14

Quality Assessment of Pareto Set Approximations

Eckart Zitzler[1], Joshua Knowles[2], and Lothar Thiele[1]

[1] ETH Zurich, Switzerland
eckart.zitzler@tik.ee.ethz.ch, thiele@tik.ee.ethz.ch
[2] University of Manchester, UK
j.knowles@manchester.ac.uk

Abstract. This chapter reviews methods for the assessment and comparison of Pareto set approximations. Existing set quality measures from the literature are critically evaluated based on a number of orthogonal criteria, including invariance to scaling, monotonicity and computational effort. Statistical aspects of quality assessment are also considered in the chapter. Three main methods for the statistical treatment of Pareto set approximations deriving from stochastic generating methods are reviewed. The *dominance ranking method* is a generalization to partially-ordered sets of a standard non-parametric statistical test, allowing collections of Pareto set approximations from two or more stochastic optimizers to be directly compared statistically. The *quality indicator method* — the dominant method in the literature — maps each Pareto set approximation to a number, and performs statistics on the resulting distribution(s) of numbers. The *attainment function method* estimates the probability of attaining each goal in the objective space, and looks for significant differences between these probability density functions for different optimizers. All three methods are valid approaches to quality assessment, but give different information. We explain the scope and drawbacks of each approach and also consider some more advanced topics, including multiple testing issues, and using combinations of indicators. The chapter should be of interest to anyone concerned with generating and analysing Pareto set approximations.

14.1 Introduction

In many application domains, it is useful to approximate the set of Pareto-optimal solutions, cf. (Ehrgott and Gandibleux, 2000; Deb, 2001; Coello Coello *et al.*, 2002). To this end, various approaches have been proposed ranging from exact methods to randomized search algorithms such as evolutionary algorithms, simulated annealing, and tabu search (see Chapters 2 and 3).

Reviewed by: Günter Rudolph, University of Dortmund, Germany
Serpil Sayin, Koç University, Turkey
Kalyanmoy Deb, Indian Institute of Technology Kanpur, India

J. Branke et al. (Eds.): Multiobjective Optimization, LNCS 5252, pp. 373–404, 2008.

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 may be 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.

This chapter addresses both quality and stochasticity. Sections 2–5 are devoted to the issue of set quality measures; they define properties of such measures and discuss selected measures in the light of these properties. The question of how to statistically assess multiple sets generated by a stochastic multiobjective optimizer is dealt with in Sections 6–8. Both aspects are summarized in Section 9.

The chapter will be of interest to anyone concerned with generating methods of any type. Those who are interested in a preference based set of solutions should find this paper useful as well.

14.2 Quantifying Quality General Considerations

14.2.1 Pareto Set Approximations

Assume a general optimization problem $(X, Z, \mathbf{f}, rel)$ where X denotes the decision space, $Z = \mathbf{R}^k$ is the objective space, $\mathbf{f} = (f_1, f_2, \ldots, f_k)$ is the vector of objective functions, and *rel* represents a binary relation over Z that defines a partial order of the objective space, which in turn induces a preorder of the decision space.[1] In the presence of a single objective function ($k = 1$), the standard relation 'less than or equal' is generally used to define the corresponding minimization problem $(X, \mathbf{R}, (f_1), \leq)$. In the case of multiple objective functions, i.e., $k > 1$, usually the relation $\preceq$ with $\mathbf{z}^1 \preceq \mathbf{z}^2 \Leftrightarrow \forall i \in \{1, \ldots, k\} : \mathbf{z}_i^1 \leq \mathbf{z}_i^2$ is taken; it represents a natural extension of $\leq$ to $\mathbf{R}^k$ and is also known as *weak Pareto dominance*. The associated strict order $\prec$ with $\mathbf{z}^1 \prec \mathbf{z}^2 \Leftrightarrow \mathbf{z}^1 \preceq \mathbf{z}^2 \wedge \neg\, \mathbf{z}^2 \preceq \mathbf{z}^1$ is often denoted as *Pareto dominance*, and instead of $\mathbf{z}^1 \prec \mathbf{z}^2$ one also says $\mathbf{z}^1$ *dominates* $\mathbf{z}^2$. Using this terminology, the

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

Pareto-optimal set comprises the set of decision vectors not dominated by any other element in the feasible set $S \subseteq X$.

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 goal is to find a set of mutually incomparable solutions (for any two decision vectors $\mathbf{x}^1, \mathbf{x}^2$, 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.

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

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

If rel is $\preceq$, then rel' represents the natural extension of weak Pareto dominance to Pareto front approximations. In the following, we will use the symbols $\preceq$ and $\prec$ as for dominance 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.

14.2.2 Outperformance and Quality Indicators

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. Sections 2–5 of this chapter focus on the quality aspect and address the issue of how to compare two (or several) Pareto set approximations. 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 Pareto set approximation is associated; the issue of stochasticity will be treated in later sections.

As discussed above, optimization is about searching in an ordered set. The partial order rel for an optimization problem $(X, Z, \mathbf{f}, rel)$ defines a preference structure on the decision space: a solution $\mathbf{x}^1$ is preferable to a solution $\mathbf{x}^2$ iff $\mathbf{f}(\mathbf{x}^1) \; rel \; \mathbf{f}(\mathbf{x}^2)$ and not $\mathbf{f}(\mathbf{x}^2) \; rel \; \mathbf{f}(\mathbf{x}^1)$. This preference structure is the basis on which the optimization process is performed. For the corresponding set problem $(\Psi, \Omega, \mathbf{f}', rel')$, this means that the most natural way to compare two Pareto set approximations A and B generated by two different multiobjective

optimizers is to use the underlying preference structure rel'. In the context of weak Pareto dominance, there can be four situations: (i) A is better than B ($A \preceq B \ \wedge \ B \not\preceq A$), (ii) B is better than A ($A \not\preceq B \ \wedge \ B \preceq A$), (iii) A and B are incomparable ($A \not\preceq B \ \wedge \ B \not\preceq A$), or (iv) A and B are indifferent ($A \preceq B \ \wedge \ B \preceq A$), where 'better' means the first set weakly dominates the second, but the second does not weakly dominate the first. 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 Pareto set approximation 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. 14.1. This is crucial for the search process itself, and almost all algorithms for approximating the Pareto set make use of additional preference information, e.g., in terms of diversity measures.

For this purpose, quantitative set quality measures have been introduced. We will use the term *quality indicator* in the following:

> A (unary) quality indicator is a function $I : \Psi \rightarrow \mathbf{R}$ that assigns each Pareto set approximation a real number.

In combination with the $\leq$ or $\geq$ relation on $\mathbf{R}$, 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, i.e., two Pareto set approximations, by comparing the corresponding indicator values.

Example 1. Let A be an arbitrary Pareto set approximation and consider the subspace Z' of the objective space $Z = \mathbf{R}^k$ that is, roughly speaking, weakly dominated by A. That means any objective vector in Z' is weakly dominated by at least one objective vector in $\mathbf{f}'(A)$.

The hypervolume indicator I_H (Zitzler and Thiele, 1999) gives the hypervolume of Z' (see Fig. 14.2). 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.

