymbol{z}\} \tag{20}$$

We can then plot the *attainment surface* of such an approximation set, defined as:

> An attainment surface of a given approximation set A is the union of all tightest goals that are known to be attainable as a result of A. Formally, this is the set $\{\boldsymbol{z} \in \mathbb{R}^n : A \preceq \boldsymbol{z} \wedge \neg A \prec\prec \boldsymbol{z}\}$.

Roughly speaking, then, the k%-attainment surface divides the objective space in two parts: the goals that have been attained and the goals that have not been attained with a frequency of at least k percent.

Example 8 *Suppose a stochastic multiobjective optimizer returns the approximation sets depicted in Fig. 8 for five different runs on a biobjective optimization problem. The corresponding attainment surfaces are shown in Fig. 9; they summarize the underlying empirical attainment function.*

A visual representation of an optimizer's performance as shown in the previous example is valuable and complementary to the use of quality indicators for the following reasons.

(i) Decision makers may have preferences towards certain regions of or shapes of Pareto front, not generally (or easily) expressible before optimization, but that are ultimately used to judge the quality of the approximations sets.

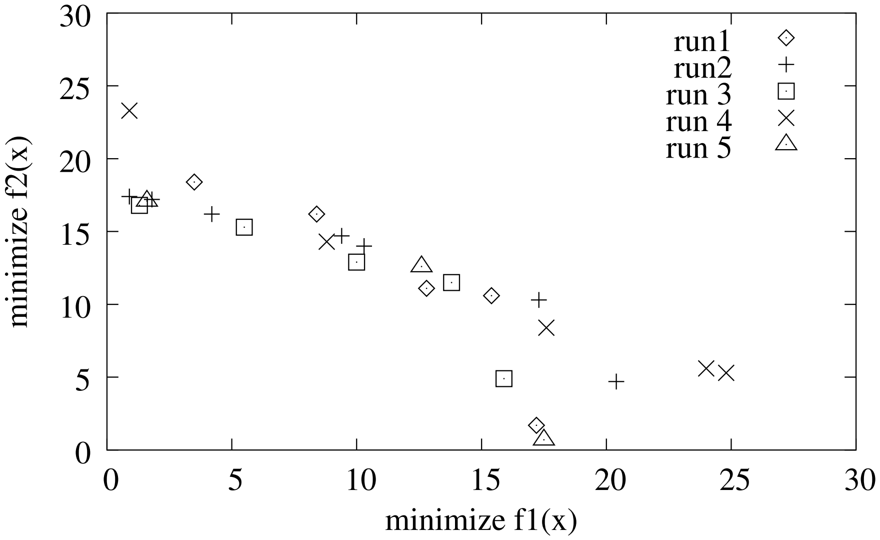


Figure 8: A plot showing five approximation sets. The visual evaluation is difficult, although there are only a few points per set, and few sets.

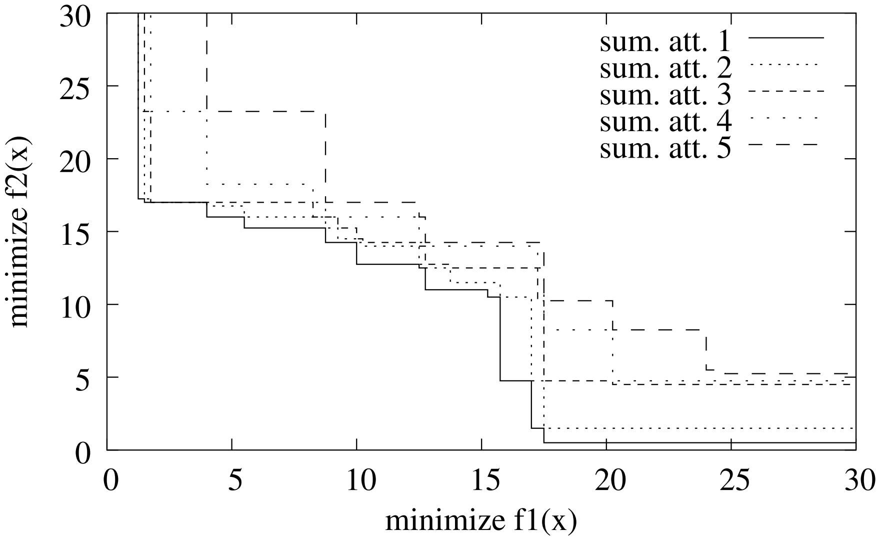


Figure 9: Attainment surface plots for the approximation sets in Figure 8. The first (solid) line represents the 20%-attainment surface, the second line the 40%-attainment surface, and so forth; the fifth line stands for the 100%-attainment surface.

(ii) Some quality indicators do not adequately express the amount by which one approximation set should be judged better than another.

(iii) Looking at approximation set shape can provide insight into the strengths and weaknesses of an optimizer, or provide information about how it is working. For example, an optimizer may converge well in the centre of the Pareto front only, or more at the extremes, perhaps depending on how it balances elitism and diversity mechanisms.

(iv) Visualization methods can provide a 'sanity check' to validate any quality indicators being used.

(v) When the actual Pareto front is known, *seeing* the distance away from it and coverage along it achieved can provide a supplement to any quality indicators used.

3.3.2 Discussion

The attainment function approach distinguishes itself from the dominance ranking and indicator approaches by the fact that the transformed samples are multidimensional, i.e., defined on Z and not on $\mathbb{R}$. Thereby, less information is lost by the transformation, and in combination with a corresponding statistical testing procedure detailed differences can be revealed between the EAFs of two algorithms (see Section 4). However, the approach is computationally expensive and therefore only applicable in the case of a few objective functions.[9] Concerning visualization of EAFs, recently, an approximate algorithm has been presented by Knowles (2005) that computes a given k%-attainment surface only at specified points on a grid and thereby achieves considerable speedups in comparison with the exact calculation of the attainment surface defined above.

4 Statistical Testing

4.1 Fundamentals

The previous section has described three different transformations that can be applied to a sample of approximation sets generated from multiple runs of an optimizer. The ultimate purpose of generating the samples and applying the transformations is to allow us to (a) describe and (b) make inferences about the underlying random approximation set distributions of the (two or more) optimizers, thus enabling us to compare their performance.

It is often convenient to summarise a random sample from a distribution using *descriptive statistics* such as the mean and variance. The mean, median and mode are sometimes referred to as first order moments of a distribution, and they describe or summarise the *location* of the distribution on the real number line. The variance, standard deviation, and inter-quartile range are known as second-order moments and they describe the spread of the data. Using box-plots (Chambers et al. 1983) or tabulating mean and standard deviation values are useful ways of presenting such data.

4.1.1 Statistical inferences

Descriptive statistics are limited, however, and should usually be given only to supplement any statistical inferences that can be made from the data. The standard statistical inference we would like to make, if it is true, is that one optimizer's underlying approximation set *distribution* is better than another one's.[10] However, we cannot determine this fact definitively because we only have access to finite-sized *samples* of approximation sets. Instead, it is standard practice to *assume* that the data is consistent with a simpler explanation known as the *null hypothesis*, H_0, and then to test how likely this is to be true, given the data. H_0 will often be of the form 'samples A and B are drawn from the same distribution' or 'samples A and B are drawn from distributions with the same mean value'. The probability of obtaining a finding at least as 'impressive' as that obtained, assuming the null hypothesis is true, is called the *p-value* and is computed using an inferential *statistical test.* The *significance level*, often denoted as α, defines the largest acceptable *p-value* and represents a threshold that is user-defined. A p-value lower than the chosen significance level α then signifies that the null hypothesis can be rejected in favour of an *alternative hypothesis*, H_A, *at a significance level of* α. The definition of the alternative hypothesis usually takes one of two forms. If H_A is of the form 'sample A comes from a better distribution than sample B' then the inferential test is a *one-tailed test.* If H_A does not specify a prediction about which distribution is better, and is of the form 'sample A and sample B are from different distributions' then it is a two-tailed test. A one-tailed test is more *powerful* than a two-tailed test, meaning that for a given alpha value, it rejects the null hypothesis more readily in cases where it is actually false.

