Noname manuscript No.
(will be inserted by the editor)

An alternative hypervolume-based selection mechanism for multi-objective evolutionary algorithms

Adriana Menchaca-Mendez · Carlos A. Coello Coello

Received: date / Accepted: date

Abstract In this paper, we are interested in selection mechanisms based on the hypervolume indicator with a particular emphasis on the mechanism used in an improved version of the *S metric selection Evolutionary Multi-Objective Algorithm* (SMS-EMOA) called iSMS-EMOA, which exploits the locality property of the hypervolume. Here, we propose a new selection scheme which approximates the contribution of solutions to the hypervolume and it is designed to preserve the locality property exploited by iSMS-EMOA. This approach is proposed as an alternative to the use of exact hypervolume calculations, and is aimed for solving many-objective optimization problems. The proposed approach is called "approximate version of the improved SMS-EMOA (aviSMS-EMOA)" and is validated using standard test problems (with three or more objectives) and performance indicators taken from the specialized literature. Our preliminary results indicate that our proposed approach is a good alternative to solve many-objective optimization problems, if we consider both quality in the solutions and running time required to obtain them because it outperforms two versions of the original SMS-EMOA that approximate the contributions to the hypervolume, it outperforms MOEA/D using Penalty Boundary Intersection (PBI) and it is competitive with respect to the original SMS-EMOA in several of the test problems adopted. Also, its computational cost is reasonable (it is slower than MOEA/D but it is faster than SMS-EMOA).

Keywords Multi-objective evolutionary algorithms · Selection operators · Hypervolume indicator

Adriana Menchaca-Mendez
CINVESTAV-IPN,
Evolutionary Computation Group
Departamento de Computación
Av. IPN 2508. San Pedro Zacatenco
México D.F. 07300, MÉXICO,
E-mail: : adriana.menchacamendez@gmail.com

Carlos A. Coello Coello
CINVESTAV-IPN,
Evolutionary Computation Group
Departamento de Computación
Av. IPN 2508. San Pedro Zacatenco
México D.F. 07300, MÉXICO,
E-mail: : ccoello@cs.cinvestav.mx

1 Introduction

We are interested in the so-called multi-objective optimization problems (MOPs). These problems involve multiple objective functions which are in conflict with each other. In MOPs, the notion of optimality refers to the best possible trade-offs among the objectives. Consequently, there are several possible solutions (the so-called *Pareto optimal set* whose image is called the *Pareto front*). The use of evolutionary algorithms for solving MOPs has become very popular and has two main goals [11]: (i) to find solutions that are as close as possible to the true Pareto front and (ii) to produce solutions that are spread along the Pareto front as uniformly as possible.

There are different indicators to assess the quality of the approximation of the Pareto-optimal set generated by a *multi-objective evolutionary algorithm (MOEA)*. Some of them are: error ratio, generational distance, hypervolume, ϵ-indicator, $R2$-indicator, two set coverage, and nondominated vector addition [11]. But only a few indicators are "Pareto Compliant" (in [32], it is shown that the hypervolume indicator is the only unary indicator which is strictly "Pareto compliant").

When studying MOEAs, we find two main types of selection mechanisms: (i) those that incorporate the concept of Pareto optimality (e.g., NSGA-II [13]), and (ii) those that do not use Pareto dominance to select individuals. Within this class, those that use an indicator-based selection mechanism [30] have become particularly popular.

The use of Pareto-based selection has been very popular for the last 20 years, but it has several limitations. From them, its poor scalability with respect to the number of objectives in a MOP is, perhaps, the most remarkable one. The quick growth in the number of nondominated solutions as we increase the number of objectives, rapidly dilutes the effect of the selection mechanism of a MOEA [17]. Because of this limitation, MOEAs of type (ii) have become relatively popular in recent years as an alternative that allows us to tackle problems having four or more objectives (the so-called "many-objective optimization problems").