Considering again Fig. 14.1, it can be seen that the hypervolume indicator reveals differences in quality that cannot be detected by the dominance relation. 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 $I_H(B) = 174$.[2] This advantage, though, comes at the expense of generality, since every quality indicator represents certain assumptions about the decision maker's preferences. Whenever $I_H(A) > I_H(B)$,

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

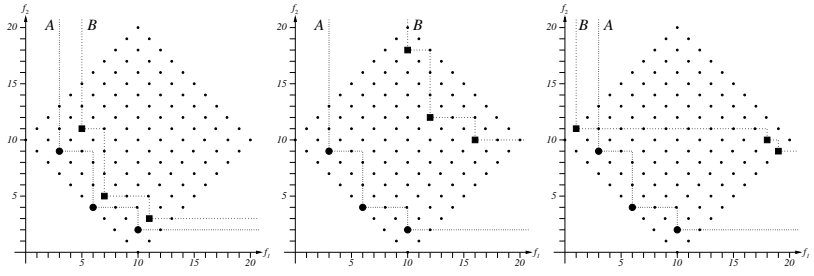

Fig. 14.1. Three examples to illustrate the limitations of statements purely based on weak Pareto dominance. In both the figures on the left, the Pareto set approximation A dominates the Pareto set approximation B, but in one case the two sets are much closer together than in the other case. On the right, A and B are incomparable, but in most situations A will be more useful to the decision maker than B. The background dots represent the image of the feasible set S in the objective space $\mathbf{R}^2$ for a discrete problem.

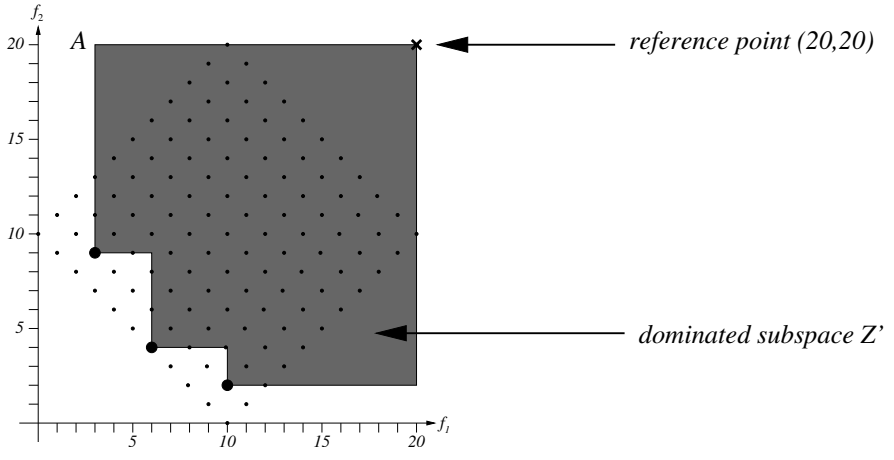

Fig. 14.2. Illustration of the hypervolume indicator. In this example, approximation set A is assigned the indicator value $I_H(A) = 277$; the objective vector $(20, 20)$ is taken as the reference point.

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 Pareto set approximation" needs

to be qualified by adding "under the assumption that I_H reflects the decision maker's preferences".

Finally, note that the following discussion focuses on unary quality indicators, although an indicator can take in principle an arbitrary number of Pareto set approximations as arguments. Several quality indicators have been proposed that assign real numbers to pairs of Pareto set approximations, which are denoted as binary quality indicators (see Hansen and Jaszkiewicz, 1998; Knowles and Corne, 2002; 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.

14.3 Properties of Unary Quality Indicators

Quality indicators serve different goals: they may be used for comparing algorithms, but also during the optimization process as guidance for the search or as stopping criterion. In principle, one may consider any function from Ω to $\mathbf{R}$ as an indicator, but clearly there are certain properties that need to be fulfilled in order to make the indicator useful. These properties may vary depending on the purpose: for instance, when comparing several algorithms on a benchmark problem one may assume that the Pareto-optimal set is known, while such information is clearly not available in a real-world scenario. In the following, we will consider four main criteria:

Monotonicity: An indicator I is said to be *monotonic* iff for any Pareto set approximation that is compared to another Pareto set approximation holds: at least as good in terms of the dominance relation implies at least as good in terms of the indicator values. Formally, this can be expressed as follows:

$$\forall A, B \in \Psi : A \preceq B \Rightarrow I(A) \geq I(B)$$

where $\preceq$ stands for the underlying dominance relation, here weak Pareto dominance.

Monotonicity guarantees that an indicator does not contradict the partial order of Ω that is imposed by the weak Pareto dominance relation, i.e., consistency with the inherent preference structure of the optimization problem under consideration is maintained. However, it does not guarantee a unique optimum with respect to the indicator values; in other words, a Pareto set approximation that has the same indicator value as the Pareto-optimal set not necessarily contains only Pareto-optimal solutions. To this end, a stronger condition is needed which leads to the property of *strict monotonicity*:

$$\forall A, B \in \Psi : A \prec B \Rightarrow I(A) > I(B)$$

Currently, the hypervolume indicator is the only strictly monotonic unary indicator known, (see Zitzler *et al.*, 2007).

Scaling invariance: In practice, the objective functions are often subject to scaling, i.e., the objective function values undergo a strictly monotonic transformation. Most common are transformations of the form of $s(f(\mathbf{x})) = (f(\mathbf{x}) - f_l)/(f_u - f_l)$ where f_l and f_u are lower and upper bounds respectively for the objective function values such that each objective vector lies in $[0, 1]^k$. In this context, it may be desirable that an indicator is not affected by any type of scaling which can be stated as follows: an indicator is denoted as *scaling invariant* iff for any strictly monotonic transformation $\mathbf{s} : \mathbf{R}^k \to \mathbf{R}^k$ the indicator values remain unaffected, i.e., for all $A \in \Psi$ the indicator value $I(A)$ is the same independently of whether we consider the problem $(\Psi, \Omega, \mathbf{f}', rel')$ or the scaled problem $(\Psi, \Omega, \mathbf{s} \circ \mathbf{f}', rel')$.[3] Scaling invariant indicators usually only exploit the dominance relation among solutions, but not their absolute objective function values.

Computation effort: A further property that is less easy to formalize addresses the computational resources needed to compute the indicator value for a given Pareto set approximation. We here consider the runtime complexity, depending on the number of solutions in the Pareto set approximation as well as the number of objectives, as a measure to compare indicators. This aspect becomes critical, if an indicator is to be used during the search process; however, even for pure performance assessment there may be limitations for certain indicators, e.g., if the running time is exponential in the number of objectives as with the hypervolume indicator (While, 2005).

Additional problem knowledge: Many indicators are parameterized and require additional information in order to be applied. Some assume the Pareto-optimal set to be known, while others rely on reference objective vectors or reference sets. In most cases, the indicator parameters are both user- and problem-dependent; therefore, it may be desirable to have as few parameters as possible.

There are many properties one may consider, and the interested reader is referred to (Knowles, 2002; Knowles and Corne, 2002) for a more detailed discussion.

[3] Alternatively, one may consider a weaker version of scaling invariance which is based on the order of the indicator values rather than on the absolute values. More precisely, the elements of Ψ would be sorted according to their indicator values; if the order remains the same for any type of scaling, then the indicator under consideration would be called scaling independent.