[9] The tools provided by Carlos M. Fonseca in this context of this review, cf. Section 5, are designed for the biobjective case.

[10] Most statistical inferences are formulated in terms of precisely two samples, in this way.

4.1.2 Non-parametric statistical inference: rank and permutation tests

Some inferential statistical tests are based on assuming the data is drawn from a distribution that closely approximates a known distribution, e.g. the normal distribution or Student's t distribution. Such known distributions are completely defined by their *parameters* (e.g. the mean and standard deviation), and tests based on these known distributions are thus termed parametric statistical tests. Parametric tests are powerful—that is, the null hypothesis is rejected in most cases where it is indeed false—because even quite small differences between the means of two normal distributions can be detected accurately. However, unfortunately, the assumption of normality cannot be theoretically justified for stochastic optimizer outputs, in general, and it is difficult to empirically test for normality with relatively small samples (less than 100 runs). Therefore, it is safer to rely on *nonparametric tests* (Conover 1999), which make no assumptions about the distributions of the variables.

Two main types of nonparametric tests exist: rank tests and permutation tests. Rank tests pool the values from several samples and convert them into ranks by sorting them, and then employ tables describing the limited number of ways in which ranks can be distributed (between two or more algorithms) to determine the probability that the samples come from the same source. Permutation tests use the original values without converting them to ranks but estimate the likelihood that samples come from the same source explicitly by Monte Carlo simulation. Rank tests are the less powerful but are also less sensitive to outliers and computationally cheap. Permutation tests are more powerful because information is not thrown away, and they are also better when there are many tied values in the samples, however they can be expensive to compute for large samples.

In the following, we describe our recommended methods for nonparametric inference testing for each of the different transformations. We follow this with a discussion of issues relating to matched samples, multiple inference testing, and assessing worst- and best-case performance.

4.2 Comparing Samples of Dominance Ranks

Dominance ranking converts the samples of approximation sets from two or more optimizers into a sample of dominance ranks. If there are just two optimizers, rank test, e.g. the Mann-Whitney rank sum test (Conover 1999) can be applied to the dominance ranks. A rank test is recommended because it is the ranking established by the dominance ranks, and not the absolute values of the dominance ranks, that we are interested in. If we have more than two optimizers, a Kruskal-Wallis rank test (Conover 1999) can be used.

Note: the ranks that the Mann-Whitney and Kruskal-Wallis tests use will be computed *from* the dominance ranks, treating the latter the same as any figure-of-merit; the dominance ranks themselves are not used by these tests. Because there are likely to be many ties in the dominance ranks, Fisher's permutation test (Efron and Tibshirani 1993, chap. 15) may be a more powerful alternative to a rank test.

4.3 Comparing Sample Indicators Values

4.3.1 Using a Single Indicator

The use of a quality indicator reduces the dimension of an approximation set to a single figure of merit. One of the main advantages, and underlying motivations, for using indicators is that this reduction to one dimension allows statistical testing to be carried out in a relatively straightforward manner using standard univariate statistical tests, i.e. as is done when comparing best-of-population fitness values (or equivalents) in single-objective algorithm comparisons.

As for the dominance ranking approach above, our recommendations are to use the Mann-Whitney rank sum test or Fisher's permutation test. The Kruskal-Wallis test can be used if multiple (more than two) algorithms are to be compared.

4.3.2 Using Several Different Indicators

In many studies on multiobjective algorithm performance, more than one indicator is used to compare approximation sets. For example, it is very common to measure diversity, evenness, and proximity to the Pareto front, using three separate 'functionally independent' indicators. We do not recommend this approach because this kind of indicator is Pareto non-compliant, as we have already stated, but we do recommend using a combination of the *Pareto compliant* indicators defined above. In that case, slightly different preferences are assessed by each of the indicators and this helps to build up a better picture of overall approximation set quality. On the other hand, using several indicators does bring into play two possible pitfalls that should be avoided:

(a) If the locations, e.g., means, of the distributions from two indicators are being tested independently, then one should avoid statements of the form: 'the approximation sets of optimizer A are preferable to those of B under both I_1 and I_2', even if the mean values are significantly higher for both indicators. This is because such a statement could be interpreted as meaning that *individual* approximation sets tend to exhibit a higher quality under *both* indicators, while this is not necessarily true. Rather, the independence of the two results should be stated explicitly, as in: 'the approximation sets of optimizer A are preferable to those of B under I_1 and, independently, under I_2'.

(b) Multiple testing issues, possibly arising from using multiple indicators on the same set of samples, are dealt with, or at least noted (see Section 4.5.2).

4.4 Comparing Empirical Attainment Functions

The EAF of an optimizer is a generalization of a univariate empirical cumulative distribution function (ECDF) (Grunert da Fonseca et al. 2001). In order to test if two ECDFs are different, the Kolmogorov-Smirnov (KS) test can be applied. This test measures the maximum difference between the ECDFs and assesses the statistical significance of this difference. An algorithm that computes a KS-like test for two EAFs is described in (Shaw et al. 1999), and is available in our software package. The test only determines if there is a significant *difference* between the two EAFs, based on the maximum difference. It does not determine whether one algorithm's entire EAF is 'above' the other one:

$$\forall z \in Z, \alpha_r^A(z) \geq \alpha_r^B(z),$$

or not. In order to probe such specific differences, one must use methods for visualizing the EAFs.

For two-objective problems, plotting significant differences in the empirical attainment functions of two optimizers, using a pair of plots, can be done by colour-coding either (i) levels of difference in the sample probability, or (ii) levels of statistical significance of a difference in sample probability, of attaining a goal, for all goals. Option (ii) is more informative and can be computed from the fact that there is a correspondence between the statistical significance level α of the KS-like test (which should not be confused with the EAF α_r) and the maximum distance between the EAFs that needs to be exceeded. Thus the KS-like test can be run for different selected α to compute these different distances. Then, the actual measured distances between the EAFs at every z can be converted to a significance level.

An example of such a pair of plots is shown in Figure 10. This kind of plot has been used to good effect in (López-Ibáñez et al. 2004).

4.5 Advanced Topics

4.5.1 Matched Samples

When comparing a pair of stochastic optimizers, two slightly different scenarios are possible. In one case, each run of each optimizer is a completely independent random sample; that is, the initial population (if appropriate), the random seed, and all other random variables are drawn independently and at random on each run. In the other case, the influence of one or more random variables is partially removed from consideration; e.g. the initial population used by the two algorithms may be matched in corresponding runs, so that the runs (and hence the final quality indicator values) should be taken as pairs. In the former scenario, the statistical testing will reveal, in quite general terms, whether there is a difference in the distributions of indicator values resulting from the two stochastic optimizers, from which a general performance difference can be inferred. In the latter scenario — taking the particular case where initial populations are matched — the statistical testing reveals whether there is a difference in the indicator value distributions *given the same initial population*, and the inference in this case relates to the optimizer's ability to *improve* the initial

