

A Hypervolume-Based Optimizer for High-Dimensional Objective Spaces

Johannes Bader and Eckart Zitzler

Abstract In the field of evolutionary multiobjective optimization, the hypervolume indicator is the only single set quality measure that is known to be strictly monotonic with regard to Pareto dominance. This property is of high interest and relevance for multiobjective search involving a large number of objective functions. However, the high computational effort required for calculating the indicator values has so far prevented to fully exploit the potential of hypervolume-based multiobjective optimization.

This paper addresses this issue and proposes a fast search algorithm that uses Monte Carlo sampling to approximate the exact hypervolume values. In detail, we present HypE (*Hypervolume Estimation Algorithm for Multiobjective Optimization*), by which the accuracy of the estimates and the available computing resources can be traded off; thereby, not only many-objective problems become feasible with hypervolume-based search, but also the runtime can be flexibly adapted. The experimental results indicate that HypE is highly effective for many-objective problems in comparison to existing multiobjective evolutionary algorithms.

1 Motivation

By far most studies in the field of evolutionary multiobjective optimization (EMO) are concerned with the following set problem: find a set of solutions that as a whole represents a good approximation of the Pareto-optimal set. To this end, the original multiobjective problem consisting of:

- The decision space X
- The objective space $Z = \mathbb{R}^n$
- A vector function $f = (f_1, f_2, \dots, f_n)$ comprising n objective functions $f_i : X \rightarrow \mathbb{R}$, which are without loss of generality to be minimized

J. Bader (✉)

Computer Engineering and Networks Lab, ETH Zurich, 8092 Zurich, Switzerland
e-mail: Johannes.Bader@tik.ee.ethz.ch

- A relation \leq on Z , which induces a preference relation \preceq on X with $a \preceq b \Leftrightarrow f(a) \leq f(b)$ for $a, b \in X$

is usually transformed into a single-objective set problem (Zitzler et al. 2008).

The search space Ψ of the resulting set problem includes all possible Pareto set approximations,¹ i.e., Ψ contains all multisets over X . The preference relation \preceq can be used to define a corresponding set preference relation \preccurlyeq on Ψ where

$$A \preccurlyeq B \Leftrightarrow \forall b \in B \exists a \in A : a \preceq b \quad (1)$$

for all Pareto set approximations $A, B \in \Psi$. In the following, we will assume that weak Pareto dominance is the underlying preference relation, i.e., $a \preceq b \Leftrightarrow f(a) \leq f(b)$ (cf. Zitzler et al. 2008).²

A key question when tackling such a set problem is how to define the optimization criterion. Many multiobjective evolutionary algorithms (MOEAs) implement a combination of Pareto dominance on sets and a diversity measure based on Euclidean distance in the objective space, e.g., NSGA-II (Deb et al. 2000) and SPEA2 (Zitzler et al. 2002). While these methods have been successfully employed in various biobjective optimization scenarios, they appear to have difficulties when the number of objectives increases (Wagner et al. 2007). As a consequence, researchers have tried to develop alternative concepts, and a recent trend is to use set quality measures, also denoted as quality indicators, for search – so far, they have mainly been used for performance assessment. Of particular interest in this context is the hypervolume indicator (Zitzler and Thiele 1998a, 1999) as it is the only quality indicator known to be fully sensitive to Pareto dominance, i.e., whenever a set of solutions dominates another set, it has a higher hypervolume indicator value than the second set. This property is especially desirable when many objective functions are involved.

Several hypervolume-based MOEAs have been proposed meanwhile (e.g., Emmerich et al. 2005; Igel et al. 2007; Brockhoff and Zitzler 2007), but their main drawback is their extreme computational overhead. Although there have been recent studies presenting improved algorithms for hypervolume calculation, currently high-dimensional problems with six or more objectives are infeasible for these MOEAs. Therefore, the question is whether and how fast hypervolume-based search algorithms can be designed that exploit the advantages of the hypervolume indicator and at the same time are scalable with respect to the number of objectives.

¹ Here, a Pareto set approximation may also contain dominated solutions as well as duplicates, in contrast to the notation in Zitzler et al. (2003).

² For reasons of simplicity, we will use the term “ u weakly dominates v ” resp. “ u dominates v ” independently of whether u and v are elements of X , Z , or Ψ . For instance, A weakly dominates b with $A \in \Psi$ and $b \in X$ means $A \preccurlyeq \{b\}$ and a dominates z with $a \in X$ and $z \in Z$ means $f(a) \leq z \wedge z \not\leq f(a)$.

2 Related Work

The hypervolume indicator was originally proposed and employed in Zitzler and Thiele (1998, 1999) to compare quantitatively the outcomes of different MOEAs. In these two first publications, the indicator was denoted as “size of the space covered”, and later also other terms such as “hyperarea metric” (Van Veldhuizen 1999), “S-metric” (Zitzler 1999), “hypervolume indicator” (Zitzler et al. 2003), and “hypervolume measure” (Beume et al. 2007) were used. Besides the names, there are also different definitions available, based on polytopes (Zitzler and Thiele 1999), the attainment function (Zitzler et al. 2007), or the Lebesgue measure (Laumanns et al. 1999; Knowles 2002; Fleischer 2003).

Knowles (2002) and Knowles and Corne (2003) were the first to propose the integration of the hypervolume indicator into the optimization process. In particular, they described a strategy to maintain a separate, bounded archive of nondominated solutions based on the hypervolume indicator. Huband et al. (2003) presented an MOEA which includes a modified SPEA2 environmental selection procedure where a hypervolume-related measure replaces the original density estimation technique. In Zitzler and Künzli (2004), the binary hypervolume indicator was used to compare individuals and to assign corresponding fitness values within a general indicator-based evolutionary algorithm (IBEA). The first MOEA tailored specifically to the hypervolume indicator was described in Emmerich et al. (2005); it combines nondominated sorting with the hypervolume indicator and considers one offspring per generation (steady state). Similar fitness assignment strategies were later adopted in Zitzler et al. (2007) and Igel et al. (2007), and also other search algorithms were proposed where the hypervolume indicator is partially used for search guidance (Nicolini 2005; Mostaghim et al. 2007). Moreover, specific aspects like hypervolume-based environmental selection (Bradstreet et al. 2006), cf. Sect. 3.3, and explicit gradient determination for hypervolume landscapes (Emmerich et al. 2007) have been investigated recently.

The major drawback of the hypervolume indicator is its high computation effort; all known algorithms have a worst-case runtime complexity that is exponential in the number of objectives, more specifically $\mathcal{O}(N^{n-1})$ where N is the number of solutions considered (Knowles 2002; While et al. 2006). A different approach was presented by Fleischer (2003) who mistakenly claimed a polynomial worst-case runtime complexity – While (2005) showed that it is exponential in n as well. Recently, advanced algorithms for hypervolume calculation have been proposed, a dimension-sweep method (Fonseca et al. 2006) with a worst-case runtime complexity of $\mathcal{O}(N^{n-2} \log N)$, and a specialized algorithm related to the Klee measure problem (Beume and Rudolph, 2006) the runtime of which is in the worst case of order $\mathcal{O}(N \log N + N^{n/2})$. Furthermore, Yang and Ding (2007) described an algorithm for which they claim a worst-case runtime complexity of $\mathcal{O}((n/2)^N)$. The fact that there is no exact polynomial algorithm available gave rise to the hypothesis that this problem in general is hard to solve, although the tightest known lower bound is of order $\Omega(N \log N)$ (Beume et al. 2007a). New results substantiate this hypothesis:

Bringmann and Friedrich (2008) have proven that the problem of computing the hypervolume is $\#P$ -complete, i.e., it is expected that no polynomial algorithm exists since this would imply $NP = P$.