14.4 Discussion of Selected Unary Quality Indicators

The unary quality indicators that will be discussed in the following represent a selection of popular measures; however, the list of indicators is by no means complete. Furthermore, only deterministic indicators are considered. A summary of the indicators and properties we consider is given in Table 14.1.

14.4.1 Outer Diameter

The *outer diameter* measures the distance between the ideal objective vector and the nadir objective vector of a Pareto set approximation in terms of a specific distance metric. We here define the corresponding indicator I_{OD} as

$$I_{\mathrm{OD}}(A) = \max_{1 \le i \le n} w_i \left((\max_{\mathbf{x} \in A} f_i(\mathbf{x})) - (\min_{\mathbf{x} \in A} f_i(\mathbf{x})) \right)$$

with weights $w_i \in \mathbf{R}^+$. If all weights are set to 1, then the outer diameter simply provides the maximum extent over all dimensions of the objective space.

The outer diameter is neither monotonic nor scaling invariant. However, it is cheap to compute (the runtime is linear in the cardinality of the Pareto set approximation A) and does not require any additional problem knowledge. The paramters w_i can be used to weight the different objectives, but they are as such not problem-specific.

14.4.2 Proportion of Pareto-Optimal Objective Vectors Found

Another measure to consider is the number of Pareto-optimal objective vectors that are weakly dominated by the image of a Pareto set approximation in objective space. The corresponding indicator I_{PF} has been introduced by Ulungu *et al.* (1999) as the *fraction of the Pareto-optimal front* P weakly dominated by a specific set $A \in \Psi$:

$$I_{\mathrm{PF}}(A) = \frac{\{\mathbf{z} \mid \exists \mathbf{x} \in A : \mathbf{f}(\mathbf{x}) \preceq \mathbf{z}\}}{|P|}$$

This measure assumes that the Pareto-optimal set resp. the Pareto-optimal front is known and that the number of optimal objective vectors is finite. The indicator value can be computed in $\mathcal{O}(|P| \cdot |A|)$ time, and they are invariant to scaling. The indicator is monotonic, but not strictly monotonic.

14.4.3 Cardinality

The *cardinality* $I_{\mathrm{C}}(A)$ of a Pareto set approximation A can be considered both in decision space and objective space, (see, e.g. Van Veldhuizen, 1999). In either case, the indicator is not monotonic. However, it is cheap to compute, scaling invariant, and does not require any additional information.

Table 14.1. Summary of selected indicators and some of their properties. See accompanying text for full details

Indicator	Monotonicity	Scaling invariance	Computational effort	Additional problem knowledge needed
Outer Diameter	✗	✗	linear time	none
Proportion of Pareto Optimal Vectors Found	not strictly	invariant	quadratic	all Pareto optima
Cardinality	✗	invariant	linear time	none
Hypervolume	strictly monotonic	✗	exponential in k	needs upper bounding vector
Completeness	not strictly	invariant	anytime as it is based on sampling, but effort grows rapidly with decision space dimension	none
Epsilon Family	not strictly	✗	quadratic	reference set
D Family	not strictly	✗	quadratic	reference set
R Family	not strictly	✗	quadratic	reference set and point
Uniformity Measures	✗	✗	quadratic	varies

14.4.4 Hypervolume Indicator

The *hypervolume indicator* I_{H}, which was introduced in (Zitzler and Thiele, 1998), gives the volume of the portion of the objective space that is weakly dominated by a specific Pareto set approximation. It can be formally defined as

$$I_H^*(A) := \int_{\mathbf{z}_{\mathrm{lower}}}^{\mathbf{z}_{\mathrm{upper}}} \alpha_A(\mathbf{z}) d\mathbf{z}$$

where $\mathbf{z}_{\mathrm{lower}}$ and $\mathbf{z}_{\mathrm{upper}}$ are objective vectors representing lower resp. upper bounds for the portion of the objective space within which the hypervolume is calculated, and where the function α_A is the attainment function (Grunert da Fonseca *et al.*, 2001) for A

$$\alpha_A(\mathbf{z}) := \begin{cases} 1 & \text{if } \exists\, \mathbf{x} \in A : \mathbf{f}(\mathbf{x}) \preceq \mathbf{z} \\ 0 & \text{else} \end{cases}$$

that returns for an objective vector a 1 if and only if it is weakly dominated by A. In practice, the lower bound $\mathbf{z}_{\mathrm{lower}}$ is not required to calculate the hypervolume for a set A. The hypervolume indicator is to be maximized.

The hypervolume indicator is currently the only unary indicator known to be strictly monotonic. This comes at the cost of high computational cost: the best known algorithms for computing the hypervolume have running times which are exponential in the number of objectives (see While, 2005; While *et al.*, 2005, 2006; Fonseca *et al.*, 2006; Beume and Rudolph, 2006). Furthermore, a reference point, an upper bound, needs to be specified; the indicator is sensitive to the choice of this upper bound, i.e. the ordering of Pareto set approximations induced by the indicator is affected by it, so the indicator is not scaling invariant by the above definition. Note: preference information can be incorporated into the hypervolume indicator, so that more emphasis can be placed on certain parts of the Pareto front than others (e.g. the middle, the extremes, etc.), whilst maintaining monotonicity (Zitzler *et al.*, 2007).

14.4.5 Completeness Indicator

The *completeness indicator* I_{CP} was introduced in (Lotov *et al.*, 2002, 2004) and goes back to the concept of completeness as defined by (Kamenev and Kondtrat'ev, 1992; Kamenev, 2001). The indicator gives the probability that a randomly chosen solution from the feasible set S is weakly dominated by a given Pareto set approximation A, i.e.,

$$I_{\mathrm{CP}}(A) = \mathrm{Prob}\left[A \preceq \{\mathcal{U}\}\right] \tag{14.1}$$

where $\mathcal{U}$ is a random variable representing the random choice from S. Provided that $\mathcal{U}$ follows a uniform probability density function, the indicator value $I_{\mathrm{CP}}(A)$ can also be interpreted as the portion of the feasible set that is dominated by A. As such, the completeness indicator is strongly related to the hypervolume indicator; the difference is that the former takes the decision space into account, while the latter considers the objective space only.

Normally, one cannot compute the completeness directly. For this reason, the indicator values can be estimated by drawing samples from the feasible set and computing the completeness for these samples. As shown by Lotov et al. (Lotov *et al.*, 2004), the confidence interval for the true value can be evaluated with any reliability, given sufficiently large samples. Furthermore, there is an extension of this indicator, namely $I^{\epsilon}_{\mathrm{CP}}(A)$, where another dominance relation, e.g., the ϵ-dominance relation, $\preceq_{\epsilon}$, as defined above, is considered which reflects a specific ϵ-neighborhood of a Pareto set approximation in the objective space, see (Lotov *et al.*, 2002, 2004) for details.

The completeness indicator is scaling-invariant as it does not rely on the absolute objective function values. Furthermore, the exact completeness indicator is as the hypervolume indicator strictly monotonic. However, as in practice sampling is necessary to estimate the exact indicator values, the indicator function based on estimates is monotonic (if always the same sample is used to compare two Pareto set approximation), while strict monotonicity cannot be ensured in general. Experimental studies have shown that the indicator estimates are effective only for relatively low-dimensional decision spaces

(not more than a dozen decision variables) and for sufficiently slowly varying objective functions (Lotov *et al.*, 2002, 2004). For a high-dimensional decision space, the Pareto-optimal set cannot be found via random point generation if it has an extremely small volume. For this reason, a generalized completeness estimate for the quality of approximation has been proposed for the case of a large number of variables and rapidly varying functions, see (Berezkin *et al.*, 2006).