In this work, we are interested in MOEAs based on the hypervolume indicator (I_H). I_H is the only unary indicator which is known to be strictly "Pareto compliant" [32]. I_H was originally proposed by Zitzler and Thiele in [31], and it's defined as the size of the space covered by the Pareto optimal solutions. I_H rewards convergence towards the Pareto front as well as the maximum spread of the solutions obtained. Fleischer proved in [18] that, given a finite search space and a reference point, *maximizing the hypervolume indicator is equivalent to finding the Pareto optimal set.* The main disadvantage of the hypervolume indicator is its high computational cost (the problem of computing I_H is **#P-hard** [6]).[1] Consequently, in the last few years, several proposals have been made to address this problem. Some authors have proposed to reduce the dimensionality of the MOP [9], others have proposed improvements to the calculation of the contribution to the hypervolume indicator of each individual in the population [5,16], as well as mechanisms to approximate the contribution of each individual in the population [23,3,7,8]. Also, other authors have proposed a new competition scheme for selection mechanisms based on I_H. In this scheme, only three individuals compete to survive. Thus, we only need to calculate the contribution of three individuals and choose the best from them [25]. In contrast, most of the current hypervolume-based MOEAs need to calculate the contribution of each individual in the population in order to choose the best from them.

In this paper, we study the competition schemes used in MOEAs based on the hypervolume indicator and, we also study some selection techniques based on the approximation of the hypervolume. Then, we propose a new selection mechanism based on the competition scheme proposed by Menchaca and Coello in [25] and on the technique to approximate the contribution to the hypervolume proposed by Bringmann and Friedrich in [7]. This idea came from the fact that we only need to approximate three contributions to I_H. Additionally, we have the hypothesis that we can decrease the error of the approximation in two ways: First, we can use a larger sample without increasing excessively the running time (compared with MOEAs that use the traditional competition scheme in which we need to know the contribution of all individuals). And, second, the probability of deleting the correct individual (i.e., the individual with the lowest contribution) is greater in this case, than if we use the traditional competition scheme because we only deal with three errors and not with P errors, where P is the population size. As we will see later on, our results indicate that our proposed selection mechanism obtains better results than those which approximate the hypervolume and do not consider its locality property.

The remainder of this paper is organized as follows. Section 2 states the problem of interest. The hypervolume indicator is defined in Section 3. The previous related work is discussed in Section 4. The selection mechanism based on the hypervolume and its locality property is described in Section 5. Our alternative selection mechanism based on the approximation of the contributions to the hypervolume is presented in Section 6. Our experimental validation and the results obtained are shown in Section 7. Finally, we provide our conclusions and some possible paths for future work in Section 8.

2 Problem Statement

The general *multi-objective optimization problem (MOP)* is defined as follows: Find $\mathbf{x}^* = [x_1^*, x_2^*, \ldots, x_n^*]^T$ which optimizes

$$\mathbf{f}(\mathbf{x}) = [f_1(\mathbf{x}), f_2(\mathbf{x}), \ldots, f_k(\mathbf{x})]^T \tag{1}$$

such that $\mathbf{x}^* \in \Omega$, where $\Omega \subset \mathbb{R}^n$ defines the feasible region of the problem. Assuming minimization problems, we have the following definitions.

Definition 1 *We say that a vector $\mathbf{x} = [x_1, \ldots, x_n]^T$ dominates vector $\mathbf{y} = [y_1, \ldots, y_n]^T$, denoted by $\mathbf{x} \prec \mathbf{y}$, if and only if $f_i(\mathbf{x}) \leq f_i(\mathbf{y})$ for all $i \in \{1, ..., k\}$ and there exists an $i \in \{1, \ldots, k\}$ such that $f_i(\mathbf{x}) < f_i(\mathbf{y})$.*

Definition 2 *We say that a vector $\mathbf{x} = [x_1, \ldots, x_n]^T$ weakly dominates vector $\mathbf{y} = [y_1, \ldots, y_n]^T$, denoted by $\mathbf{x} \preceq \mathbf{y}$, if $\mathbf{f}(\mathbf{x})$ is not worse than $\mathbf{f}(\mathbf{y})$ in all objectives.*

[1] I_H cannot be computed exactly in polynomial time in the number of objective functions unless $P = NP$.

Definition 3 *A point* $\mathbf{x}^* \in \Omega$ *is Pareto optimal if there does not exist any* $\mathbf{x} \in \Omega$ *such that* $\mathbf{x} \prec \mathbf{x}^*$.

Definition 4 *For a given MOP,* $\mathbf{f}(\mathbf{x})$*, the Pareto optimal set is defined as:* $\mathcal{P}^* = \{\mathbf{x} \in \Omega | \neg\exists \mathbf{y} \in \Omega : \mathbf{y} \prec \mathbf{x}\}$.