The issue of speeding up the hypervolume indicator has been addressed in different ways: by automatically reducing the number of objectives (Brockhoff and Zitzler 2007) and by approximating the indicator values using Monte Carlo simulation (Everson et al. 2002; Bader et al. 2008; Bringmann and Friedrich 2008). Everson et al. (2002) used a basic Monte Carlo technique for performance assessment in order to estimate the values of the binary hypervolume indicator (Wagner et al. 2007); with their approach the error ratio is not polynomially bounded. In contrast, the scheme presented in Bringmann and Friedrich (2008) is a fully polynomial randomized approximation scheme where the error ratio is polynomial in the input size. Another study (Bader et al. 2008) – a precursor study for the present paper – employed Monte Carlo sampling for fast hypervolume-based search. The main idea is to estimate – by means of Monte Carlo simulation – the ranking of the individuals that is induced by the hypervolume indicator and not to determine the exact indicator values. This paper proposes an advanced method called HypE (*Hypervolume Estimation Algorithm for Multiobjective Optimization*) that is based on the same idea, but uses a novel fitness assignment scheme for both mating and environmental selection, that can be effectively approximated.

As we will show in the following, the proposed search algorithm can be easily tuned regarding the available computing resources and the number of objectives involved. Thereby, it opens a new perspective on how to treat many-objective problems, and the presented concepts may also be helpful for other types of quality indicators to be integrated in the optimization process.

3 HypE: Hypervolume Estimation Algorithm for Multiobjective Optimization

When considering the hypervolume indicator as the objective function of the underlying set problem, the main question is how to make use of this measure within a multiobjective optimizer to guide the search. In the context of a MOEA, this refers to selection and one can distinguish two situations:

1. The selection of solutions to be varied (mating selection)
2. The selection of solutions to be kept in the population (environmental selection)

In the following, we outline a new algorithm based on the hypervolume indicator called HypE. Thereafter, the two selection steps mentioned above as realized in HypE are presented in detail.

3.1 Algorithm

HypE belongs to the class of simple indicator-based evolutionary algorithm, as for instance discussed in Zitzler et al. (2007). As outlined in Algorithm 1, HypE reflects

Algorithm 1 HypE Main Loop

Require: reference set $R \subseteq Z$, population size $N \in \mathbb{N}$, number of generations g_{\max} , number of sampling points $M \in \mathbb{N}$

- 1: initialize population P by selecting N solutions from X uniformly at random
 - 2: $g \leftarrow 0$
 - 3: **while** $g \leq g_{\max}$ **do**
 - 4: $P' \leftarrow \text{matingSelection}(P, R, N, M)$
 - 5: $P'' \leftarrow \text{variation}(P', N)$
 - 6: $P \leftarrow \text{environmentalSelection}(P \cup P'', R, N, M)$
 - 7: $g \leftarrow g + 1$
-

a standard evolutionary algorithm that consists of the successive application of mating selection, variation, and environmental selection. As to mating selection, binary tournament selection is proposed here, although any other selection scheme could be used as well, where the tournament selection is based on the fitness proposed in Sect. 3.2. The procedure *variation* encapsulates the application of mutation and recombination operators to generate N offspring. Finally, environmental selection aims at selecting the most promising N solutions from the multiset-union of parent population and offspring; more precisely, it creates a new population by carrying out the following two steps:

1. First, the union of parents and offspring is divided into disjoint partitions using the principle of nondominated sorting (Goldberg 1989; Deb et al. 2000), also known as dominance depth. Starting with the lowest dominance depth level, the partitions are moved one by one to the new population as long as the first partition is reached that cannot be transferred completely. This corresponds to the scheme used in most hypervolume-based multiobjective optimizers (Emmerich et al. 2005; Igel et al. 2007; Brockhoff and Zitzler 2007).
2. The partition that only fits partially into the new population is then processed using the novel fitness scheme presented in Sect. 3.3. In each step, the fitness values for the partition under consideration are computed and the individual with the worst fitness is removed – if multiple individuals share the same minimal fitness, then one of them is selected uniformly at random. This procedure is repeated until the partition has been reduced to the desired size, i.e., until it fits into the remaining slots left in the new population.

The scheme of first applying non-dominated sorting is similar to other algorithms (e.g., Igel et al. 2007; Emmerich et al. 2005). The differences are: (1) the fitness assignment scheme for mating, (2) the one for environmental selection, and (3) the method how the fitness values are determined. The estimation of the fitness values by means of Monte Carlo sampling is discussed in Sect. 3.4.

3.2 Basic Scheme for Mating Selection

To begin with, we formally define the hypervolume indicator as a basis for the following discussions. Different definitions can be found in the literature, and we here use the one from Zitzler et al. (2008) which draws upon the Lebesgue measure as proposed in Laumanns et al. (1999) and considers a reference set of objective vectors.

Definition 1. Let $A \in \Psi$ be a Pareto set approximation and $R \subset Z$ be a reference set of mutually nondominating objective vectors. Then the hypervolume indicator I_H can be defined as

$$I_H(A, R) := \lambda(H(A, R)), \quad (2)$$

where

$$H(A, R) := \{z \in Z ; \exists a \in A \exists r \in R : f(a) \leq z \leq r\} \quad (3)$$

and λ is the Lebesgue measure with $\lambda(H(A, R)) = \int_{\mathbb{R}^n} \mathbf{1}_{H(A, R)}(z) dz$ and $\mathbf{1}_{H(A, R)}$ being the characteristic function of $H(A, R)$.

The set $H(A, R)$ denotes the set of objective vectors that are enclosed by the front $f(A)$ given by A and the reference set R . It can be further split into partitions $H(S, A, R)$, each associated with a specific subset $S \subseteq A$:

$$H(S, A, R) := \left[\bigcap_{s \in S} H(\{s\}, R) \right] \setminus \left[\bigcup_{a \in A \setminus S} H(\{a\}, R) \right]. \quad (4)$$

The set $H(S, A, R) \subseteq Z$ represents the portion of the objective space that is jointly weakly dominated by the solutions in S and not weakly dominated by any other solution in A . The partitions $H(S, A, R)$ are disjoint and the union of all partitions is $H(A, R)$ which is illustrated in Fig. 1.