14.4.6 Epsilon Indicator Family

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 definition is closely related to the notion of epsilon efficiency (Helbig and Pateva, 1994). The binary multiplicative epsilon indicator, $I_\epsilon(A, B)$, gives the minimum factor ϵ by which objective vector associated with B can be multiplied such that the resulting transformed Pareto front approximation is weakly dominated by the Pareto front approximation represented by A:

$$I_\epsilon(A, B) = \inf_{\epsilon \in \mathbf{R}} \{\forall \mathbf{x}^2 \in B \; \exists \mathbf{x}^1 \in A \; : \; \mathbf{x}^1 \preceq_\epsilon \mathbf{x}^2\}. \tag{14.2}$$

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

$$\mathbf{x}^1 \preceq_\epsilon \mathbf{x}^2 \iff \forall i \in 1..n \; : \; f_i(\mathbf{x}^1) \leq \epsilon \cdot f_i(\mathbf{x}^2) \tag{14.3}$$

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

$$I^1_\epsilon(A) = I_\epsilon(A, R), \tag{14.4}$$

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

$$\mathbf{x}^1 \preceq_{\epsilon+} \mathbf{x}^2 \iff \forall i \in 1..n \; : \; f_i(\mathbf{x}^1) \leq \epsilon + f_i(\mathbf{x}^2). \tag{14.5}$$

Both unary indicators are to be minimized. An indicator value less than or equal to 1 (I^1_ϵ) respectively 0 ($I^1_{\epsilon+}$) implies that A weakly dominates the reference set R.

The unary epsilon indicators are monotonic, but not strictly monotonic. They are sensitive to scaling and require a reference set relatively to which the epsilon value is calculated. For any finite Pareto set approximation A and any finite reference set R, the indicator values are cheap to compute; the runtime complexity is of order $\mathcal{O}(n \cdot |A| \cdot |R|)$.

14.4.7 The D Indicator Family

The *D indicators* are similar to the additive epsilon indicator and measure the average resp. worst case component-wise distance in objective space to

the closest solution in a reference R, as suggested in (Czyzak and Jaskiewicz, 1998). Czyzak and Jaskiewicz (1998) introduced two versions, I_{D1} and I_{D2}; the first considers the average distance regarding the set R:

$$I_{\mathrm{D1}}(A) = \frac{1}{|R|} \sum_{\mathbf{x}^2 \in R} \min_{\mathbf{x}^1 \in A} \max_{1 \le i \le k} \left(0, w_i(f_i(\mathbf{x}^1) - f_i(\mathbf{x}^2))\right)$$

where the w_i are weights associated with the specific objective functions. Alternatively, the worst case distance may be considered:

$$I_{\mathrm{D2}}(A) = \max_{\mathbf{x}^2 \in R} \min_{\mathbf{x}^1 \in A} \max_{1 \le i \le k} \left(0, w_i(f_i(\mathbf{x}^1) - f_i(\mathbf{x}^2))\right)$$

As with the epsilon indicator family, the D indicators are monotonic, but not strictly monotonic, scaling dependent, and require a reference set. The running time complexity is of order $\mathcal{O}(n \cdot |A| \cdot |R|)$.

14.4.8 The R Indicator Family

The *R indicators* proposed in (Hansen and Jaszkiewicz, 1998) can be used to assess and compare Pareto set approximations on the basis of a set of utility functions. Here, a utility function u is defined as a mapping from the set $\mathbf{R}^k$ of k-dimensional objective vectors to the set of real numbers:

$$u : \mathbf{R}^k \mapsto \mathbf{R}.$$

Now, suppose that the decision maker's preferences are given in terms of a parameterized utility function u_λ and a corresponding set Λ of parameters. For instance, u_λ could represent a weighted sum of the objective values, where $\lambda = (\lambda_1, \ldots \lambda_n) \in \Lambda$ 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:[4]

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

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

[4] 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. This is a way of enabling preference information regarding different parts of the Pareto front to be accounted for, e.g. more utility functions can be placed in the middle of the Pareto front in order to emphasise that region. The interested reader is referred to the original paper for further information.

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

With respect to the choice of the parameterized utility function u_λ, there are various possibilities. A first utility function u that can be used in the above is a weighted linear function

$$u_\lambda(\mathbf{z}) = -\sum_{j\in 1..n} \lambda_j |z^*_j - z_j|, \tag{14.6}$$

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

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

$$u_\lambda(\mathbf{z}) = -\max_{j\in 1..n} \lambda_j |z^*_j - z_j|. \tag{14.7}$$

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,

$$u_\lambda(\mathbf{z}) = -\left(\max_{j\in 1..n} \lambda_j |z^*_j - z_j| + \rho \sum_{j\in 1..n} |z^*_j - z_j|\right), \tag{14.8}$$

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 λ with $\forall i \in 1..n : \lambda_n \geq 0 \wedge \sum_{j=1..n} \lambda_j = 1$.

The R indicators are monotonic, but not strictly, scaling dependent and require both a reference set as well as an ideal objective vector. The runtime complexity for computing the indicator values is of order $\mathcal{O}(n \cdot |\Lambda| \cdot |A| \cdot |R|)$.

14.4.9 Uniformity Measures

Various indicators have been proposed that measure how well the solutions of a Pareto set approximations are distributed in the objective space; often, the main focus is on a uniform distribution. To this end, one can consider the standard deviation of nearest neighbor distances, (see, e.g. Schott, 1995) and

(Deb *et al.*, 2002). Further examples can be found in (Knowles, 2002; Knowles and Corne, 2002).

In general, uniformity measures are not monotonic and not scaling invariant. The computation time required to compute the indicator values is usually quadratic in the cardinality of the Pareto set approximation under consideration, i.e., $\mathcal{O}(n \cdot |A|^2)$. Most measures of this class do not require additional information, but some involve certain problem-dependent parameters.

14.5 Indicator Combinations and Binary Indicators

The ideal quality indicator is strictly monotonic, scaling invariant, cheap to compute and does not require any additional information. However, it can be seen from the discussion above that such an ideal indicator does not exist. For instance, all monotonic unary quality indicators require a reference point and/or a reference set. The only strictly monotonic indicator currently known, the hypervolume indicator, is by far the most computationally expensive indicator. An obvious way to circumvent some of these problems is to combine multiple indicators. One has to define how exactly the resulting information is combined, for instance, one may consider a sequence of indicators. Suppose we would like to combine the epsilon indicator and the hypervolume indicator: one may say A is preferable to B if either the epsilon value for A is better or the epsilon values are identical and the hypervolume value for A is better. The resulting indicator combination would be strictly monotonic, but in average much less expensive than the hypervolume computation alone because in many cases the decision can be already made on the basis of the epsilon indicator. Another possibility is the use of binary quality indicators; see (Zitzler *et al.*, 2003) for a detailed discussion. Here, both scaling invariance and strict monotonicity can be achieved at the same time, e.g., with the coverage indicator (Zitzler and Thiele, 1998).