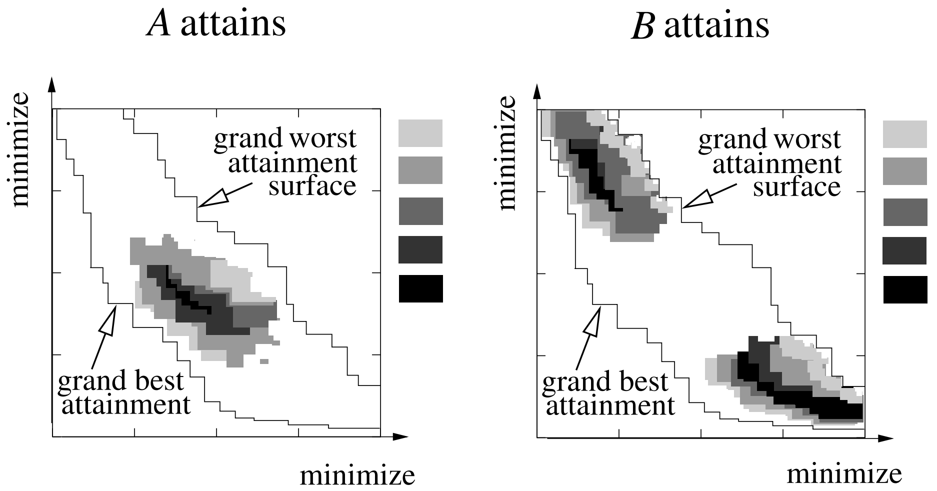


Figure 10: Individual differences between the probabilities of attaining different goals on a two-objective minimization problem with optimizer A and optimizer B, shown using a color-coded plot. The grand attainment surfaces (the same in both plots) indicate the borders beyond which the goals are never attained or always attained, computed from the *combined* collection of approximation sets. Differences in the frequency with which certain goals are met by the respective algorithms A and B are then represented in the region between these two surfaces. In the left plot, darker regions indicate goals that are attained more frequently by A than by B. In the right plot, the reverse is shown. The intensity of the shading can correspond to either the magnitude of a difference in the sample probabilities, or to the level of statistical significance of a difference in these probabilities.

population. While the former scenario is more general, the latter may give more statistically significant results.

If matched samples have been collected, then the Wilcoxon signed rank test (Conover 1999) or Fisher's matched samples test (Conover 1999) can be used instead of the Mann-Whitney rank sum test respectively Fisher's permutation test.

4.5.2 Multiple Testing

The confidence levels resulting from a statistical testing procedure for measuring the differences between distributions only has a meaning if certain assumptions are true. One of these assumptions, which is easy to overlook, is that the data on which the test has been carried out is not being used to make more than one inference. Imagine the same set of data were used to make five different inferences, and each had a p-value of 0.95. The chance that at least one of the inferences will be a type-1 error (i.e. the null hypothesis is wrongly rejected) is $1-(0.95^5) \simeq 23\%$, when assuming that the null hypothesis was true in every case. The situation is made even worse if we only report the cases where the null hypothesis was rejected, and do not report that the other tests were performed: in that case, results can look convincing to a reader when, in fact, they are *not* significant.

Multiple testing issues in the case of assessing stochastic multiobjective optimizers can arise for at least two different reasons:

- There are more than two algorithms and we wish to make inferences about performance differences between all or a subset of them.

- There are multiple hypotheses that we wish to test with the *same* data, e.g., differences in the distributions of more than one indicator.

Clearly, this is a complicated issue and we can only touch on the correct procedures here. The important thing to know is that the issue exists, and to do something to minimize the problem. We briefly consider six possible approaches:

(i) Do all tests as normal (with uncorrected p-values) but report all tests done openly and notify the reader that the significance levels are not, therefore, reliable.

(ii) In the special case where we have multiple algorithms but just one statistic (e.g. one indicator), use a statistical test that is designed explicitly for assessing several independent samples. The Kruskal-Wallis test (Conover 1999), is an extension of the two-sample Mann-Whitney test that works for multiple samples. Similarly, the Friedman test (Conover 1999) extends the paired Wilcoxon signed rank test to any number of related samples.

(iii) In the special case where we want to use multiple statistics (e.g. multiple different indicators) for just two algorithms, and we are interested *only* in an inference derived *per-sample* from all statistics, (e.g. we want to test the significance of a difference in hypervolume between those pairs A_i and B_i *where* the diversity difference between them is positive), then the permutation test can be used to derive the null distribution, as usual.

(iv) Minimize the number of different tests carried out on the same data by carefully choosing which tests to apply before collecting the data.

(v) For each test to carry out, generate new independent data which is not to be used for any other test.

(vi) Apply the tests on the same data but use methods for correcting the p-values for the reduction in confidence associated with data re-use.

Approach (i) can be done by anyone with minimal fuss; it does not allow powerful conclusions to be drawn, but it at least avoids mis-representation of results. The second approach is quite restrictive as it only applies to a single test being applied to multiple algorithms — and uses rank tests, which might not be appropriate in all circumstances. Similarly, (iii) only applies in the special case noted. A more general approach is (iv), which is just the conservative option. It says don't do tests unless there is some realistic chance that the null hypothesis can be rejected (and the result would be interesting). This careful conservatism can then be accommodated by doing (v) — generating new data for each test. However, while following (iv) and (v) might be possible much of the time, sometimes it is essential to do several tests on limited data and to be as confident as possible about any positive results. In this case, one must then use approach (vi).

The simplest and most conservative, i.e., weakest approach for this is the Bonferroni correction. Suppose we would like to consider an overall significance level of α and that altogether n comparisons, i.e., distinct statistical tests, are performed per sample. Then, the significance level α_s for each distinct test is set to

$$\alpha_s = \frac{\alpha}{n} \tag{21}$$

Explicitly, given n tests T_i for hypotheses $H_i (1 \leq i \leq n)$ under the assumption H_0 that all hypotheses H_i are false, and if the individual test critical values are $\leq \alpha/n$, then the experiment-wide critical value is $\leq \alpha$. In equation form, if

$$P(T_i \text{ passes } \mid H_0) \leq \frac{\alpha}{n} \text{ for } 1 \leq i \leq n, \tag{22}$$

then

$$P(\text{some } T_i \text{ passes } \mid H_0) \leq \alpha. \tag{23}$$

In most cases, the Bonferroni approach is too weak to be useful and other methods should be used (Perneger 1998), e.g., resampling based methods (Westfall and Young 1993).

4.5.3 Assessing Worst-case or Best-case Performance

In certain circumstances, it may be important to compare the worst-case or best-case performance of two optimizers. Obtaining statistically significant inferences for these is more computationally demanding than when assessing differences in mean or typical performance, however, it can be done using permutation methods, such as bootstrapping or variants of Fisher's permutation test (Efron and Tibshirani 1993, chap. 15).

For example, let us say that we wish to estimate whether there is a difference in the expected worst indicator value of two algorithms, when each is run ten times. To assess this: run each algorithm for 30 batches of 10 runs, and find the mean of the worst-in-a-batch value, for each algorithm. Then, to compute the null distribution, permute the labels of all 600 samples randomly, and find the worst indicator value from those with a label in $1, \dots, 10$. By sampling this statistic very many times, the desired p-value that the mean of the worst-in-a-batch statistics are significantly different, can be computed. Quite obviously, such a testing procedure is very general and it can be tailored to answer many questions related to worst-case or best-case performance.

5 Case Study

The case study will follow the chain of processes/tools shown in Fig. 11. The necessary tool environment including statistical methods, optimization algorithms and test problems are based on the Platform and Programming Language Independent Interface PISA (Bleuler et al. 2003) and available for download from *http:/www.tik.ee.ethz.ch/pisa/*. For all tools, it is supposed that the objective functions are to be minimized.

The starting point is the set of runs for all combinations of optimization methods and test problems that will be considered. As a result, one obtains a set of files, each one containing the approximation sets for all runs that have been generated by a particular selector-variator pair after the same number of generations. Here, the terms selector and variator denote the optimizer and the test problem (including variation operators), respectively.