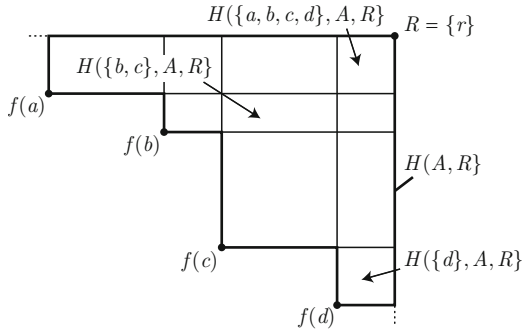


Fig. 1 Illustration of the notions of $H(A, R)$ and $H(S, A, R)$ in the objective space for a Pareto set approximation $A = \{a, b, c, d\}$ and reference set $R = \{r\}$

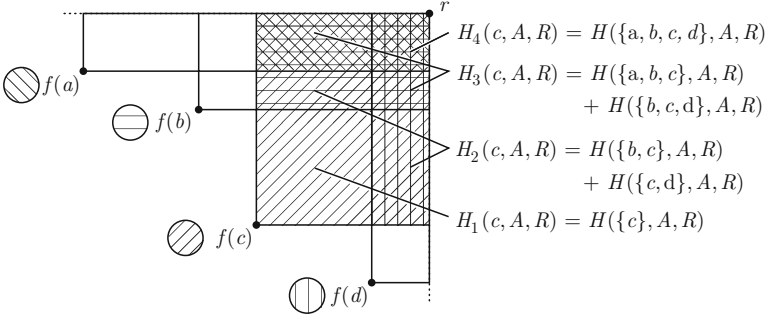


Fig. 2 Illustration of the notions of $H(A, R)$ and $H_i(a, A, R)$ in the objective space for a Pareto set approximation $A = \{a, b, c, d\}$ and reference set $R = \{r\}$

In practice, it is infeasible to determine all distinct $H(S, A, R)$ due to combinatorial explosion. Instead, we will consider a more compact splitting of the dominated objective space that refers to single solutions:

$$H_i(a, A, R) := \bigcup_{\substack{S \subseteq A \\ a \in S \\ |S|=i}} H(S, A, R). \quad (5)$$

According to this definition, $H_i(a, A, R)$ stands for the portion of the objective space that is jointly and solely weakly dominated by a and any $i - 1$ further solutions from A , see Fig. 2. Note that the sets $H_1(a, A, R), H_2(a, A, R), \dots, H_{|A|}(a, A, R)$ are disjoint for a given $a \in A$ while the sets $H_i(a, A, R)$ and $H_i(b, A, R)$ may be overlapping for fixed i and different solutions $a, b \in A$. This slightly different notion has reduced the number of subspaces to be considered from $2^{|A|}$ for $H(S, A, R)$ to $|A|^2$ for $H_i(a, A, R)$.

Now, given an arbitrary population $P \in \Psi$ one obtains for each solution a contained in P a vector $(\lambda(H_1(a, P, R)), \lambda(H_2(a, P, R)), \dots, \lambda(H_{|P|}(a, P, R)))$ of hypervolume contributions. These vectors can be used to assign fitness values to solutions; while most hypervolume-based search algorithms only take the first components, i.e., $\lambda(H_1(a, P, R))$, into account, we here propose the following scheme to aggregate the hypervolume contributions into a single scalar value.

Definition 2. Let $A \in \Psi$ and $R \subset Z$. Then the function I_h with

$$I_h(a, A, R) := \sum_{i=1}^{|A|} \frac{1}{i} \lambda(H_i(a, A, R)) \quad (6)$$

gives for each solution $a \in A$ the hypervolume that can be attributed to a with regard to the overall hypervolume $I_H(A, R)$.

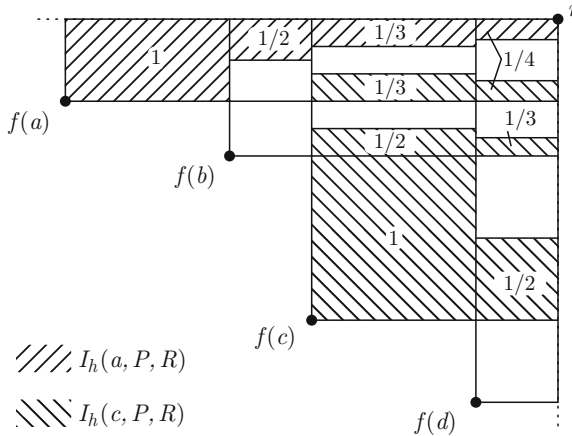


Fig. 3 Illustration of the basic fitness assignment scheme. The fitness of two individuals a and c of a Pareto set approximation $A = \{a, b, c, d\}$ is set to $F_a = I_h(a, A, R)$ and $F_b = I_h(b, A, R)$ respectively (diagonally hatched areas)

The motivation behind this definition is simple: the hypervolume contribution of each partition $H(S, A, R)$ is shared equally among the dominating solutions $s \in S$. That means the portion of Z solely weakly dominated by a specific solution a is fully attributed to a , the portion of Z that a weakly dominates together with another solution b is attributed half to a and so forth – the principle is illustrated in Fig. 3. Thereby, the overall hypervolume is distributed among the distinct solutions according to their hypervolume contributions as the following theorem shows (the proof can be found in the appendix). Note that this scheme does not require that the solutions of the considered Pareto set approximation A are mutually non-dominating; it applies to nondominated and dominated solutions alike.

Next, we will extend and generalize the fitness assignment scheme with regard to the environmental selection phase.

3.3 Extended Scheme for Environmental Selection

In EMO, environmental selection is mostly carried out by first merging parents and offspring and then truncating the resulting union by choosing the subset that represents the best Pareto set approximation. The number k of solutions to be achieved usually equals the population size, and therefore the exact computation of the best subset is computationally infeasible. Instead, the optimal subset is approximated in terms of a greedy heuristic (Zitzler and Künzli 2004; Brockhoff and Zitzler 2007): all solutions are evaluated with respect to their usefulness and the least important solution is removed; this process is repeated until k solutions have been removed.

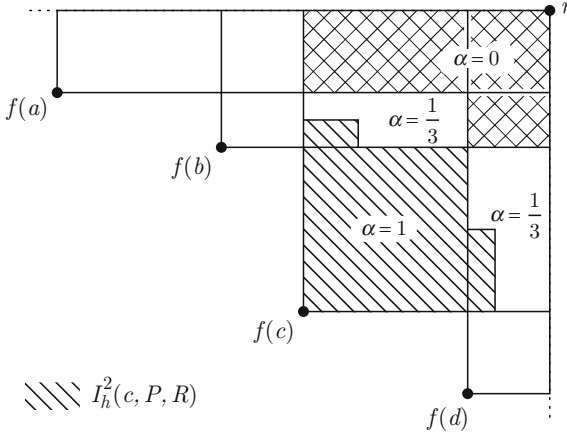


Fig. 4 The figure shows for $A = \{a, b, c, d\}$ and $R = \{r\}$ (1) which portion of the objective space remains dominated if any two solutions are removed from A (shaded area), and (2) the probabilities α that a particular area that can be attributed to $a \in A$ is lost if a is removed from A together with any other solution in A

The key issue with respect to the above greedy strategy is how to evaluate the usefulness of a solution. The scheme presented in Definition 2 has the drawback that portions of the objective space are taken into account that for sure will not change. Suppose, for instance, a population with four solutions as shown in Fig. 4; when two solutions need to be removed ($k = 2$), then the subspaces $H(\{a, b, c\}, P, R)$, $H(\{b, c, d\}, P, R)$, and $H(\{a, b, c, d\}, P, R)$ remain weakly dominated independently of which solutions are deleted. This observation led to the idea of considering the expected loss in hypervolume that can be attributed to a particular solution when exactly k solutions are removed. In detail, we consider for each $a \in P$ the average hypervolume loss over all subsets $S \subseteq P$ that contain a and $k - 1$ further solutions; this value can be easily computed by slightly extending the scheme from Definition 2 as follows.