14.6 Stochasticity: General Considerations

So far, we have assumed that each algorithm under consideration always generates the same Pareto set approximation 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 Pareto set approximation may be returned. In this sense, with each randomized algorithm a random variable is associated whose possible values are Pareto Set approximations, i.e., elements of Ψ; 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 Pareto set approximations. Now, comparing two stochastic optimizers basically means comparing the two corresponding samples of Pareto set approximations. 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 samples of Pareto set approximations into samples of real values using quality indicators; then, the resulting samples of indicator values are compared based on standard statistical testing procedures.

Example 2. Consider two hypothetical stochastic multiobjective optimizers and assume that the outcomes of three independent optimization runs are as depicted in Fig. 14.3. 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 Pareto set approximations 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 $\mathbf{z}$ in the objective space, there is a certain probability p that the resulting Pareto set approximation contains an element $\mathbf{x}$ such $\mathbf{f}(\mathbf{x}) \preceq \mathbf{z}$. We say p is the probability that $\mathbf{z}$ is *attained* by the optimizer. The *attainment function* gives for each objective vector $\mathbf{z}$ in the objective space the probability that $\mathbf{z}$ 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. The attainment function is a first order moment measure, meaning that it estimates the probability that $\mathbf{z}$ is attained in one optimization run of the considered algorithm *independently* of attaining any other $\mathbf{z}$. For the consideration of higher order attainment functions, Grunert da Fonseca *et al.* (2001) have developed corresponding statistical testing procedures.

Example 3. Consider Fig. 14.3. For the scenario on the right, the three Pareto front approximations cut the objective space into four regions: the upper right

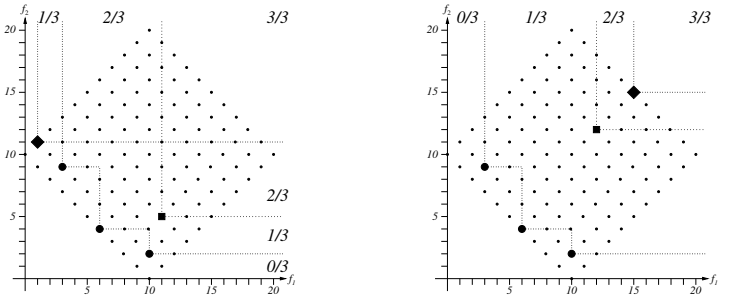

Fig. 14.3. Hypothetical outcomes of three runs for two different stochastic optimizers (left and right). The numbers in the figures give the relative frequencies according to which the distinct regions in the objective space were attained.

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 to statistical analysis of approximation sets 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. Basically, for each Pareto set approximation generated by one optimizer it is computed by how many Pareto set approximations produced by another optimizer it is dominated. As a result, one obtains, for each algorithm, a set of ranks and can statistically verify whether the rank distributions for two algorithms differ significantly or not. We call this method, dominance ranking.

Example 4. To compare the outcomes of the two hypothetical optimizers depicted in Fig. 14.3, we check for each pair consisting of one Pareto set approximation of the first optimizer and one Pareto set approximation of the second optimizer whether either is better or not. For the Pareto front approximation represented by the diamond on the left hand side, none of the three Pareto front approximations on the right is better and therefore it is assigned the lowest rank 0. The Pareto front approximation associated with the diamond on the right hand side is worse than all three Pareto front approximations on the left and accordingly its rank is 3. Overall, the resulting rank distributions are $(0, 0, 1)$ for the algorithm on the left hand side and $(0, 2, 3)$ for the algorithm on the right hand side. A special statistical test can be used to determine whether the two rank distributions are significantly different.

14.7 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—specifically, 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 14.8.

14.7.1 Dominance Ranking

Principles and Procedure

Suppose that we wish to compare the quality of Pareto set approximations generated by two stochastic multiobjective optimizers, where $A_1, A_2, \ldots, A_r$ represent the approximations generated by the first optimizer in r runs, while $B_1, B_2, \ldots, B_s$ denote the approximations generated by the second optimizer in s runs. Using the preference structure of the underlying set problem $(\Psi, \Omega, \mathbf{f}', rel')$, one can now compare all A_i with all B_j and thereby assign a figure of merit or a rank to each Pareto set approximation, similarly to the way that dominance-based fitness assignment works in multiobjective evolutionary algorithms. In principle, there are several ways to assign each Pareto set approximation a rank on the basis of a dominance relation, e.g., by counting the number of sets by which a specific approximation is dominated (Fonseca and Fleming, 1993) or by performing a nondominated sorting on the Pareto set approximations under considerations. Here, the former approach in combination with the extended weak Pareto dominance $\preceq$, cf. Section 14.2 on Page 374, is preferred as it produces a finer-grained ranking, with fewer ties, than nondominated sorting:

$$rank(A_i) = |\{B | B \in \{B_1, \ldots, B_s\} \ \wedge B \prec A_i\}|. \tag{14.9}$$

The ranks for $B_1, \ldots, B_s$ are determined analogously. The lower the rank, the better the corresponding Pareto set approximation with respect to the entire collection of sets associated with the other optimizer.

The result of this procedure is that each A_i and B_j is associated with a figure of merit. Accordingly, the samples of Pareto set approximations associated with each algorithm have been transformed into samples of ranks: $(rank(A_1), rank(A_2), \ldots, rank(A_r))$ and $(rank(B_1), rank(B_2), \ldots, rank(B_s))$.

An example performance comparison study using the dominance ranking procedure can be found in (Knowles *et al.*, 2006).

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 underlying preference structure. It may be interesting and worthwhile to use either quality indicators or the attainment function to characterize further the differences in the distributions of the Pareto set approximations, 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.

14.7.2 Quality Indicators

Principles and Procedures

As stated earlier, a *unary* quality indicator I is defined as a mapping from Ψ to the set of real numbers. The order that I establishes on Ω is supposed to represent the quality of the Pareto set approximations. Thus, given a pair of approximations, 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*.

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 14.8. In contrast to the dominance ranking approach, it is also possible to make quantitative statements about the differences in quality, even for incomparable Pareto set approximations. 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.

14.7.3 Empirical Attainment Function

Principles and Procedures

The central concept in this approach is the notion of an *attainment function*. Since the multiobjective optimizers that we consider may be stochastic, the result of running the optimizer can be described by a distribution. Because the optimizer returns a Pareto set approximation in any given run, the distribution can be described in the objective space by a random *set* $\mathcal{Z}$ of random objective vectors $\breve{\mathbf{z}}^j$, with the cardinality of the set, m, also random, as follows:

$$\mathcal{Z} = \{\breve{\mathbf{z}}^j \in \mathbf{R}^k, j = 1, \ldots, m\}, \tag{14.10}$$

where k is the number of objectives of the problem. 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 Pareto front approximation returned by the optimizer. It is defined by the function $\alpha_{\mathcal{Z}}(.) : \mathbf{R}^n \mapsto [0, 1]$ with

$$\alpha_{\mathcal{Z}}(z) = P(\breve{\mathbf{z}}^1 \preceq \mathbf{z} \vee \breve{\mathbf{z}}^2 \preceq \mathbf{z} \vee \ldots \vee \breve{\mathbf{z}}^m \preceq \mathbf{z}) \tag{14.11}$$

$$= P(\mathcal{Z} \preceq \{\mathbf{z}\}) \tag{14.12}$$

$$= P(\text{that the optimizer attains goal } \mathbf{z} \text{ in a single run}), \tag{14.13}$$