Definition 5 *Let* $\mathbf{f}(\mathbf{x})$ *be a given MOP and* $\mathcal{P}^*$ *the Pareto optimal set. Then, the Pareto Front is defined as:* $\mathcal{PF}^* = \{\mathbf{f}(\mathbf{x}) \mid \mathbf{x} \in \mathcal{P}^*\}$.

Definition 6 *An* ***approximation of the Pareto optimal set*** *is a subset of* Ω *composed of mutually nondominated vectors (e.g.,* $\mathcal{A} \subseteq \Omega$ *such that for any two vectors* $\mathbf{x}, \mathbf{y} \in \mathcal{A}$ *is true that* $\mathbf{x} \nprec \mathbf{y}$ *and* $\mathbf{y} \nprec \mathbf{x}$*).*

3 Hypervolume indicator

The hypervolume indicator (I_H) was originally proposed by Zitzler and Thiele in [31], and it's defined as the size of the space covered by the Pareto optimal solutions. I_H is a "Pareto Compliant" indicator.[2]

If Λ denotes the Lebesgue measure, I_H is defined as:

$$I_H(\mathcal{A}, \mathbf{y}_{ref}) = \Lambda\left(\bigcup_{\mathbf{y}\in\mathcal{A}} \{\mathbf{x} \mid \mathbf{y} \prec \mathbf{x} \prec \mathbf{y}_{ref}\}\right) \quad (2)$$

where $\mathcal{A}$ is the approximation of the Pareto optimal set and $\mathbf{y}_{ref} \in \mathbb{R}^k$ denotes a reference point which should be dominated by all possible points.

The contribution to the hypervolume of a solution $\mathbf{x}$ is defined as:

$$C_H(\mathbf{x}, \mathcal{A}) = I_H(\mathcal{A}, \mathbf{y}_{ref}) - I_H(\mathcal{A} \setminus \mathbf{x}, \mathbf{y}_{ref}) \quad (3)$$

where $\mathbf{x} \in \mathcal{A}$. Then, the contribution of $\mathbf{x}$ is the space that is only covered by $\mathbf{x}$. See Figure 1.

Auger et al. [2] conducted a study about the optimal μ-distributions and the choice of the reference point in the hypervolume indicator. They mentioned one interesting property of this indicator when $k = 2$ (two objective functions), called **locality** which says: *given three consecutive points on the Pareto front, moving the middle point will only affect the hypervolume contribution that is solely dedicated to this point, but the joint hypervolume contribution of the other points remains fixed*. See Figure 2. Also, Auger et al. conducted a similar study for $k = 3$ in [1] and they mentioned that the optimal placement of a single solution is not determined by only two neighbors, anymore, as it is the case for $k = 2$, since in this case, all solutions can have an influence on the optimal placement of one point.

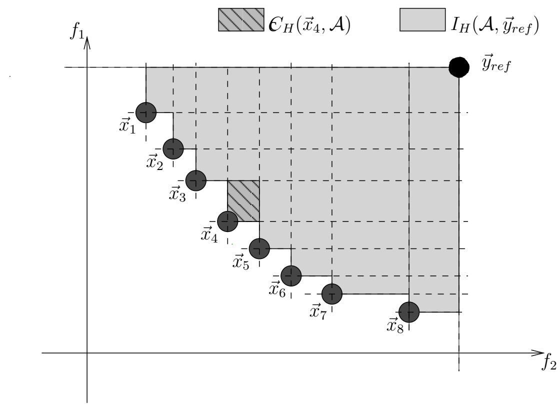


Fig. 1 Let $\mathcal{A} = \{\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_8\}$ be the approximation of the Pareto optimal set and $\mathbf{y}_{ref}$ be the reference point. Then, the gray area is the hypervolume of set $\mathcal{A}$ and the hatched area is the contribution to the hypervolume of the solution $\mathbf{x}_4$.