Definition 3. Let $A \in \Psi$, $R \subset Z$, and $k \in \{0, 1, \dots, |A|\}$. Then the function I_h^k with

$$I_h^k(a, A, R) := \sum_{i=1}^k \frac{\alpha_i}{i} \lambda(H_i(a, A, R)), \tag{7}$$

where

$$\alpha_i := \prod_{j=1}^{i-1} \frac{k-j}{|A|-j} \tag{8}$$

gives for each solution $a \in A$ the expected hypervolume loss that can be attributed to a when a and $k - 1$ uniformly randomly chosen solutions from A are removed from A .

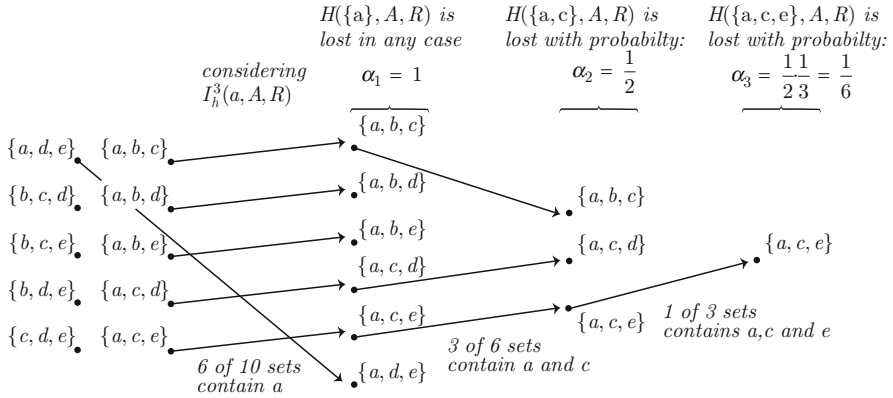


Fig. 5 Illustrates the calculation of α_i when removing a and two other individuals from the set $A = \{a, b, c, d, e\}$. Each subset of three individuals that contains a is considered equally likely; therefore, the chance of losing a and c is $3/6$ and the probability of losing a, c , and e is $(1/2)(1/3)$

The correctness of (7) can be proved by mathematical induction (Bader and Zitzler 2008). In the following example we illustrates the calculation of α_i to put the idea of expected loss across:

Example 1. Out of five individuals $A = \{a, b, c, d, e\}$, three have to be removed and we want to know $I_h^3(a, A, R)$ (see Fig. 5). To this end, the expected shares of $H_i(a, A, R)$ that are lost when removing a – represented by α_i – have to be determined. The first coefficient is $\alpha_1 = 1$, because $H_1(a, A, R)$ is lost for sure, since the space is only dominated by a . Calculating the probability of losing the partitions dominated by a and a second individual $H_2(a, A, R)$ is a little bit more involved. Without loss of generality we consider the space dominated only by a and c . In addition to a , there are two individuals to be removed out of the remaining four individuals. Since we assume, by approximation every removal is equally probable, the chance of removing c and thereby losing the partition $H(\{a, c\}, A, R)$ is $\alpha_2 = 1/2$. Finally, for α_3 three points including a have to be taken off. The probability of removing a and any other individual was just derived to be $1/2$. Out of three remaining individuals, the chance to remove one particular, say e , is $1/3$. This gives the coefficient $\alpha_3 = (1/2)(1/3)$. For reasons of symmetry, these calculations hold for any other partition dominated by a and $i - 1$ points. Therefore, the expected loss of $H_i(a, A, R)$ is $\alpha_i \lambda(H_i(a, A, R))$. Since i points share the partition $H_i(a, A, R)$, the expected loss is additionally multiplied by $1/i$.

Notice that $I_h^1(a, A, R) = \lambda(H_1(a, A, R))$ and $I_h^{|A|}(a, A, R) = I_h(a, A, R)$, i.e., this modified scheme can be regarded as a generalization of the scheme presented in Definition 2 and the commonly used fitness assignment strategy for hypervolume-based search (Knowles and Corne 2003; Emmerich et al. 2005; Igel et al. 2007; Bader et al. 2008).

The fitness values $I_h^k(a, A, R)$ can be calculated on the fly for all individuals $a \in A$ by a slightly extended version of the “hypervolume by slicing objectives” (Zitzler 2001; Knowles 2002; While et al. 2006) algorithm, which traverse recursively one objective after another. It differs from existing methods in that it allows (1) to consider a set R of reference points and (5) to compute all fitness values, e.g., the $I_h^1(a, P, R)$ values for $k = 1$, in parallel for any number of objectives instead of subsequently as in Beume et al. (2007). Basically, the modification concerns also considering solutions, which are dominated according to a particular scanline. A detailed description of the extended algorithm can be found in Bader and Zitzler (2008).

Unfortunately, the worst case complexity of the algorithm is $\mathcal{O}(|A|^n)$ which renders the exact calculation inapplicable for problems with more than about five objectives. However, in the context of randomized search heuristics one may argue that the exact fitness values are not crucial and approximated values may be sufficient. These considerations lead to the idea of estimating the fitness values by Monte Carlo sampling, whose basic principle is described in the following section.

3.4 Estimating the Fitness Values Using Monte Carlo Sampling

To approximate the fitness values according to Definitions 2 and 3, we need to estimate the Lebesgue measures of the domains $H_i(a, P, R)$ where $P \in \Psi$ is the population. Since these domains are all integrable, their Lebesgue measure can be approximated by means of Monte Carlo simulation.

For this purpose, a sampling space $S \subseteq Z$ has to be defined with the following properties: (1) the hypervolume of S can easily be computed, (2) samples from the space S can be generated fast, and (3) S is a superset of the domains $H_i(a, P, R)$ the hypervolumes of which one would like to approximate. The latter condition is met by setting $S = H(P, R)$, but since it is hard both to calculate the Lebesgue measure of this sampling space and to draw samples from it, we propose using the axis-aligned minimum bounding box containing the $H_i(a, P, R)$ subspaces instead, i.e.:

$$S := \{(z_1, \dots, z_n) \in Z \mid \forall 1 \leq i \leq n : l_i \leq z_i \leq u_i\}, \quad (9)$$

where

$$\begin{aligned} l_i &:= \min_{a \in P} f_i(a) \\ u_i &:= \max_{(r_1, \dots, r_n) \in R} r_i \end{aligned} \quad (10)$$

for $1 \leq i \leq n$. Hence, the volume V of the sampling space S is given by $V = \prod_{i=1}^n \max\{0, u_i - l_i\}$.