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 Pareto set approximation 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

$$\alpha_r(\mathbf{z}) = \frac{1}{r} \sum_{i=1}^{r} \boldsymbol{I}(\mathbf{f}'(A_i) \preceq \{\mathbf{z}\}), \tag{14.14}$$

where A_i is the ith Pareto set approximation (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 Pareto front approximation, 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 14.8.4. 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%-*attainment set*:

A Pareto set approximation A is called the $k\%$-*attainment set* of an EAF $\alpha_r(\mathbf{z})$, iff the corresponding Pareto front approximation weakly dominates exactly those objective vectors that have been attained in at least k percent of the r runs. Formally,

$$\forall \mathbf{z} \in Z : \alpha_r(\mathbf{z}) \geq k/100 \Leftrightarrow \mathbf{f}'(A) \preceq \{\mathbf{z}\} \tag{14.15}$$

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

An attainment surface of a given Pareto set approximation A is the union of all tightest goals that are known to be attainable as a result of A. Formally, this is the set $\{\mathbf{z} \in \mathbf{R}^k \mid \mathbf{f}'(A) \preceq \mathbf{z} \wedge \nexists \mathbf{z}^2 \in \mathbf{R}^k : \mathbf{f}'(A) \preceq \mathbf{z}^2 \prec \mathbf{z}\}$.

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 5. Suppose a stochastic multiobjective optimizer returns the Pareto front approximations depicted in Fig. 14.4 for five different runs on a biobjective optimization problem. The corresponding attainment surfaces are shown in Fig. 14.5; they summarize the underlying empirical attainment function.

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 $\mathbf{R}$. 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 14.8). However, the approach is computationally

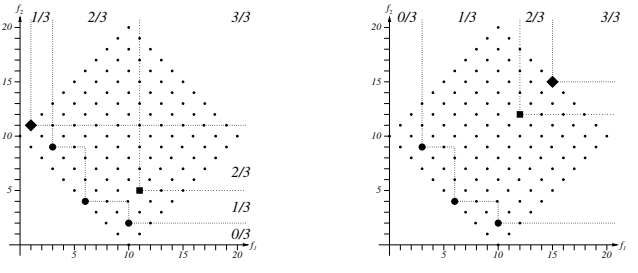

Fig. 14.4. A plot showing five Pareto front approximations. The visual evaluation is difficult, although there are only a few points per set, and few sets.

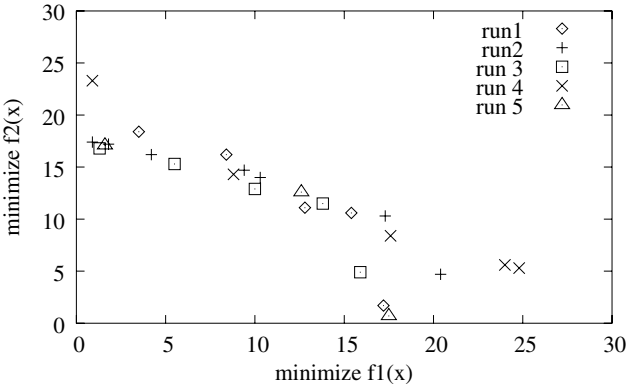

Fig. 14.5. Attainment surface plots for the Pareto fron approximations in Figure 14.4. The first (solid) line represents the 20%-attainment surface, the second line the 40%-attainment surface, and so forth; the fifth line stands for the 100%-attainment surface.

expensive and therefore only applicable in the case of a few objective functions. 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.

14.8 Statistical Testing

14.8.1 Fundamentals

The previous section has described three different transformations that can be applied to a sample of Pareto set approximations 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.

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 Pareto set approximation *distribution* is better than another one's.[5] However, we cannot determine this fact definitively because we only have access to finite-sized *samples* of Pareto set approximations. 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.

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.

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

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 selected 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.

14.8.2 Comparing Samples of Dominance Ranks

Dominance ranking converts the samples of approximation sets from two or more optimizers into a sample of dominance ranks. A test statistic is computed from these ranks by summing over the ranks in each of the two samples and taking the difference of these sums. In order to determine whether the value of the test statistic is significant, a permutation test must be used. The standard Mann-Whitney rank sum test and tables (Conover, 1999) cannot be used here because the rank distributions are affected by the fact that the sets are partially ordered (rather than totally ordered numbers). Thus, to compute the null distribution, the assignment of the Pareto set approximations to the optimizers must be permuted. Basically, the set $\{A_1, A_2, \ldots, A_r, B_1, B_2, \ldots, B_s\}$ is partitioned into one set of r approximations and another set of s approximations; for each partitioning the difference between the rank sums can be determined, finally yielding a distribution of rank sum differences. Details for this statistical testing procedure are given in (Knowles *et al.*, 2006).

14.8.3 Comparing Sample Indicators Values

The use of a quality indicator reduces the dimension of a Pareto set approximation 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. Here, the Mann-Whitney rank sum test or Fisher's permutation test can be used (Conover, 1999); the Kruskal-Wallis test may be more appropriate if multiple (more than two) algorithms are to be compared.

In the case that a combination of multiple quality indicators is considered (see Page 386), slightly different preferences are assessed by each of the indicators and this may help to build up a better picture of the overall quality of the Pareto set approximations. On the other hand, using several indicators does bring into play multiple testing issues if the distributions from different indicators are being tested independently, cf. Section 14.8.5.

14.8.4 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). 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 and the maximum distance between the EAFs that needs to be exceeded. Thus the KS-like test can be run for different selected α values to compute these different distances. Then, the actual measured distances between the EAFs at every $\mathbf{z}$ can be converted to a significance level.

An example of such a pair of plots is shown in Figure 14.6. This kind of plot has been used to good effect in (López-Ibáñez *et al.*, 2006). Note also that Fonseca *et al.* (2005) have devised plots that can indicate second-order information, i.e. the probability of an optimizer attaining pairs of goals simultaneously.

14.8.5 Advanced Topics

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

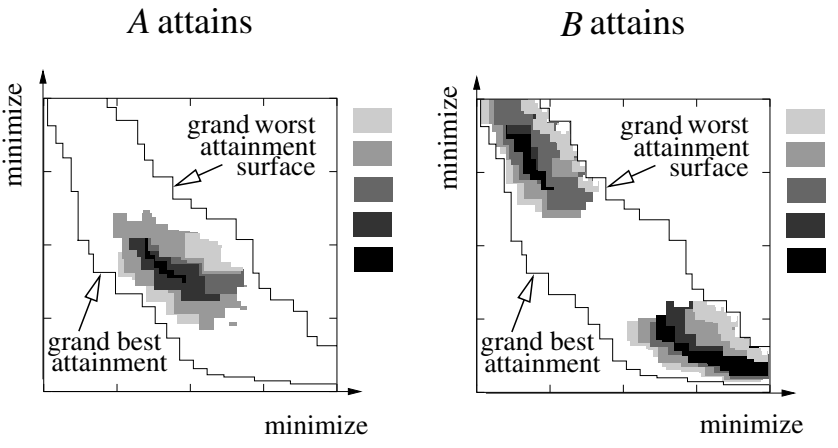

Fig. 14.6. Individual differences between the probabilities of attaining different goals on a two-objective minimization problem with optimizer O_1 and optimizer O_2, shown using a greyscale plot. The grand best and worst attainment surfaces (the same in both plots) indicate the borders beyond which the goals are never attained or always attained, computed from the *combined* collection of Pareto set approximations. Differences in the frequency with which certain goals are met by the respective algorithms O_1 and O_2 are then represented in the region between these two surfaces. In the left plot, darker regions indicate goals that are attained more frequently by O_1 than by O_2. 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.