4 Previous Related Work

In recent years, there have been several proposals to incorporate the hypervolume into a MOEA. However, in most cases, the MOEAs use the same competition scheme: if we have a population $\mathcal{P}$ and a new individual $\mathbf{x}_{new}$, we calculate the contribution to the hypervolume of each individual in $\mathcal{P}$ and the contribution of the new individual. If $\mathbf{x}_{new}$ is better than $\mathbf{x}_{worst}$ (according to the contribution), $\mathbf{x}_{new}$ replaces $\mathbf{x}_{worst}$. Otherwise, the population remains the same. Some of these proposals are the following:

- Knowles and Corne [24] used a bounded archive to save the nondominated solutions found at each generation. When the archive was full and Pareto dominance could no longer discard solutions, they proposed to use the above competition scheme.
- Huband et al. [20] have used the hypervolume with an evolution strategy. They used Pareto ranking as the primary selection criterion and the hypervolume as a second selection criterion (in the same way as described before). However, it is important to mention that the authors used exact calculations of the contribution to I_H only for MOPs with two objective functions.[3]
- Emmerich et al. [15] proposed an algorithm based on NSGA-II and the archived strategies proposed by Knowles, Corne and Fleisher. They called it "SMS-EMOA". SMS-EMOA creates an initial population and then, it generates only one solution by iteration using the operators (crossover and mutation)

[2] An indicator $I : \Omega \to \mathbb{R}$ is **Pareto compliant** if for all $\mathcal{A}, \mathcal{B} \subseteq \Omega : \mathcal{A} \preceq \mathcal{B} \Rightarrow I(\mathcal{A}) \geq I(\mathcal{B})$ assuming that greater indicator values correspond to higher quality, where $\mathcal{A}$ and $\mathcal{B}$ are approximations of the Pareto optimal set, Ω is the feasible region and $\mathcal{A} \preceq B$ means that every point $\mathbf{b} \in \mathcal{B}$ is weakly dominated by at least one point $\mathbf{a} \in \mathcal{A}$.

[3] Given a nondominated front of individuals, the hypervolume value for an individual i is equal to the product of the one-dimensional lengths to the next worse objective function value in the front for each objective.

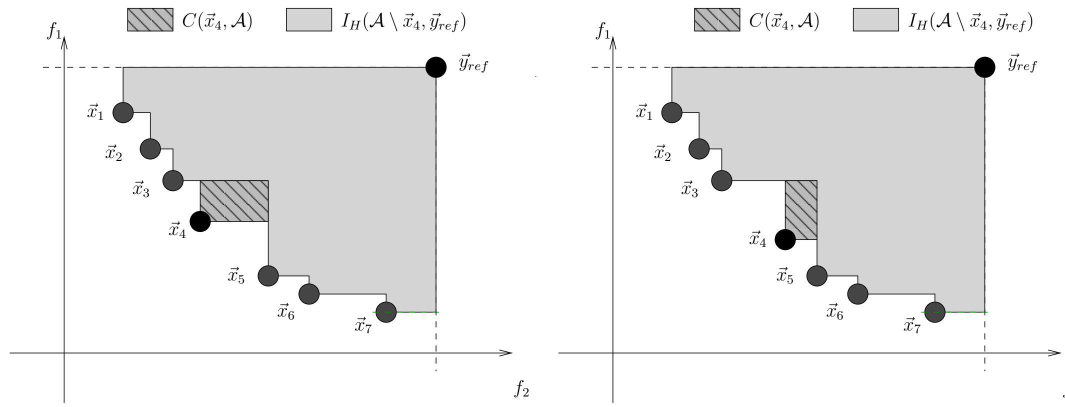


Fig. 2 Let $\mathcal{A} = \{\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_7\}$ be the approximation of the Pareto optimal set. If we move $\mathbf{x}_4$ between $\mathbf{x}_3$ and $\mathbf{x}_5$, the covered space by $\{\mathcal{A} \setminus \mathbf{x}_4\}$ is not affected and only the contribution to the hypervolume of $\mathbf{x}_4$ is affected.

of the NSGA-II. After that, it applies Pareto ranking. When the last front has more than one solution, SMS-EMOA uses the above competition scheme to decide which solution will be removed. Beume et al. [4] proposed not to use the contribution to the hypervolume indicator when, after applying the Pareto ranking procedure we obtain more than one front. In that case, they proposed to use the number of solutions which dominate one solution (the solution that is dominated by more solutions is removed). The authors argue that the motivation for using the hypervolume indicator is to improve the distribution in the nondominated front and then it is not necessary in fronts different to the nondominated front.