Now given S , sampling is carried out by selecting M objective vectors s_1, \dots, s_M from S uniformly at random. For each s_j it is checked whether it lies in any partition $H_i(a, P, R)$ for $1 \leq i \leq k$ and $a \in P$. This can be determined in two steps: first, it is verified that s_j is “below” the reference set R , i.e., it exists $r \in R$ that is dominated by s_j ; second, it is verified that the multiset A of those population members

dominating s_j is not empty. If both conditions are fulfilled, then we know that – given A – the sampling point s_j lies in all partitions $H_i(a, P, R)$ where $i = |A|$ and $a \in A$. This situation will be denoted as a *hit* regarding the i th partition of a . If any of the above two conditions is not fulfilled, then we call s_j a *miss*. Let $X_j^{(i,a)}$ denote the corresponding random variable that is equal to 1 in case of a hit of s_j regarding the i th partition of a and 0 otherwise.

Based on the M sampling points, we obtain an estimate for $\lambda(H_i(a, P, R))$ by simply counting the number of hits and multiplying the hit ratio with the volume of the sampling box:

$$\hat{\lambda}(H_i(a, P, R)) = \frac{\sum_{j=1}^M X_j^{(i,a)}}{M} V. \quad (11)$$

This value approaches the exact value $\lambda(H_i(a, P, R))$ with increasing M by the law of large numbers. Due to the linearity of the expectation operator, the fitness scheme according to (7) can be approximated by replacing the Lebesgue measure with the respective estimates given by (11):

$$\hat{I}_h^k(a, P, R) = \sum_{i=1}^k \frac{\alpha_i}{i} \left(\frac{\sum_{j=1}^M X_j^{(i,a)}}{M} V \right). \quad (12)$$

Note that the partitions $H_i(a, P, R)$ with $i > k$ do not need to be considered for the fitness calculation as they do not contribute to the I_h^k values that we would like to estimate, cf. Definition 3.

4 Experiments

4.1 Experimental Setup

HypE is implemented within the PISA framework (Bleuler et al. 2003) and tested in two versions: the first, denoted by HypE, uses fitness-based mating selection as described in Sect. 3.2, while the second, HypE*, employs a uniform mating selection scheme where all individuals have the same probability of being chosen for reproduction. Unless stated otherwise, for sampling the number of sampling points is fixed to $M = 10,000$ and kept constant during a run.

On the one hand, HypE and HypE* are compared to three popular MOEAs, namely NSGA-II (Deb et al. 2000), SPEA2 (Zitzler et al. 2002), and IBEA (in combination with the ε -indicator) (Zitzler and Künzli 2004). Since these algorithms are not designed to optimize the hypervolume, it cannot be expected that they perform particularly well when measuring the quality of the approximation in terms of the hypervolume indicator. Nevertheless, they serve as an important reference as they are considerably faster than hypervolume-based search algorithms and therefore can execute a substantially larger number of generations when keeping the

available computation time fixed. On the other hand, we include the sampling-based optimizer proposed in Bader et al. (2008), here denoted as SHV (sampling-based hypervolume-oriented algorithm); finally, to study the influence of the nondominated sorting we also include a simple HypE variant named RS (random selection) where all individuals of the same dominance depth level are assigned the same constant fitness value. Thereby, the selection pressure is only maintained by the nondominated sorting carried out during the environmental selection phase.

In this paper, we focus on many-objective problems for which exact hypervolume-based methods (e.g., Emmerich et al. 2005; Igel et al. 2007) are not applicable. Therefore, we did not include these algorithms in the experiments. However, the interested reader is referred to Bader and Zitzler (2008) where HypE is compared to an exact hypervolume algorithm.

As basis for the comparisons, the DTLZ (Deb et al. 2005), and the WFG (Huband et al. 2006) test problem suites are considered since they allow the number of objectives to be scaled arbitrarily – here, ranging from 2 to 50 objectives. For the DTLZ problem, the number of decision variables is set to 300, while for the WFG problems individual values are used.³

The individuals are represented by real vectors, where a polynomial distribution is used for mutation and the SBX-20 operator is used for recombination (Deb 2001). The recombination and mutation probabilities are set according to Deb et al. (2005).

For each benchmark function, 30 runs are carried out per algorithm using a population size of $N = 50$. Either the maximum number of generations was set to $g_{\max} = 200$ (results shown in Table 1) or the runtime was fixed to 30 min for each run (results shown in Fig. 6). For each run, the hypervolume of the last population is determined, where for less than six objectives these are calculated exactly and otherwise approximated by Monte Carlo sampling. For each algorithm A_i , the hypervolume values are then subsumed under the performance score $P(A_i)$, which represents the number of other algorithms that achieved significantly higher hypervolume values on the particular test case. The test for significance is done using Kruskal–Wallis and the Conover Inman post hoc tests (Conover 1999). For a full description of the performance score, please see Zamora and Burguete (2008).

4.2 Results

Table 1 shows the performance score and mean hypervolume of the different algorithms on six test problems. In 18 instances, HypE is better than HypE*, while vice versa HypE* is better than HypE only in four cases. HypE reaches the best performance score overall. Summing up all performance scores, HypE yields the best

³ The number of decision variables (first value in parenthesis) and their decomposition into position (second value) and distance variables (third value) as used by the WFG test function for different number of objectives are: 2d (24, 4, 20); 3d (24, 4, 20); 5d (50, 8, 42); 7d (70, 12, 58); 10d (59,9,50); 25d (100, 24, 76); 50d (199,49,150).

Table 1 Comparison of HypE to different EAs for the hypervolume indicator, where n denotes the number of objectives. The first number represents the performance score $P(A_i)$, which stands for the number of contenders significantly dominating algorithm A_i . The number in brackets denote the hypervolume value (to be maximized), normalized to the minimum and maximum value observed on the testproblem