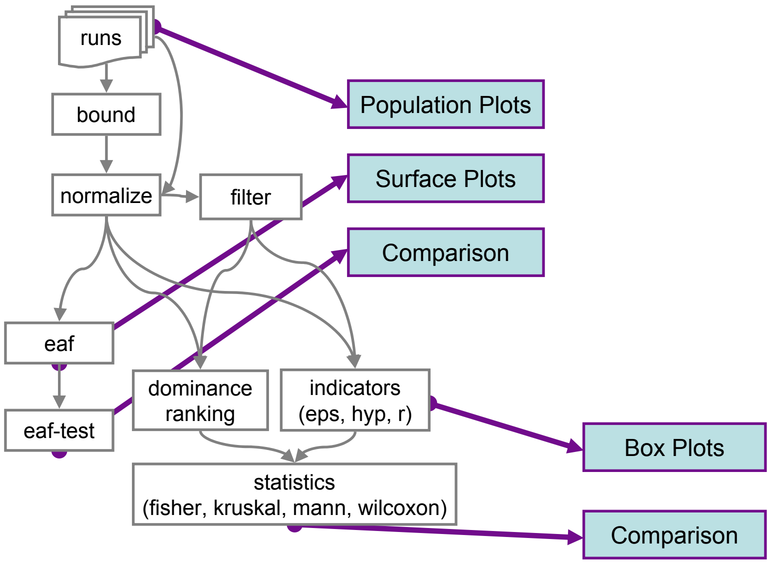


Figure 11: Overview of the tools accompanying this paper.

We will use three selectors (optimizers), i.e., NSGA-II (Deb et al. 2000), SPEA2 (Zitzler et al. 2002), and IBEA (Zitzler and Künzli 2004). As test problems (variators), we consider DTLZ2 (3 objectives, 100 decision variables) (Deb et al. 2002), ZDT6 (2 objectives, 100 decision variables) (Zitzler et al. 2000), QV (a test problem proposed by Quagliarella and Vicini with 2 objectives and 100 decision variables) (Zitzler et al. 2002), and the multiobjective knapsack problem (2 objectives, 500 decision variables) (Zitzler and Thiele 1999). Note that the knapsack problem used here implements a slighlty modified version of the problem presented in (Zitzler and Thiele 1999). The ith objective function f_i does not give the overall profit with

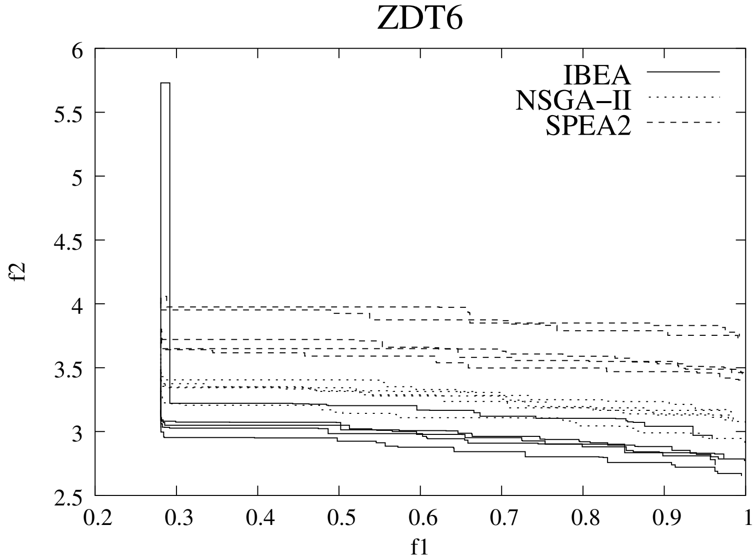


Figure 12: Attainment surface plots for the first five approximation sets generated by IBEA, SPEA2, and NSGA-II for the ZDT6 problem.

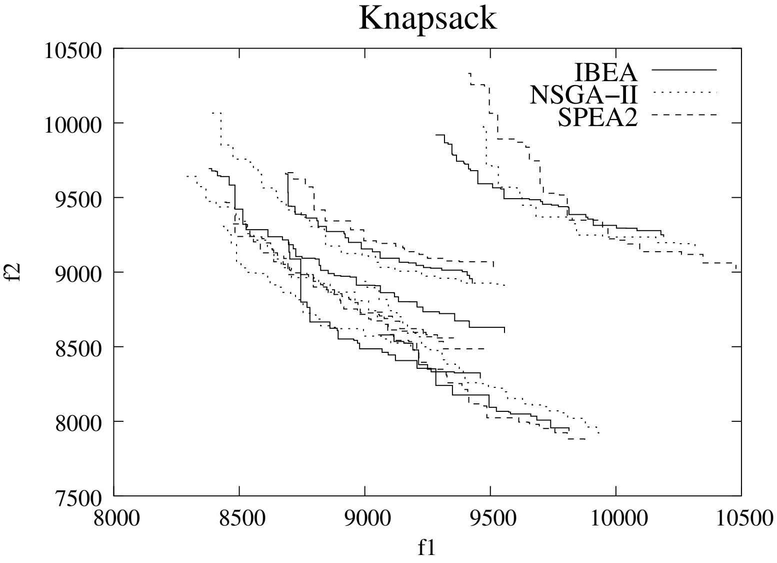


Figure 13: Attainment surface plots for the first five approximation sets generated by IBEA, SPEA2, and NSGA-II for the knapsack problem.

respect to knapsack i, but the difference between the maximum achievable profit minus the actual profit for knapsack i; therefore, all objectives are to be minimized.

All nine selector-variator pairs have been evaluated, each one with 30 runs and 200 generations. According to Fig. 11, one can at first try to visualize individual runs using population (approximation set) plots. As can be seen from Fig. 12, sometimes it appears that certain statements about the quality of runs can be made. One may guess that IBEA outperforms NSGA-II which outperforms SPEA2 on ZDT6 (the attainment surfaces shown correspond to the approximation sets from the first five runs with regard to the last generation). In case of the knapsack problem in Fig. 13, no conclusion is possible on the basis of the population plots. In addition, in case of higher dimensional objective spaces, a visual comparison is not intuitive anymore.

There are three tools that perform a post-processing on the approximation sets found by the three algorithms:

- `bound`: This program calculates lower and upper bounds of the objective vectors. To this end, all approximation sets for a specific generation and a particular optimization problem are collected, and the maximal and minimal values in each objective dimension are determined.

- `normalize`: Based on the bounds determined above, the program transforms all objective vectors contained in a given file in such a way that (i) all values lie in the interval $[1, 2]$, and (ii) all objectives are to be minimized. This way, the assessment tools give reliable results if different selectors are compared for a specific problem.

- `filter`: This program determines the objective vectors that are not dominated among all the objective vectors stored in a given file. It is used in order to generate a reference set for the unary indicators. To this end, all normalized objective vectors from all populations are collected and the objective vectors not dominated are extracted.

Based on the normalized approximation sets for each optimization problem, the dominance ranking procedure, which is implemented by the program `dominance-rank`, can be used to compare the different selectors. Here, we consider algorithms in ordered pairs and apply the one-tailed Mann-Whitney rank sum test to the results. The uncorrected p-values for these tests are shown in Table 3. None of the results for DTLZ2, Knapsack and ZDT6 is statistically significant regarding an overall significance level $\alpha = 0.05$, which yields a significance level $\alpha_s = 0.05/4 = 0.0125$ per single test using Bonferroni correction with 4 one-sided tests being performed per sample. This indicates that no selector-variator generates approximation sets that are significantly *better*, cf. Table 2, than another selector-variator, on any of these three problems. For ZDT6, this outcome may be suprising, because Fig. 12 suggests that IBEA outperforms SPEA2 in this setting. However, as the corresponding attainment surfaces are crossing each other on the left hand side, all approximation sets are incomparable and therefore no significant differences can be detected. This yields a situation similarly to the one depicted in Fig. 3 on the right. On the other hand, for the QV problem, IBEA is significantly better than NSGA2 as well as SPEA2.