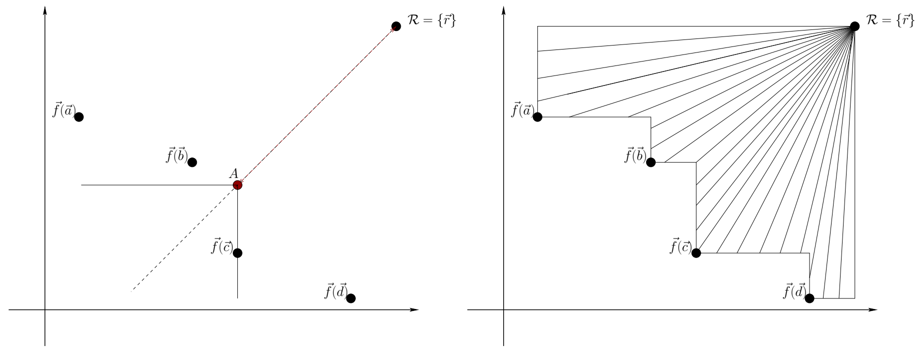
- Igel et al. [21] have used the hypervolume indicator with an evolution strategy. They used Pareto ranking as a primary selection criterion and crowding or hypervolume as a second selection criterion (in the same way as described before).
- Mostaghim et al. [26] designed a MOEA based on particle swarm optimization in which the hypervolume indicator was used in the leader selection mechanism.

If we use eq. (3) in the above competition scheme, we need to calculate $|\mathcal{P}|+1$ contributions to the hypervolume indicator, and therefore, the above algorithms won't be able to deal with MOPs with more than five objective functions (solving a MOP with five objective functions will require several hours using a recent personal computer). In order to address this problem, Bradstreet et al. [5] proposed a method to calculate the contribution to the hypervolume indicator of each solution in a fast way without calculating the hypervolume for each solution. The main idea is the following: when we eliminate one solution of the population, not all the contributions of the other solutions are affected. Emmerich and Fonseca [16] proposed a dimension sweep algorithm for computing all contributions to the hypervolume in three dimensions with a time complexity equal to $O(n \log n)$. Also, they showed that for $k > 3$ (more than three objective functions), the time complexity is bounded below by $\Omega(n \log n)$. However, the calculation of the minimal contribution is an **NP-hard** [7] problem.

Other authors have chosen to approximate the contribution to the hypervolume. For example, Ishibuchi et al. [22] proposed using a number of achievement scalarizing functions with uniformly distributed weight vectors to approximate the hypervolume. They measure the distance from the reference point to the solution set, using scalarizing functions, see Figure 3.

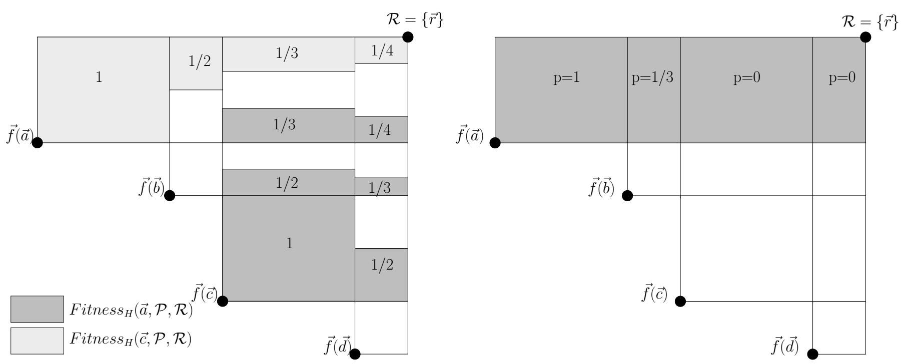
Bader and Zitzler [3] proposed to assign a fitness to each individual using an approximation of the hypervolume based on the idea that is not necessary to know the exact contribution to the hypervolume of each solution, since we only aim to obtain a good ranking of the solutions in the population. The technique that they use to assign fitness to each individual is not easy because they do not consider only the contribution to the hypervolume as we defined in eq. (3), but also all the space dominated by one solution, see Figure 4(a). They used Monte Carlo simulation to approximate the dominated regions and then assign fitness. Also, they proposed a method to remove m individuals from a population $\mathcal{P}$ considering the expected loss in hypervolume that can be attributed to a particular solution when exactly m solutions are removed, see Figure 4(b).

Bringmann and Friedrich [7] indicated that it is not necessary to calculate the hypervolume of all the individuals in the population, in order to know the hypervolume contribution of a single solution. Let $\mathcal{A}$ be the approximation of the Pareto optimal set; they proposed to approximate the contribution of a solution $\mathbf{x} \in \mathcal{A}$ as



(a) (b)

Fig. 3 In (a), the dotted line represents a scalarizing function. Let $\mathcal{A} = \{\mathbf{a}, \mathbf{b}, \mathbf{c}, \mathbf{d}\}$ be the approximation of the Pareto optimal set and $\mathbf{r}$ the reference point. Ishibuchi et al. propose to measure the distance from $\mathbf{r}$ to A. In (b) many uniformly distributed weight vectors are used and the average length is taken as an approximation of the hypervolume.