	n	SHV	IBEA	NSGA-II	RS	SPEA2	HypE	HypE*
DTLZ 2	2	2 (0.438)	0 (0.871)	5 (0.306)	5 (0.278)	2 (0.431)	1 (0.682)	4 (0.362)
	3	2 (0.995)	0 (0.998)	5 (0.683)	6 (0.491)	4 (0.888)	1 (0.996)	3 (0.994)
	5	1 (0.998)	0 (0.999)	4 (0.808)	6 (0.324)	5 (0.795)	2 (0.998)	3 (0.998)
	7	3 (0.998)	0 (1.000)	5 (0.808)	6 (0.340)	4 (0.850)	1 (0.999)	1 (0.999)
	10	3 (0.999)	2 (1.000)	5 (0.825)	6 (0.290)	4 (0.868)	0 (1.000)	0 (1.000)
DTLZ 4	25	3 (0.999)	2 (1.000)	4 (0.965)	6 (0.301)	5 (0.882)	0 (1.000)	0 (1.000)
	50	3 (1.000)	2 (1.000)	4 (0.998)	6 (0.375)	5 (0.917)	0 (1.000)	0 (1.000)
	2	1 (0.848)	0 (0.928)	3 (0.732)	3 (0.834)	2 (0.769)	1 (0.779)	3 (0.711)
	3	1 (0.945)	0 (0.989)	3 (0.777)	3 (0.774)	2 (0.860)	0 (0.987)	2 (0.922)
	5	1 (0.997)	0 (0.998)	4 (0.749)	5 (0.558)	6 (0.537)	2 (0.992)	2 (0.992)
DTLZ 7	7	1 (0.999)	0 (1.000)	4 (0.902)	6 (0.569)	5 (0.814)	2 (0.999)	2 (0.999)
	10	2 (1.000)	0 (1.000)	4 (0.988)	6 (0.560)	5 (0.960)	1 (1.000)	0 (1.000)
	25	3 (1.000)	2 (1.000)	4 (1.000)	6 (0.546)	5 (0.991)	0 (1.000)	0 (1.000)
	50	2 (1.000)	2 (1.000)	4 (1.000)	6 (0.517)	5 (0.999)	0 (1.000)	0 (1.000)
	2	0 (0.945)	1 (0.898)	6 (0.739)	2 (0.818)	4 (0.817)	2 (0.853)	1 (0.876)
DTLZ 7	3	0 (0.993)	1 (0.987)	6 (0.633)	4 (0.794)	5 (0.722)	3 (0.970)	2 (0.980)
	5	0 (0.988)	0 (0.986)	6 (0.478)	4 (0.672)	5 (0.569)	2 (0.868)	2 (0.862)
	7	0 (0.981)	1 (0.958)	5 (0.348)	4 (0.559)	5 (0.352)	2 (0.877)	2 (0.870)
	10	0 (0.986)	1 (0.831)	4 (0.137)	6 (0.057)	4 (0.166)	2 (0.744)	1 (0.781)
	25	0 (0.973)	0 (0.966)	3 (0.856)	2 (0.893)	4 (0.671)	2 (0.889)	3 (0.825)
50	1 (0.767)	0 (0.966)	5 (0.233)	4 (0.254)	6 (0.020)	2 (0.684)	3 (0.675)	

Table 1 (continued)

WFG 1	2	4 (0.567)	0 (0.949)	1 (0.792)	6 (0.160)	1 (0.776)	2 (0.744)	4 (0.557)
	3	4 (0.792)	3 (0.811)	3 (0.827)	6 (0.207)	1 (0.881)	0 (0.985)	1 (0.894)
	5	4 (0.766)	5 (0.703)	2 (0.832)	6 (0.291)	2 (0.820)	0 (0.973)	1 (0.951)
	7	4 (0.647)	5 (0.649)	2 (0.814)	6 (0.189)	2 (0.812)	0 (0.956)	1 (0.937)
	10	6 (0.402)	4 (0.843)	2 (0.932)	5 (0.562)	2 (0.937)	0 (0.977)	0 (0.975)
WFG 2	25	6 (0.183)	4 (0.930)	0 (0.971)	5 (0.815)	3 (0.965)	0 (0.972)	0 (0.973)
	50	6 (0.210)	4 (0.869)	2 (0.962)	4 (0.823)	2 (0.961)	0 (0.971)	0 (0.970)
	2	1 (0.987)	4 (0.962)	3 (0.974)	6 (0.702)	4 (0.969)	0 (0.990)	0 (0.989)
	3	0 (0.556)	3 (0.475)	3 (0.406)	6 (0.261)	2 (0.441)	0 (0.446)	0 (0.372)
	5	0 (0.671)	0 (0.533)	0 (0.644)	6 (0.351)	0 (0.624)	0 (0.557)	3 (0.503)
WFG 3	7	0 (0.632)	0 (0.747)	1 (0.409)	5 (0.155)	0 (0.837)	0 (0.528)	0 (0.630)
	10	0 (0.971)	0 (0.988)	0 (0.978)	5 (0.020)	2 (0.962)	0 (0.981)	1 (0.966)
	25	0 (0.951)	0 (0.951)	2 (0.935)	6 (0.072)	2 (0.933)	2 (0.934)	2 (0.928)
	50	3 (0.538)	0 (0.962)	0 (0.959)	6 (0.076)	0 (0.952)	2 (0.945)	3 (0.943)
	2	2 (0.994)	0 (0.997)	4 (0.991)	6 (0.559)	4 (0.990)	0 (0.997)	2 (0.994)
WFG 3	3	2 (0.995)	3 (0.981)	4 (0.966)	6 (0.689)	4 (0.966)	0 (0.999)	1 (0.998)
	5	6 (0.339)	0 (0.974)	3 (0.946)	5 (0.760)	4 (0.932)	0 (0.977)	0 (0.971)
	7	6 (0.105)	2 (0.975)	3 (0.961)	5 (0.709)	4 (0.958)	0 (0.983)	0 (0.982)
	10	6 (0.088)	1 (0.973)	3 (0.947)	5 (0.792)	4 (0.933)	0 (0.980)	1 (0.976)
	25	6 (0.037)	0 (0.983)	3 (0.965)	5 (0.758)	3 (0.963)	1 (0.974)	1 (0.977)
50	6 (0.059)	0 (0.981)	2 (0.972)	5 (0.731)	2 (0.973)	0 (0.976)	0 (0.979)	