As a next step, unary quality indicators can be applied using the normalized approximation sets and the reference set for each optimization problem. The PISA performance assessment tools currently contain the programs `eps_ind` (unary epsilon indicators I^1_ϵ and $I^1_{\epsilon+}$), `hyp_ind` (unary hypervolume indicator I_H) and `r_ind` (R indicators I^1_{R2} and I^1_{R3}). Here, we use the $I^1_{\epsilon+}$ indicator, the I^-_H indicator with a reference point of $(2.1, 2.1, \ldots, 2.1)$ where the indicator value is the difference in hypervolume between the reference set (recall that the reference set here is the set of pooled nondominated vectors) and the approximation set under consideration, and the I^1_{R2} indicator in combination with the augmented Tchebycheff utility functions where $\boldsymbol{z}^* = (1, 1, \ldots, 1)$, $\rho = 0.01$ and Λ consists of 500 uniformly dispersed normalized weight combinations. The distribution of the resulting transformed approximation sets can be graphically visualized using conventional box plots (see Fig. 14).

Next, non-parametric statistical tests can be applied to the transformed approximation sets in order to obtain valid statements about the quality of optimization methods as applied to selected optimization problems. To this end, various programs are provided (for further information, see (Conover 1999)):

- `fisher-indep`: It implements a non-parametric test for differences between precisely two independent samples. The output is the p-value of the one-tailed test.

- `fisher-matched`: It implements a non-parametric test for differences between two paired (or matched) samples. The output is the p-value of the one-tailed test.

- `kruskal-wallis`: It implements a non-parametric test for differences between multiple independent samples. If and only if a first test for significance of any differences between the samples is passed, at the given alpha value, then the output will be the one-tailed p-values for each pair-wise combination.

- `mann-whit`: It implements a non-parametric test for differences between precisely two independent samples. The output is the p-value of the one-tailed test for each pair of sample populations.

- `wilcoxon-sign`: It implements a non-parametric test for differences between two paired (or matched) samples. If the sample size $s \geq 50$, then the normal approximation is used. If $s < 50$, the exact quantiles are taken from a table. In this case, only the lowest from a set of critical p-values that

Table 3: Dominance ranking results using Mann-Whitney rank sum test. The tables contain for each pair of optimizers O_R (row) and O_C (column) the p-values with respect to the alternative hypothesis H_A that the dominance ranks for O_R are significantly better than those for O_C. On ZDT6 and DTLZ2 no approximation set dominates any other one and so all have the same rank, 1, and hence no differences between the algorithms are found. For the knapsack problem, differences between the algorithms are discovered but none of them is statistically significant ($\alpha = 0.05$). In case of the QV problem, there are statistically significant differences (IBEA outperforms SPEA2 and NSGA2).

Knapsack			
	IBEA	NSGA-II	SPEA2
IBEA	–	0.28	0.19
NSGA-II	0.72	–	0.36
SPEA2	0.81	0.64	–

QV			
	IBEA	NSGA-II	SPEA2
IBEA	–	0.002	$2.11e^{-5}$
NSGA-II	0.998	–	0.24
SPEA2	1	0.76	–

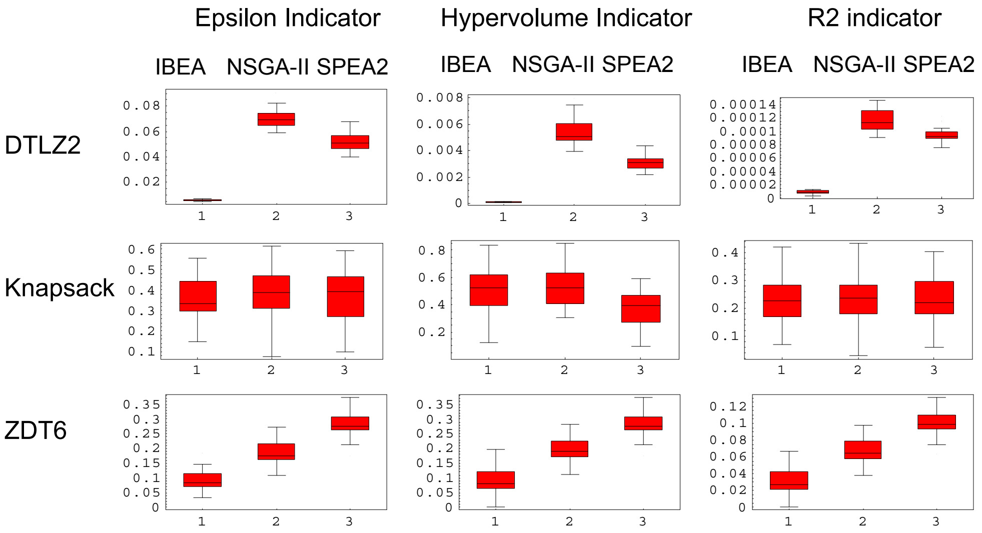


Figure 14: Box plots for different unary indicators.

Table 4: Kruskal-Wallis test applied to selected test problems and unary indicators. The tables contain for each pair of optimizers O_R (row) and O_C (column) the p-values with respect to the alternative hypothesis H_A that the indicator values for O_R are significantly better than those for O_C.

DTLZ2 / I^1_{R2}			
	IBEA	NSGA-II	SPEA2
IBEA	–	$1.7e^{-43}$	$9.6e^{-23}$
NSGA-II	> 0.05	–	> 0.05
SPEA2	> 0.05	$6.0e^{-22}$	–

Knapsack / I^-_H			
	IBEA	NSGA-II	SPEA2
IBEA	–	> 0.05	> 0.05
NSGA-II	> 0.05	–	> 0.05
SPEA2	> 0.05	> 0.05	–

QV / $I^1_{\epsilon+}$			
	IBEA	NSGA-II	SPEA2
IBEA	–	$6.7e^{-18}$	$4.8e^{-24}$
NSGA-II	> 0.05	–	0.0015
SPEA2	> 0.05	> 0.05	–

ZDT6 / $I^1_{\epsilon+}$			
	IBEA	NSGA-II	SPEA2
IBEA	–	$2.7e^{-16}$	$1.2e^{-34}$
NSGA-II	> 0.05	–	$3.7e^{-16}$
SPEA2	> 0.05	> 0.05	–

exceeds the ranksum is returned: no interpolation of values in the table is used. The output is the one-tailed p-value for each pair.

As an example, the Kruskal-Wallis test has been applied to a selected set of indicators and test problems. Tab. 4 shows for example that IBEA performs significantly better on ZDT6 than NSGA-II and SPEA2 with respect to the additive epsilon indicator $I^1_{\epsilon+}$ and a significance level α of 5%. Moreover, on DTLZ2, SPEA2 performs significantly better than NSGA-II regarding the I^1_{R2} indicator ($\alpha = 0.05$). As could have been guessed from Fig. 13, there are no significant differences between the algorithms on the knapsack problem for the hypervolume indicator ($\alpha = 0.05$). For the QV problem, IBEA performs significantly better than NSGA2 as well as SPEA2, and NSGA2 outperforms SPEA2 with respect to the additive epsilon indicator $I^1_{\epsilon+}$ (even though the dominance ranking results do not show a significant difference between NSGA2 and SPEA2 on this problem). Note that here no adjustement of the significance level is necessary, e.g., according to Bonferroni correction, as the Kruskal-Wallis test is used and only one indicator was applied to each problem.