(a) (b)

Fig. 4 In (a), we illustrate the fitness assignment proposed by Bader and Zitzler. The space dominated by the four solutions ($\mathbf{a}$, $\mathbf{b}$, $\mathbf{c}$ and $\mathbf{d}$) is divided in regions. Suppose that we want to calculate the fitness of individuals $\mathbf{a}$ and $\mathbf{c}$. Regions labeled with number 1 indicate that this portion of the space is only dominated by the solution $\mathbf{a}$ or $\mathbf{c}$. Therefore, this region is attributed to $\mathbf{a}$ or $\mathbf{c}$. Regions labeled with 1/2 indicate that this portion of the space is dominated by two solutions, and then, to each of these two solutions, it corresponds half of this region. Regions with 1/3 and 1/4 indicate that the portion is dominated by three and four solutions, respectively. Thus, for each of them, it corresponds a third or a fourth of this region. (b) shows the probability p that a dominated region is lost if solution $\mathbf{a}$ is removed together with any other solution ($m = 2$). It is interesting to look at the region with probability $p = 1/3$. If we remove solution $\mathbf{a}$, then only solution $\mathbf{b}$ can dominate this region and the probability of choosing $\mathbf{b}$ is 1/3. Regions with probability $p = 0$ indicate that if we remove $\mathbf{a}$ together with any solution, this region is still dominated by one of the remaining solutions.

follows: Let $BB_{\mathbf{x}}$ be the bounding box of $\mathbf{x}$; then, we do a random sampling in $BB_{\mathbf{x}}$. For each random point, we have to check if it is uniquely dominated by $\mathbf{x}$, and then we can approximate the contribution of $\mathbf{x}$ using:

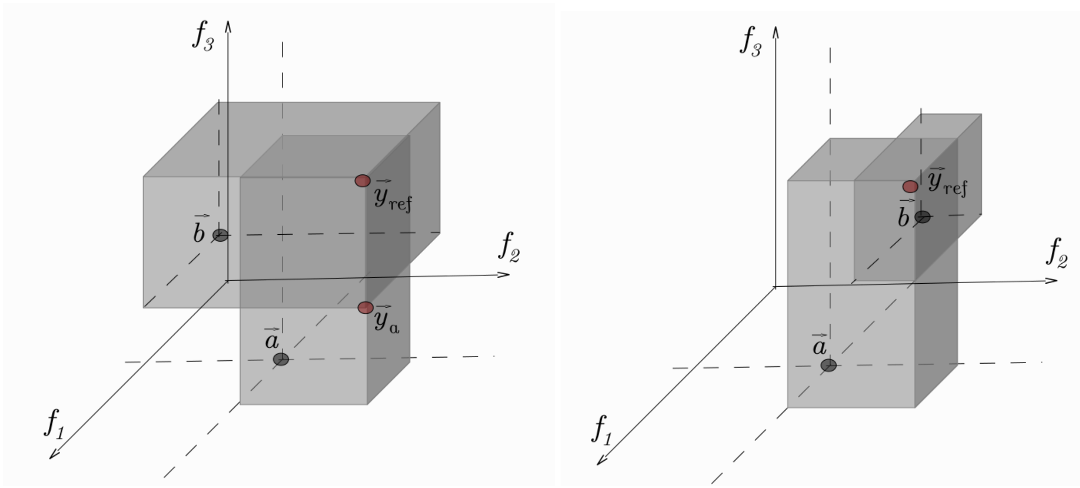
$$\tilde{C}_H(\mathbf{x}, \mathcal{A}) = \frac{SuccessSamples}{Samples} VOL(BB_{\mathbf{x}}) \tag{4}$$