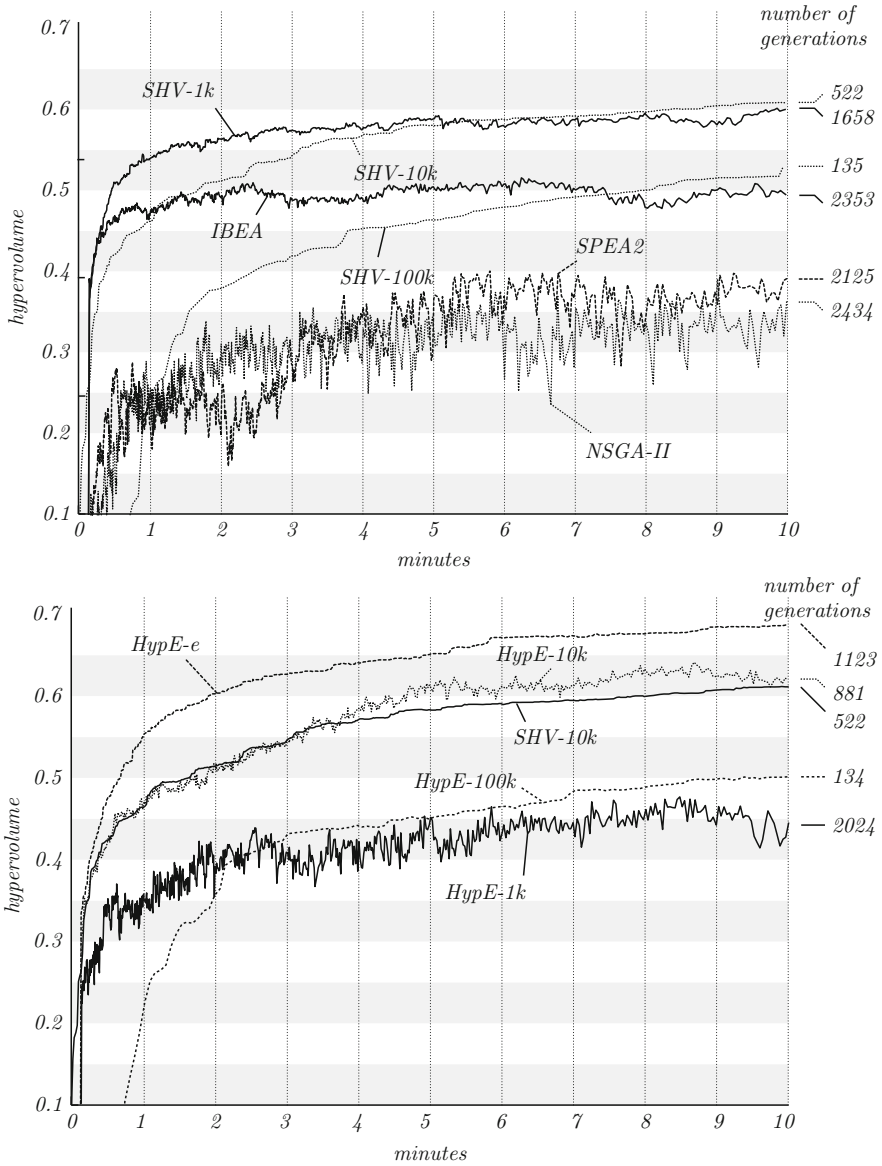


Fig. 6 Hypervolume process over 30 min of HypE and SHV for 100; 1,000; 10,000 and 100,000 samples each, denoted by the suffices 100, 1k, 10k and 100k respectively. The test problem is WFG9 for three objectives and NSGA-II, SPEA2, and IBEA are shown as reference. The numbers at the right border of the figures indicate the total number of generations reached after 30 min. The results are split in two figures with identical axis for the sake of clarity

total (33), followed by HypE* (55), IBEA (55) and SHV, the method proposed in Bader et al. (2008) (97). SPEA2 and NSGA-II reach almost the same score (136 and 140 respectively), clearly outperforming random selection (217). For five out of six testproblems HypE obtains better hypervolume values than SHV. On DTLZ7 however, SHV as well as IBEA outperform HypE. This might be due to the discontinuous shape of the DTLZ7 testfunction, for which the advanced fitness scheme does not give an advantage.

The better Pareto set approximations of HypE come at the expense of longer execution time, e.g., in comparison to SPEA2 or NSGA-II. We therefore investigate, whether the fast NSGA-II and SPEA2 will not overtake HypE given a constant amount of time. Figure 6 shows the hypervolume of the Pareto set approximations over time for HypE using the exact fitness values as well as the estimated values for different samples sizes M .

Even though SPEA2, NSGA-II and even IBEA are able to process twice as many generations as the exact HypE, they do not reach its hypervolume. In the three dimensional example used, HypE can be run sufficiently fast without approximating the fitness values. Nevertheless, the sampled version is used as well to show the dependency of the execution time and quality on the number of samples M . Via M , the execution time of HypE can be traded off against the quality of the Pareto set approximation. The fewer samples are used, the more the behavior of HypE resembles random selection. On the other hand by increasing M , the quality of exact calculation can be achieved, increasing the execution time, though. For example, with $M = 1,000$, HypE is able to carry out nearly the same number of generations as SPEA2 or NSGA-II, but the Pareto set is just as good as when 100,000 samples are used, needing only a fifteenth the number of generations. In the example given, $M = 10,000$ represents the best compromise, but the number of samples should be increased in two cases: (1) when the fitness evaluation take more time, and (2) when more generations are used. The former will affect the faster algorithm much more and increasing the number of samples will influence the execution time much less. (ii) More generations are used. In the latter case, HypE using more samples might overtake the faster versions with fewer samples, since those are more vulnerable to stagnation.

In this three objective scenario, SHV can compete with HypE. This is mainly because the sampling boxes that SHV relies on are tight for small number of objectives; as this number increases, however, experiments not shown here indicate that the quality of SHV decreases in relation to HypE.

5 Conclusion

This paper proposes HypE, a novel hypervolume-based evolutionary algorithm. Both its environmental and mating selection step rely on a new fitness assignment scheme based on the Lebesgue measure, where the values can be both exactly

calculated and estimated by means of sampling. In contrast to other hypervolume based algorithms, HypE is thus applicable to higher dimensional problems.

The performance of HypE was compared to other algorithms against the hypervolume indicator of the Pareto set approximation. In particular, the algorithms were tested on test problems of the WFG and DTLZ test suite. Simulations results show that HypE is a competitive search algorithm; this especially applies to higher dimensional problems, which indicates using the Lebesgue measure on many objectives is a convincing approach.

A promising direction of future research is the development of advanced adaptive sampling strategies that exploit the available computing resources most effectively.

HypE is available for download at <http://www.tik.ee.ethz.ch/sop/> → download/supplementary/hype/.

Acknowledgements Johannes Bader has been supported by the Indo-Swiss Joint Research Program IT14.

References

- Bader J, Zitzler E (2008) HypE: an algorithm for fast hypervolume-based many-objective optimization. TIK Report 286, Computer Engineering and Networks Laboratory (TIK), ETH Zurich, November 2008
- Bader J, Deb K, Zitzler E (2008) Faster hypervolume-based search using Monte Carlo sampling. In: Conference on multiple criteria decision making (MCDM 2008). Springer, Berlin
- Beume N, Rudolph G (2006) Faster S-metric calculation by considering dominated hypervolume as Klee's measure problem. Technical Report CI-216/06, Sonderforschungsbereich 531 Computational Intelligence, Universität Dortmund. Shorter version published at IASTED international conference on computational intelligence (CI 2006)
- Beume N, Fonseca CM, Lopez-Ibanez M, Paquete L, Vahrenhold J (2007a) On the complexity of computing the hypervolume indicator. Technical Report CI-235/07, University of Dortmund, December 2007
- Beume N, Naujoks B, Emmerich M (2007b) SMS-EMOA: multiobjective selection based on dominated hypervolume. *Eur J Oper Res* 181:1653–1669
- Bleuler S, Laumanns M, Thiele L, Zitzler E (2003) PISA – a platform and programming language independent interface for search algorithms. In: Fonseca CM et al (eds) Conference on evolutionary multi-criterion optimization (EMO 2003). LNCS, vol 2632. Springer, Berlin, pp 494–508
- Bradstreet L, Barone L, While L (2006) Maximising hypervolume for selection in multi-objective evolutionary algorithms. In: Congress on evolutionary computation (CEC 2006), pp 6208–6215, Vancouver, BC, Canada. IEEE
- Bringmann K, Friedrich T (2008) Approximating the volume of unions and intersections of high-dimensional geometric objects. In: Hong SH, Nagamochi H, Fukunaga T (eds) International symposium on algorithms and computation (ISAAC 2008), LNCS, vol 5369. Springer, Berlin, pp 436–447
- Brockhoff D, Zitzler E (2007) Improving hypervolume-based multiobjective evolutionary algorithms by using objective reduction methods. In: Congress on evolutionary computation (CEC 2007), pp 2086–2093. IEEE Press
- Conover WJ (1999) Practical nonparametric statistics, 3rd edn. Wiley, New York
- Deb K (2001) Multi-objective optimization using evolutionary algorithms. Wiley, Chichester, UK