Finally, empirical attainment functions (see Fig. 11) can also be used to obtain complementary performance assessment results. To this end, the program `eaf` computes the attainment surface for a given set of runs. Figure 15 visualizes the 50%-attainment surface as computed by `eaf` for the ZDT6 and knapsack problems. Again, one could guess that there are significant differences in the case of ZDT6, but not for the knapsack problem. This can be statistically verified by applying the Kolmogorov-Smirnov test (Shaw et al. 1999) that is implemented in the program `eaf-test`. As a result, one obtains that on ZDT6 there is a significant difference between all pairs and on knapsack there is no significant difference between any pair—considering a signifance level of 5%.

6 Summary

This papers deals with the issue of performance assessment of stochastic multiobjective optimizers for optimization scenarios that are based on the concept of Pareto dominance. It reviews and discusses the two

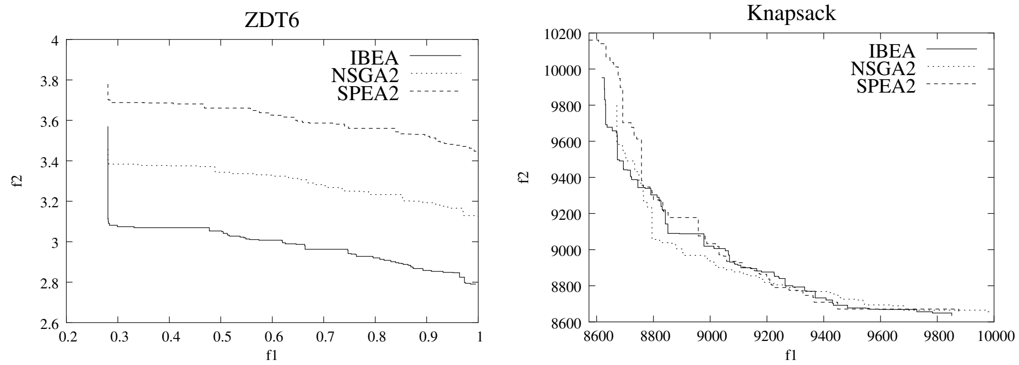


Figure 15: 50%-attainment surface plots for the optimization problems ZDT6 and knapsack.

current principal approaches, the quality indicator method and the attainment function method, and in addition proposes a third approach, the dominance-ranking technique.

Dominance ranking stands for a general, preference-independent assessment method that is based on pairwise comparisons of approximation sets, similarly to Pareto dominance-based fitness assignment in evolutionary multiobjective optimization. In detail, all approximation sets generated by the different optimizers under consideration are collected, pairwisely compared, and ranked according to the number of approximation sets by which a specific set is dominated. As a result, each algorithm is associated with a sample of ranks where the ranks are to be minimized. In order to make relative statements of outperformance, one needs to statistically compare the various rank samples. The advantage of this approach is that it is based only on Pareto dominance relations between sets and a ranking procedure, and so is not biased with respect to any decision maker's preferences. On the other hand, it also represents the least informative among the three approaches as differences in quality cannot be quantified respectively localized.

Quality indicators represent a means to express and measure quality differences between approximation sets—on the basis of additional preference information. When using unary quality indicators, i.e., functions that map each approximation set to a real number as a single figure of merit, the algorithms are assigned samples of indicator values. On the basis of statistical testing procedures, it is then possible to check whether an algorithm provides significantly better approximation sets than another optimizer *with respect to the preferences represented by the considered indicator*. It is crucial, though, to use only quality indicators that are compliant with Pareto dominance: indicators that have the property that whenever an approximation set is preferred over another set regarding the indicator, then the former approximation must not be dominated by the latter. Only a few quality indicators proposed in the literature are Pareto compliant; three of them have been discussed in this paper. The quality indicator approach is attractive because it is usually feasible also for a high number of objectives (though some indicators are computationally more expensive than others) and provides quantitative information about differences in quality. However, the statements possible always need to be seen in the context of the preferences considered; furthermore, the transformation of approximation sets into real numbers inevitably means a loss in information, and therefore it is still possible that quality differences exist that cannot be detected by the indicators used.

Attainment functions give for each vector in the objective space the probability that it is attained, i.e., weakly dominated, by the approximation set generated by one run of an optimizer. An *empirical attainment function* represents an estimation of the usually unknown attainment function and summarizes the outcomes of multiple runs of one algorithm by, roughly speaking, calculating for each objective vector the relative frequency that it has been attained. In contrast to the other two approaches, only little information is lost by this transformation and accordingly the statistical comparison of two empirical attainment functions can reveal a lot of information concerning where the outcomes of two algorithms differ. This is especially useful in algorithm development. In this sense, the attainment function approach is more sensitive to differences in the quality than the other two assessment methods. However, the disadvantage is that it is computationally demanding and currently restricted to two and three objectives, from a practical point of view.

In the light of the above discussion, it becomes clear that there is no 'best' performance assessment technique. Instead, we generally recommend to use the complementary strengths of the three approaches. As a first step in a comparison, any significant differences between the optimizers considered should be probed using the dominance-ranking approach, because such an analysis allows the strongest type of statements to be made. Quality indicators can then be applied in order to quantify the potential differences in quality and to detect differences that could not be revealed by dominance ranking. The corresponding statements are always restricted as they only hold for the preferences that are represented by the indicators used. The computation and the statistical comparison of the empirical attainment functions are especially useful in terms of visualization and to add another level of detail; for instance, plotting the regions of significant difference gives information on *where* the outcomes of two algorithms differ. Important, though, for all three approaches is that the comparisons are made on the basis of appropriate statistical testing procedures, taking specific issues such as multiple testing into account, and that the corresponding significance values are reported in the paper.

Finally, note that there are several issues that have not been treated in the present review, e.g., binary quality indicators and indicators taking the decision vectors into account. Many of these issues represent current research directions which will probably lead to modified or additional performance assessment methods in the near future.

Acknowledgments

The authors would like to thank Carlos M. Fonseca for valuable discussions and for providing the EAF tools.

References

Bleuler, S., M. Laumanns, L. Thiele, and E. Zitzler (2003). PISA — a platform and programming language independent interface for search algorithms. In C. M. Fonseca, P. J. Fleming, E. Zitzler, K. Deb, and L. Thiele (Eds.), *Evolutionary Multi-Criterion Optimization (EMO 2003)*, Lecture Notes in Computer Science, Berlin, pp. 494–508. Springer.

Chambers, J., W. Cleveland, B. Kleiner, and P. Tukey (1983). *Graphical Methods for Data Analysis.* Belmont, CA: Wadsworth.

Coello Coello, C. A., D. A. Van Veldhuizen, and G. B. Lamont (2002). *Evolutionary Algorithms for Solving Multi-Objective Problems.* New York: Kluwer Academic Publishers.

Conover, W. J. (1999). *Practical Nonparametric Statistics* (Third ed.). New York, NY: John Wiley and Sons.

Deb, K. (2001). *Multi-Objective Optimization using Evolutionary Algorithms.* Chichester, UK: John Wiley & Sons.

Deb, K., S. Agrawal, A. Pratap, and T. Meyarivan (2000). A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: NSGA-II. In M. Schoenauer et al. (Eds.), *Parallel Problem Solving from Nature (PPSN VI)*, Lecture Notes in Computer Science Vol. 1917, pp. 849–858. Springer.