where $SuccessSamples$ is the number of samples that were only dominated by $\mathbf{x}$ (there does not exist another $\mathbf{y} \in \mathcal{A}$ such that $\mathbf{y}$ dominates the sample) and $Samples$ is the total number of samples. See Figure 5. To determine $BB_{\mathbf{x}}$, we construct a bounding box, $B_{\mathbf{y}}$, for each solution $\mathbf{y} \in \mathcal{A}$, using the reference point $\mathbf{y}_{ref}$ as in Figure 5. Then, we can cut $BB_{\mathbf{x}}$ as follows: start with the box $B_{\mathbf{x}}$ itself, iterating over all other boxes $B_{\mathbf{y}}$, such that $\mathbf{x} \neq \mathbf{y}$. If $B_{\mathbf{y}}$ dominates $B_{\mathbf{x}}$ in all but one dimension, then we can cut the bounding box, $B_{\mathbf{x}}$, in the nondominated dimension to obtain $BB_{\mathbf{x}}$. See Figure 6.

5 Selection Mechanism based on Hypervolume and its Locality Property

A new competition scheme for selection mechanisms based on I_H was proposed in [25] and it works as follows: Let's assume that at each iteration of a MOEA, only one solution $\mathbf{x}_{new}$ is created and the current population is $\mathcal{P}$. After that, we calculate the Euclidean distance of the new solution to each solution in the current population:

$$dist_i = \|\mathbf{x}_i - \mathbf{x}_{new}\| \text{ such that } \mathbf{x}_i \in \mathcal{P} \tag{5}$$



(a) (b)

Fig. 6 In (a), we can see that $\mathbf{a}$ is dominated by $\mathbf{b}$ in all objective functions except for f_3. Then, we can cut the bounding box of $\mathbf{a}$ and use $\mathbf{y}_a$ instead of $\mathbf{y}_{ref}$ because the cut region is completely covered by $\mathbf{b}$. In (b), $\mathbf{b}$ is better than $\mathbf{a}$ only in f_1. Therefore, we cannot move the reference point and cut the bounding box because the cut region is not completely covered by $\mathbf{b}$.

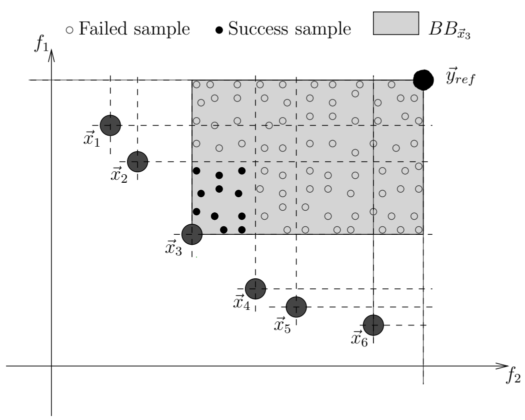


Fig. 5 Let $\mathcal{A} = \{\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3, \mathbf{x}_4, \mathbf{x}_5, \mathbf{x}_6\}$ be the approximation of the Pareto optimal set and $\mathbf{y}_{ref}$ be the reference point. We approximate the contribution of solution $\mathbf{x}_3$ as follows. We construct the bounding box $BB_{\mathbf{x}_3}$ from the reference point $\mathbf{y}_{ref}$. After that, we generate random points in $BB_{\mathbf{x}_3}$. The black points into $BB_{\mathbf{x}_3}$ are success samples; these points are only dominated by $\mathbf{x}_3$. The remaining random points in $BB_{\mathbf{x}_3}$ are also dominated by other points and therefore they are failed samples. Finally, the contribution is approximately $\frac{SuccessSamples}{SuccessSamples+FailedSamples} VOL(BB_{\mathbf{x}_3})$. It is important to mention that this is an example simply for illustrating the procedure by which we can approximate the hypervolume contribution of a solution. However, for two dimensions, we can calculate the exact contribution, when we execute the procedure to cut $BB_{\mathbf{x}_3}$.

and, we choose the nearest solution:

$$\mathbf{x}_{near} \text{ such that } dist_{near} = \min dist_i \tag{6}$$

where $i = \{1, \cdots, |\mathcal{P}|\}$. These two solutions (the new solution, $\mathbf{x}_{new}$, and its nearest neighbor, $\mathbf{x}_{near}$) compete to survive. The core idea is to move a solution within its neighborhood with the aim of improving its contribution to the hypervolume. See Figure 7(a). However, we must consider the case in which the new solution is located in an unexplored region (a region with few solutions) as shown in Figure 7(a). In this case, it is not a good idea to remove the new solution or its nearest neighbor. To address this problem, the authors proposed to choose randomly another solution, $\mathbf{x}_{rand}$. Then, $\mathbf{x}_{rand}$ will also compete with the other two ($\mathbf{x}_{new}$ and $\mathbf{x}_{near}$). This is considering that the probability of choosing a solution in an unexplored region is low. See Figure 7(b).