- Deb K, Agrawal S, Pratap A, Meyarivan T (2000) A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: NSGA-II. In: Schoenauer M et al (eds) Conference on parallel problem solving from nature (PPSN VI). LNCS, vol 1917. Springer, Berlin, pp. 849–858
- Deb K, Thiele L, Laumanns M, Zitzler E (2005) Scalable test problems for evolutionary multi-objective optimization. In: Abraham A, Jain R, Goldberg R (eds) Evolutionary multiobjective optimization: theoretical advances and applications, chap 6. Springer, Berlin, pp 105–145
- Emmerich M, Beume N, Naujoks B (2005) An EMO algorithm using the hypervolume measure as selection criterion. In: Conference on evolutionary multi-criterion optimization (EMO 2005). LNCS, vol 3410. Springer, Berlin, pp 62–76
- Emmerich M, Deutz A, Beume N (2007) Gradient-based/evolutionary relay hybrid for computing Pareto front approximations maximizing the S-metric. In: Hybrid metaheuristics. Springer, Berlin, pp 140–156
- Everson R, Fieldsend J, Singh S (2002) Full elite-sets for multiobjective optimisation. In: Parmee IC (ed) Conference on adaptive computing in design and manufacture (ADCM 2002), pp 343–354. Springer, London
- Fleischer M (2003) The measure of Pareto optima. Applications to multi-objective metaheuristics. In: Fonseca CM et al (eds) Conference on evolutionary multi-criterion optimization (EMO 2003), Faro, Portugal. LNCS, vol 2632. Springer, Berlin, pp 519–533
- Fonseca CM, Paquete L, López-Ibáñez M (2006) An improved dimension-sweep algorithm for the hypervolume indicator. In: Congress on evolutionary computation (CEC 2006), pp 1157–1163, Sheraton Vancouver Wall Centre Hotel, Vancouver, BC Canada. IEEE Press
- Goldberg DE (1989) Genetic algorithms in search, optimization, and machine learning. Addison-Wesley, Reading, MA
- Huband S, Hingston P, White L, Barone L (2003) An evolution strategy with probabilistic mutation for multi-objective optimisation. In: Congress on evolutionary computation (CEC 2003), vol 3, pp 2284–2291, Canberra, Australia. IEEE Press.
- Huband S, Hingston P, Barone L, White L (2006) A review of multiobjective test problems and a scalable test problem toolkit. *IEEE Trans Evol Comput* 10(5):477–506
- Igel C, Hansen N, Roth S (2007) Covariance matrix adaptation for multi-objective optimization. *Evol Comput* 15(1):1–28
- Knowles JD (2002) Local-search and hybrid evolutionary algorithms for Pareto optimization. PhD thesis, University of Reading
- Knowles J, Corne D (2003) Properties of an adaptive archiving algorithm for storing nondominated vectors. *IEEE Trans Evol Comput* 7(2):100–116
- Laumanns M, Rudolph G, Schwefel H-P (1999) Approximating the Pareto set: concepts, diversity issues, and performance assessment. Technical Report CI-7299, University of Dortmund
- Mostaghim S, Branke J, Schmeck H (2007) Multi-objective particle swarm optimization on computer grids. In: Proceedings of the 9th annual conference on genetic and evolutionary computation (GECCO 2007), pp 869–875, New York, USA. ACM
- Nicolini M (2005) A two-level evolutionary approach to multi-criterion optimization of water supply systems. In: Conference on evolutionary multi-criterion optimization (EMO 2005). LNCS, vol 3410. Springer, Berlin, pp 736–751
- Van Veldhuizen DA (1999) Multiobjective evolutionary algorithms: classifications, analyses, and new innovations. PhD thesis, Graduate School of Engineering, Air Force Institute of Technology, Air University
- Wagner T, Beume N, Naujoks B (2007) Pareto-, aggregation-, and indicator-based methods in many-objective optimization. In: Obayashi S et al (eds) Conference on evolutionary multi-criterion optimization (EMO 2007). LNCS, vol 4403. Springer, Berlin, pp 742–756. Extended version published as internal report of Sonderforschungsbereich 531 Computational Intelligence CI-217/06, Universität Dortmund, September 2006
- White L (2005) A new analysis of the LeBMeasure algorithm for calculating hypervolume. In: Conference on evolutionary multi-criterion optimization (EMO 2005), Guanajuato, México. LNCS, vol 3410. Springer, Berlin, pp 326–340

- While L, Hingston P, Barone L, Huband S (2006) A faster algorithm for calculating hypervolume. *IEEE Trans Evol Comput* 10(1):29–38
- Yang Q, Ding S (2007) Novel algorithm to calculate hypervolume indicator of Pareto approximation set. In: *Advanced intelligent computing theories and applications. With aspects of theoretical and methodological issues, Third international conference on intelligent computing (ICIC 2007)*, vol 2, pp 235–244
- Zamora LP, Burguete STG (2008) Second-order preferences in group decision making. *Oper Res Lett* 36:99–102
- Zitzler E (1999) Evolutionary algorithms for multiobjective optimization: methods and applications. PhD thesis, ETH Zurich, Switzerland
- Zitzler E (2001) Hypervolume metric calculation. <ftp://ftp.tik.ee.ethz.ch/pub/people/zitzler/hypervol.c>
- Zitzler E, Künzli S (2004) Indicator-based selection in multiobjective search. In: Yao X et al (eds) *Conference on parallel problem solving from nature (PPSN VIII)*. LNCS, vol 3242. Springer, Berlin, pp 832–842
- Zitzler E, Thiele L (1998a) An evolutionary approach for multiobjective optimization: the strength Pareto approach. TIK Report 43, Computer Engineering and Networks Laboratory (TIK), ETH Zurich
- Zitzler E, Thiele L (1998b) Multiobjective optimization using evolutionary algorithms – a comparative case study. In: *Conference on parallel problem solving from nature (PPSN V)*, pp 292–301, Amsterdam
- Zitzler E, Thiele L (1999) Multiobjective evolutionary algorithms: a comparative case study and the strength Pareto approach. *IEEE Trans Evol Comput* 3(4):257–271
- Zitzler E, Laumanns M, Thiele L (2002) SPEA2: improving the strength Pareto evolutionary algorithm for multiobjective optimization. In: Giannakoglou KC 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, Thiele L, Laumanns M, Fonseca CM, Grunert da Fonseca V (2003) Performance assessment of multiobjective optimizers: an analysis and review. *IEEE Trans Evol Comput* 7(2):117–132
- Zitzler E, Brockhoff D, Thiele L (2007) The hypervolume indicator revisited: on the design of Pareto-compliant indicators via weighted integration. In: Obayashi S et al (eds) *Conference on evolutionary multi-criterion optimization (EMO 2007)*. LNCS, vol 4403. Springer, Berlin, pp 862–876
- Zitzler E, Thiele L, Bader J (2008) On set-based multiobjective optimization. TIK Report 300, Computer Engineering and Networks Laboratory (TIK), ETH Zurich