Deb, K., L. Thiele, M. Laumanns, and E. Zitzler (2002). Scalable multi-objective optimization test problems. In *Congress on Evolutionary Computation (CEC)*, pp. 825–830. IEEE Press.

Efron, B. and R. Tibshirani (1993). *An introduction to the bootstrap.* London: Chapman and Hall.

Fonseca, C. M. and P. J. Fleming (1993). Genetic Algorithms for Multiobjective Optimization: Formulation, Discussion and Generalization. In S. Forrest (Ed.), *Proceedings of the Fifth International Conference on Genetic Algorithms*, San Mateo, California, pp. 416–423. University of Illinois at Urbana-Champaign: Morgan Kauffman Publishers.

Fonseca, C. M., V. Grunert da Fonseca, and L. Paquete (2005, March). Exploring the performance of stochastic multiobjective pptimisers with the second-order attainment function. In C. A. Coello Coello, A. Hernández Aguirre, and E. Zitzler (Eds.), *Evolutionary Multi-Criterion Optimization. Third International Conference, EMO 2005*, Guanajuato, México, pp. 250–264. Springer. Lecture Notes in Computer Science Vol. 3410.

Goldberg, D. E. (1989). *Genetic Algorithms in Search, Optimization and Machine Learning.* Reading, Massachusetts: Addison-Wesley Publishing Company.

Grunert da Fonseca, V., C. M. Fonseca, and A. O. Hall (2001). Inferential performance assessment of stochastic optimisers a nd the attainment function. In E. Zitzler, K. Deb, L. Thiele, C. A. C. Coello, and D. Corne (Eds.), *Proceedings of the First International Conference on Evolutionary Multi-Criterion Optimization (EMO 2001)*, Volume 1993 of *Lecture Notes in Computer Science*, Berlin, pp. 213–225. Springer-Verlag.

Hansen, M. P. and A. Jaszkiewicz (1998). Evaluating the quality of approximations to the non-dominated set. Technical Report IMM-REP-1998-7, Technical University of Denmark.

Knowles, J. (2005). A summary-attainment-surface plotting method for visualizing the performance of stochastic multiobjective optimizers. In *Computational Intelligence and Applications (Proceedings of the Fifth International Workshop on Intelligent Systems Design and Applications: ISDA'05).*

Knowles, J. and D. Corne (2002). On metrics for comparing non-dominated sets. In *Congress on Evolutionary Computation (CEC 2002)*, Piscataway, NJ, pp. 711–716. IEEE Press.

López-Ibáñez, M., L. Paquete, and T. Stützle (2004). Hybrid population-based algorithms for the bi-objective quadratic assignment problem. Technical Report AIDA–04–11, FG Intellektik, FB Informatik, TU Darmstadt. Accepted by Journal of Mathematical Modelling and Algorithms.

Okabe, T., Y. Jin, and B. Sendhoff (2003). A critical survey of performance indices for multi-objective optimization. In *Proceedings of the IEEE Congress on Evolutionary Computation (CEC 2003)*, pp. 878–885. IEEE Press.

Perneger, T. V. (1998). What's wrong with Bonferroni adjustments. *British Medical Journal 316*, 1236–1238.

Sayin, S. (2000). Measuring the quality of discrete representations of efficient sets in multiple objective mathematical programming. *Mathematical Programming 87*(3), 543–560.

Schott, J. R. (1995). Fault Tolerant Design Using Single and Multicriteria Genetic Algorithm Optimization. Master's thesis, Department of Aeronautics and Astronautics, Massachusetts Institute of Technology, Cambridge, Massachusetts.

Shaw, K. J., A. L. Nortcliffe, M. Thompson, J. Love, C. M. Fonseca, and P. J. Fleming (1999). Assessing the Performance of Multiobjective Genetic Algorithms for Optimization of a Batch Process Scheduling Problem. In *1999 Congress on Evolutionary Computation*, Washington, D.C., pp. 37–45. IEEE Service Center.

Veldhuizen, D. A. V. (1999). *Multiobjective Evolutionary Algorithms: Classifications, Analyses, and New Innovations.* Ph. D. thesis, Department of Electrical and Computer Engineering. Graduate School of Engineering. Air Force Institute of Technology, Wright-Patterson AFB, Ohio.

Westfall, P. H. and S. S. Young (1993). *Resampling-based multiple testing.* New York, NY: Wiley.

While, L. (2005). A new analysis of the lebmeasure algorithm for calculating hypervolume. In *Evolutionary Multi-Criterion Optimization (EMO 2005)*, Number 3410 in Lecture Notes in Computer Science, pp. 326–340. Springer.

While, L., L. Bradstreet, L. Barone, and P. Hingston (2005). Heuristics for optimising the calculation of hypervolume for multi-objective optimisation problems. In *Proceedings of the IEEE Congress on Evolutionary Computation (CEC 2005)*, pp. 192–199. IEEE Press.

Wu, J. and S. Azarm (2001). Metrics for Quality Assessment of a Multiobjective Design Optimization Solution Set. *Transactions of the ASME, Journal of Mechanical Design 123*, 18–25.

Zitzler, E., K. Deb, and L. Thiele (2000). Comparison of multiobjective evolutionary algorithms: Empirical results. *Evolutionary Computation 8*(2), 173–195.

Zitzler, E. and S. Künzli (2004). Indicator-based selection in multiobjective search. In X. Yao et al. (Eds.), *Parallel Problem Solving from Nature (PPSN VIII)*, Berlin, Germany, pp. 832–842. Springer-Verlag.

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

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

Zitzler, E., L. Thiele, M. Laumanns, C. M. Foneseca, and V. Grunert da Fonseca (2003). Performance assessment of multiobjective optimizers: An analysis and review. *IEEE Transactions on Evolutionary Computation 7*(2), 117–132.

A A Summary of 'Functionally-Independent' Quality Indicators

Cardinality-based indicators

Principles: Based on counting the number of solutions in an approximation set, or some subset of it.

Instances:

- $ONVG$ (Veldhuizen 1999), gives the number of internally nondominated points in the approximation set. Pareto non-compliant.
- error ratio (Veldhuizen 1999) counts the number of points that are *not* true Pareto optima, and divides by the total number of points. Pareto-compliant; can only be used, if the Pareto front is known and the algorithms actually reach the front.
- NDC (Wu and Azarm 2001) counts the number of grid regions that are occupied by a point, where there are $1/\mu^m$ grid regions dividing up the m-dimensional objective space, and μ is a user-defined parameter. Pareto non-compliant.
- cluster (Wu and Azarm 2001) measures the ratio N/NDC, where N is the cardinality of the approximation set, and NDC is defined above. Pareto non-compliant.

Advantages and caveats: Cheap to compute and free from problems with normalization and scaling of objectives.

Distance-based indicators for proximity

Principles: The idea is to measure the intuitive notion of an approximation set's distance from the true Pareto front (if known), or some other reference set of points.

Instances:

- generational distance (Veldhuizen 1999) measures the root-mean-square of the distances of points in the approximation set from their nearest point in the true Pareto front. Pareto non-compliant.
- maximum Pareto front error (Veldhuizen 1999) measures the largest distance between a vector in the approximation set and the corresponding closest point in the true Pareto front. Pareto non-compliant.
- coverage error (Sayin 2000) is the same as Maximum Pareto front error, except generalized to the case of continuous Pareto fronts. Pareto non-compliant.

Advantages and caveats: There are issues with normalization, scaling, and other factors relating to measuring distances in possibly non-commensurable objectives.

Distance-based indicators for evenness and diversity

Principles: The idea is to measure the intuitive notion of the evenness of the location of points, and/or their extent in objective space.