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 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.

Multiple Testing

Multiple testing (Benjamini and Hochberg, 1995; Bland and Altman, 1995; Miller, 1981; Perneger, 1998; Westfall and Young, 1993) occurs when one wishes to consider several statistical hypotheses (or comparisons) simultaneously. When considering multiple tests, the significance of each single result needs to be adjusted to account for the fact that, as more tests are considered, it becomes more and more likely that some (unspecified) result will give an extreme value, resulting in a rejection of the null hypothesis for that test.

For example, imagine we carry out a study consisting of twenty different hypothesis tests, and assume that we reject the null hypothesis of each test if the p-value is 0.05 or less. Now, the chance that at least one of the inferences will be a type-1 error (i.e. the null hypothesis is wrongly rejected) is $1 - (0.95^{20}) \simeq 64\%$, when assuming that the null hypothesis was true in every case. In other words, more often than not, we wrongly claim a significant result (on at least one test). This 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 be utterly misleading to a reader.

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 just two algorithms, but we wish to make multiple statistical tests of their performance, e.g., considering 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 five 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 a pair of algorithms by carefully choosing which tests to apply before collecting the data. Collect independent data for each test to be carried out.

v). 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) 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; the underlying strategy is to perform a test only if 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. However, while following (iv) 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 should then use approach (v).

The simplest and most conservative, i.e., weakest approach for correcting the p-values is the Bonferroni correction (Bland and Altman, 1995). 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

$$\alpha_s = \frac{\alpha}{n} \tag{14.16}$$

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

$$P(T_i \text{ passes } \mid H_0) \leq \frac{\alpha}{n} \text{ for } 1 \leq i \leq n, \tag{14.17}$$

then

$$P(\text{some } T_i \text{ passes } \mid H_0) \leq \alpha. \tag{14.18}$$

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

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, one can 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, the labels of all 600 samples can be randomly permuted, and the worst indicator value from those with a label in $1, \ldots, 10$ are determined. By sampling this statistic 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 quite general and it can be tailored to answer many questions related to worst-case or best-case performance.

14.9 Summary

This chapter deals with the issue of assessing and comparing the quality of Pareto set approximations. Two current principal approaches, the quality indicator method and the attainment function method, are discussed, and, in addition, a third approach, the dominance-ranking technique, is presented.[6]

As discussed, there is no 'best' quality assessment technique with respect to both quality measures and statistical analysis. Instead, it appears to be reasonable to use the complementary strengths of the three general approaches. As a first step in a comparison, it can be checked whether the considered optimizers exhibit significant differences using the dominance-ranking approach, because such an analysis allows the strongest type of statements. 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 considered indicators. 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 hints on *where* the outcomes of two algorithms differ.

[6] Implementations for selected quality indicators as well as statistical testing procedures can be downloaded at *http://www.tik.ee.ethz.ch/sop/pisa/* under the heading 'performance assessment'.

We noted when discussing quality indicators that, as well as their traditional use to assess optimization outcomes, they can also be used within optimizers, to guide the generating process (Beume *et al.*, 2007; Fleischer, 2003; Smith *et al.*, 2008; Wagner *et al.*, 2007; Zitzler and Künzli, 2004). Optimizers that seek to maximize a quality indicator directly are effectively conducting the search in the space of approximation sets, rather than in the space of solutions or points. This seems a logical and attractive approach when attempting to generate a Pareto front approximation, because ultimately the outcome will be assessed using a quality indicator (usually). However, although such approaches are improving, some of them still rely on approximation of the set-based indicator function, or they do not rely solely on the indicator, but make use of heuristics concerning individuals (point/solutions) (e.g., an individual's nondominated rank) as well. A recent study even compared set-based selection with individual-based selection, and found the latter to be generally preferable.

Quality indicators for assessing Pareto front approximations are sometimes used without explicitly stating what the DM preferences are. Really, the indicator(s) used should reflect any information one has about the DM preferences, so that approximation sets are assessed appropriately. The work of Hansen and Jaszkiewicz (1998) defined some quality indicators in terms of sets of utility functions, a framework that easily allows for DM preferences to be incorporated into assessment. A similar approach was recently proposed by Zitzler *et al.* (2007) for the hypervolume. Both of these indicator families can be used to incoporate preferences within generating methods (potentially in an interactive fashion).

Finally, note that there are several further issues that have not been treated in this chapter, e.g., binary quality indicators; indicators taking the decision vectors into account; computation of indicators on parallel or distributed architectures. Many of these issues represent current research directions which will probably lead to modified or additional performance assessment methods in the near future.

Acknowledgements

Sections 14.1 to 14.5 summarize the results of the discussion of the working group on set quality measures during the Dagstuhl seminar on evolutionary multiobjective optimization 2006. The working group consisted of the following persons: Jörg Fliege, Carlos M. Fonseca, Christian Igel, Andrzej Jaszkiewicz, Joshua D. Knowles, Alexander Lotov, Serpil Sayin, Lothar Thiele, Andrzej Wierzbicki, and Eckart Zitzler. The authors would also like to thank Carlos M. Fonseca for valuable discussion and for providing the EAF tools.

References

Benjamini, Y., Hochberg, Y.: Controlling the false discovery rate: a practical and powerful approach to multiple testing. Journal of the Royal Statistical Society, Series B (Methodological) 57, 125–133 (1995)

Berezkin, V.E., Kamenev, G.K., Lotov, A.V.: Hybrid adaptive methods for approximating a nonconvex multidimensional pareto frontier. Computational Mathematics and Mathematical Physics 46(11), 1918–1931 (2006)

Beume, N., Rudolph, G.: Faster S-Metric Calculation by Considering Dominated Hypervolume as Klee's Measure Problem. In: Proceedings of the Second IASTED Conference on Computational Intelligence, pp. 231–236. ACTA Press, Anaheim (2006)

Beume, N., Naujoks, B., Emmerich, M.: SMS-EMOA: Multiobjective selection based on dominated hypervolume. European Journal on Operational Research 181, 1653–1669 (2007)

Bland, J.M., Altman, D.G.: Multiple significance tests: the bonferroni method. British Medical Journal 310, 170 (1995)

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

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

Conover, W.J.: Practical Nonparametric Statistics, 3rd edn. John Wiley and Sons, New York (1999)

Czyzak, P., Jaskiewicz, A.: Pareto simulated annealing—a metaheuristic for multiobjective combinatorial optimization. Multi-Criteria Decision Analysis 7, 34–47 (1998)

Deb, K.: Multi-objective optimization using evolutionary algorithms. Wiley, Chichester (2001)