Instances:

- spacing (Schott 1995). Pareto non-compliant.
- maximum spread (Wu and Azarm 2001). Pareto non-compliant.
- deviation from uniform distribution (Deb 2001). Pareto non-compliant.
- minimum distance between two solutions (Sayin 2000). Pareto non-compliant.

Advantages and caveats: These have an intuitive meaning but there are issues with normalization, scaling, and other factors relating to measuring distances in possibly non-commensurable objectives.

Figure 16: A summary of 'functionally-independent' quality indicators. Most of these are *Pareto non-compliant* in the sense defined on page 8.

B An Empirical Evaluation of Quality Indicators

One might argue that it is not really important for quality indicators to be *Pareto compliant* in the sense discussed on page 8. If Pareto non-compliant indicators were only to contradict the dominance ordering in certain pathological cases it might be acceptable to use them when there are other good reasons to appreciate them—for instance, if they are computationally efficient, or if the property that they evaluate accords well with our intuitive notion of a good approximation set.

In this appendix, we investigate this issue by conducting an empirical evaluation of four *Pareto non-compliant* quality indicators. The results of this study show that the quality indicators considered do *frequently* contradict the dominance partial ordering even when *non-pathological* approximation sets are considered.

The evaluation is based on randomly generating pairs of approximation sets A and B, with the property that set A is always *better* than set B in the dominance sense, and evaluating the two sets using a number of quality indicators, (see Alg. 1). Because the sets are generated randomly, using a scheme which does not obviously construct better distributions of points for B than for A, we avoid any charge that the cases where B is evaluated better than A (by some quality indicator) are 'pathological' or artificially constructed. Moreover, because we repeat the process a large number of times, we can estimate the 'error-rate' of the measures quantitatively for the first time—albeit only with respect to the scheme we are using to generate these approximation sets.

Algorithm 1 Empirical testing pseudocode

```
for each quality indicator $I$ do
    Error_rate($I$) $\leftarrow 0$
end for
for each $i$ in 1 to $10,000$ do
    randomly generate two approximation sets $A$ and $B$, such that $A \lhd B$.
    for each quality indicator $I$ do
        evaluate $I(A)$ and $I(B)$
        if $B$ preferred to $A$ by $I$ then Error_rate($I$) $\leftarrow$ Error_rate($I$) $+ 1/10,000$
        end if
    end for
end for
```

B.1 Generating the Approximation Sets

The procedure for generating the approximation sets is given in Alg. 2. Steps 2 to 11 create two random sample populations of objective vectors from the same distribution. The Pareto front of the distribution has the equation $z^2 = 1 - \sqrt{z^1}$, which is the same Pareto front as is used in the ZDT2 test function. The further away a point is from this Pareto front (in the z^2 objective) the more probable it is, which creates a 'gradient' away from the Pareto front, as is normal in most optimization problems. The two sample populations A and B are then filtered to remove dominated vectors in Steps 12 and 13, respectively. Finally, in the last step all objective vectors in B that are not dominated by the set A are removed from B. This ensures that the set A dominates the set B.

B.2 Quality Indicators

Four quality indicators are evaluated in this study, as follows.

B.2.1 Generational Distance (GD)

Generational distance (Veldhuizen 1999) measures the root-mean-square of the distances of points in the approximation set from their nearest point in the true Pareto front. It is to be minimized.

Algorithm 2 Generating approximation sets A and B. The function U[.,.) returns a uniform random deviate in the specified semi-open interval.

```
$A \leftarrow \emptyset,\ B \leftarrow \emptyset$
for each $i$ in 1 to $1,000$ do
    $z^1 \leftarrow U[0,1),\ z^2 \leftarrow U[0,5),\ z \leftarrow (z^1, z^2)$
    if min $\left[0, (z^2 - (1 - \sqrt{z^1}))/5.0\right] < U[0,1)$ then
        $A \leftarrow A \cup \{z\}$
    end if
    $z^1 \leftarrow U[0,1),\ z^2 \leftarrow U[0,5), z \leftarrow (z^1, z^2)$
    if min $\left[0, (z^2 - (1 - \sqrt{z^1}))/5.0\right] < U[0,1)$ then
        $B \leftarrow B \cup \{z\}$
    end if
end for
$A \leftarrow \{z \in A : A \not\prec \{z\}\}$
$B \leftarrow \{z \in B : B \not\prec \{z\}\}$
$B \leftarrow \{z \in B : A \prec \{z\}\}$
```

B.2.2 Deb et al's Spacing Metric (Spacing)

This spacing indicator (Deb et al. 2000) measures the average discrepancy between the spacing of consecutive points, and the mean spacing of consecutive points. It is to be minimized. Note: in our computation of it we do not take into account the boundaries of the Pareto front.

B.2.3 Maximum Pareto Front Error (MPFE)

The maximum Pareto front error (Veldhuizen 1999) measures the largest distance between a vector in the approximation set and the corresponding closest point in the true Pareto front. It is to be minimized.

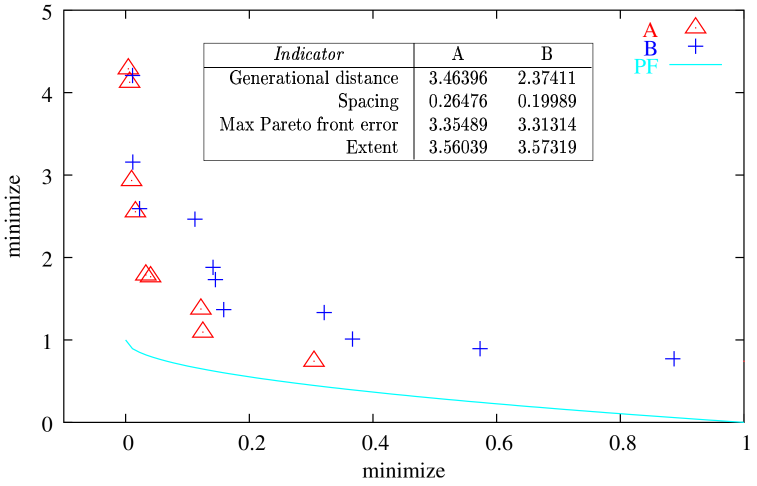
B.2.4 Extent of the Approximation Set (Extent)

This indicator simply measures the largest (normalized) Euclidean distance between a pair of points in the approximation set. It is to be maximized. This indicator is not derived from any used in the literature but follows the principle, e.g. referred to in (Deb 2001), and often repeated, that a good approximation set should have a good distribution in terms of spread (as well as evenness).

B.3 Results

An example of a case for which all four of the indicators simultaneously judge B to better than A is shown in Figure 17. Obviously, the existence of such cases within a fairly small random sampling of approximation sets indicates the degree to which these indicators are Pareto non-compliant, even when used in concert.

Table 5 gives the independent error rates for each of the indicators, and the mean number of errors made (out of a possible 4.0) per pair of approximation sets. It can be seen that two of the indicators are wrong more frequently than they are right. Even in concert, these measures clearly have a strong tendency to mislead.



Indicator	A	B
Generational distance	3.46396	2.37411
Spacing	0.26476	0.19989
Max Pareto front error	3.35489	3.31314
Extent	3.56039	3.57319

Figure 17: Two approximation sets A and B with $A \lhd B$. All four of the indicators judge B to be better, simultaneously.

Table 5: Error rates for each indicator (independently) and the overall mean number of errors per pair of approximation sets. The reader should note that quality indicator such as the hypervolume indicator and epsilon indicator would give a value of zero in the error rate row, since these are Pareto compliant in the sense defined in Section 3.2.

	Indicator			
Statistic	GD	Spacing	MPFE	Extent
Error rate	0.598	0.270	0.621	0.231
Overall mean number of errors	1.720			