Deb, K., Pratap, A., Agrawal, S., Meyarivan, T.: A fast and elitist multi-objective genetic algorithm: NSGA-II. IEEE Transactions on Evolutionary Computation 6(2), 181–197 (2002)

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

Ehrgott, M., Gandibleux, X.: A Survey and Annotated Bibliography of Multiobjective Combinatorial Optimization. OR Spektrum 22, 425–460 (2000)

Fleischer, M.: The Measure of Pareto Optima. In: Fonseca, C.M., Fleming, P.J., Zitzler, E., Deb, K., Thiele, L. (eds.) EMO 2003. LNCS, vol. 2632, pp. 519–533. Springer, Heidelberg (2003)

Fonseca, C.M., Fleming, P.J.: Genetic Algorithms for Multiobjective Optimization: Formulation, Discussion and Generalization. In: Forrest, S. (ed.) Proceedings of the Fifth International Conference on Genetic Algorithms, pp. 416–423. Morgan Kaufmann, San Mateo (1993)

Fonseca, C.M., Grunert da Fonseca, V., Paquete, L.: Exploring the performance of stochastic multiobjective optimisers with the second-order attainment function. In: Coello Coello, C.A., Hernández Aguirre, A., Zitzler, E. (eds.) EMO 2005. LNCS, vol. 3410, pp. 250–264. Springer, Heidelberg (2005)

Fonseca, C.M., Paquete, L., López-Ibáñez, M.: An Improved Dimension-Sweep Algorithm for the Hypervolume Indicator. In: Congress on Evolutionary Computation (CEC 2006), Sheraton Vancouver Wall Centre Hotel, Vancouver, BC Canada, pp. 1157–1163. IEEE Computer Society Press, Los Alamitos (2006)

Grunert da Fonseca, V., Fonseca, C.M., Hall, A.O.: Inferential Performance Assessment of Stochastic Optimisers and the Attainment Function. In: Zitzler, E., Deb, K., Thiele, L., Coello Coello, C.A., Corne, D.W. (eds.) EMO 2001. LNCS, vol. 1993, pp. 213–225. Springer, Heidelberg (2001)

Hansen, M.P., Jaszkiewicz, A.: Evaluating the quality of approximations of the nondominated set. Technical report, Institute of Mathematical Modeling, Technical University of Denmark. IMM Technical Report IMM-REP-1998-7 (1998)

Helbig, S., Pateva, D.: On several concepts for ϵ-efficiency. OR Spektrum 16(3), 179–186 (1994)

Kamenev, G., Kondratíev, D.: Method for the exploration of non-closed nonlinear models (in Russian). Matematicheskoe Modelirovanie 4(3), 105–118 (1992)

Kamenev, G.K.: Approximation of completely bounded sets by the deep holes method. Computational Mathematics And Mathematical Physics 41, 1667–1676 (2001)

Knowles, J.: 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 (2005)

Knowles, J., Corne, D.: On Metrics for Comparing Non-Dominated Sets. In: Congress on Evolutionary Computation (CEC 2002), pp. 711–716. IEEE Press, Piscataway (2002)

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

Knowles, J.D.: Local-Search and Hybrid Evolutionary Algorithms for Pareto Optimization. Ph.D. thesis, University of Reading (2002)

López-Ibáñez, M., Paquete, L., Stützle, T.: Hybrid population-based algorithms for the bi-objective quadratic assignment problem. Journal of Mathematical Modelling and Algorithms 5(1), 111–137 (2006)

Lotov, A.V., Kamenev, G.K., Berezkin, V.E.: Approximation and Visualization of Pareto-Efficient Frontier for Nonconvex Multiobjective Problems. Doklady Mathematics 66(2), 260–262 (2002)

Lotov, A.V., Bushenkov, V.A., Kamenev, G.K.: Interactive Decision Maps. Approximation and Visualization of Pareto Frontier. Kluwer Academic Publishers, Boston (2004)

Miller, R.G.: Simultaneous Statistical Inference, 2nd edn. Springer, New York (1981)

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

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

Shaw, K.J., Nortcliffe, A.L., Thompson, M., Love, J., Fonseca, C.M., Fleming, P.J.: 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 Computer Society Press, Los Alamitos (1999)

Smith, K.I., Everson, R.M., Fieldsend, J.E., Murphy, C., Misra, R.: Dominance-based multiobjective simulated annealing. IEEE Transactions on Evolutionary Computation. In press (2008)

Ulungu, E.L., Teghem, J., Fortemps, P.H., Tuyttens, D.: Mosa method: A tool for solving multiobjective combinatorial optimization problems. Journal of Multi-Criteria Decision Analysis 8(4), 221–236 (1999)

Van Veldhuizen, D.A.: Multiobjective Evolutionary Algorithms: Classifications, Analyses, and New Innovations. Ph.D. thesis, Graduate School of Engineering, Air Force Institute of Technology, Air University (1999)

Wagner, T., Beume, N., Naujoks, B.: Pareto-, Aggregation-, and Indicator-Based Methods in Many-Objective Optimization. In: Obayashi, S., Deb, K., Poloni, C., Hiroyasu, T., Murata, T. (eds.) EMO 2007. LNCS, vol. 4403, pp. 742–756. Springer, Heidelberg (2007), extended version published as internal report of Sonderforschungsbereich 531 Computational Intelligence CI-217/06, Universität Dortmund (September 2006).

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

While, L.: A New Analysis of the LebMeasure Algorithm for Calculating Hypervolume. In: Coello Coello, C.A., Hernández Aguirre, A., Zitzler, E. (eds.) EMO 2005. LNCS, vol. 3410, pp. 326–340. Springer, Heidelberg (2005)

While, L., Bradstreet, L., Barone, L., Hingston, P.: Heuristics for Optimising the Calculation of Hypervolume for Multi-objective Optimisation Problems. In: Congress on Evolutionary Computation (CEC 2005), IEEE Service Center, Edinburgh, Scotland, pp. 2225–2232. IEEE Computer Society Press, Los Alamitos (2005)

While, L., Hingston, P., Barone, L., Huband, S.: A Faster Algorithm for Calculating Hypervolume. IEEE Transactions on Evolutionary Computation 10(1), 29–38 (2006)

Zitzler, E., Künzli, S.: Indicator-Based Selection in Multiobjective Search. In: Yao, X., Burke, E.K., Lozano, J.A., Smith, J., Merelo-Guervós, J.J., Bullinaria, J.A., Rowe, J.E., Tiňo, P., Kabán, A., Schwefel, H.-P. (eds.) PPSN 2004. LNCS, vol. 3242, pp. 832–842. Springer, Heidelberg (2004)

Zitzler, E., Thiele, L.: Multiobjective Optimization Using Evolutionary Algorithms - A Comparative Case Study. In: Eiben, A.E., Bäck, T., Schoenauer, M., Schwefel, H.-P. (eds.) PPSN 1998. LNCS, vol. 1498, pp. 292–301. Springer, Heidelberg (1998)

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

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

Zitzler, E., Brockhoff, D., Thiele, L.: The Hypervolume Indicator Revisited: On the Design of Pareto-compliant Indicators Via Weighted Integration. In: Obayashi, S., Deb, K., Poloni, C., Hiroyasu, T., Murata, T. (eds.) EMO 2007. LNCS, vol. 4403, pp. 862–876. Springer, Heidelberg (2007)