The experimental results, presented by the authors of this selection scheme, showed that this scheme is a good option to deal with many-objective optimization problems because it is able to reduce the running time significantly without losing quality in the solutions unlike other methods based on the approximation of the hypervolume indicator which have a significant quality loss. However, it is important to note that if we use the competition scheme based on I_H and its locality property, we still have difficulties to solve many-objective optimization problems because although the running time is much lower than that required by MOEAs which use the traditional selection scheme based on I_H , we need to calculate the exact contribution to I_H and this is an NP-hard problem. Also, the authors conducted a study about the randomly-chosen solution in MOPs with $k = 3$. For this, they calculated, at each generation, the number of times in which the random solution is eliminated and considering 30 independent runs, they showed that the random solutions are eliminated at the

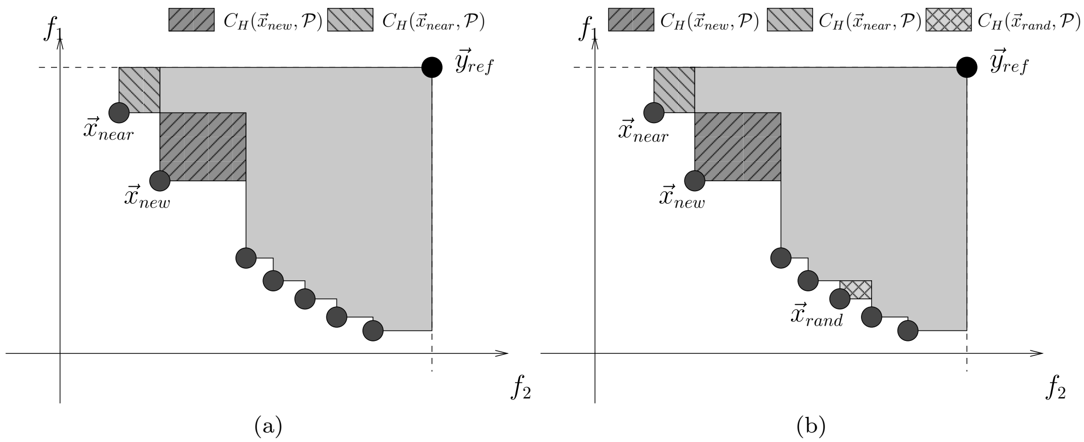


Fig. 7 In (a), $\mathbf{x}_{new}$ competes only with its nearest neighbor $\mathbf{x}_{near}$ and then $\mathbf{x}_{near}$ is eliminated. In (b), we choose $\mathbf{x}_{rand}$ randomly. Then, $\mathbf{x}_{new}$, $\mathbf{x}_{near}$ and $\mathbf{x}_{rand}$ compete to survive and $\mathbf{x}_{rand}$ is eliminated.

beginning of the search process. And, as the search process progresses, the new solution or its nearest neighbor is eliminated more often. From these results, we can see that although for the case in which $k = 3$ the optimal placement of a single solution is not determined by only two neighbors, the competition scheme based on I_H and its locality property still works. This is because this selection scheme does not need to know the entire neighborhood, it only considers to move one solution in the direction corresponding to its nearest neighbor. Therefore, it is not important if the optimal placement of one solution is determined by many (even all) solutions of the population.

6 An alternative selection mechanism based on the approximation of the contributions to the hypervolume

As we mentioned above, there are some proposals to approximate the hypervolume. However, these techniques were incorporated into MOEAs that use the traditional selection scheme in which all individuals compete to survive. In this work, we propose to use the approximation technique proposed by Bringmann and Friedrich in [7] into the selection mechanism proposed by Menchaca and Coello in [25]. There are two main motivations to adopt this approach: First, Bringmann and Friedrich proposed a technique to approximate the contribution to the hypervolume of an individual without having to calculate the hypervolume and the new selection mechanism proposed by Menchaca and Coello is based on the hypervolume contributions. And second, we have the following hypothesis: since the new selection mechanism needs to calculate the contribution of only three individuals, we can reduce the error of the approximation, by increasing the number of samples and this will not increase the running time in an excessive manner. Also, the probability of deleting the individual with the lowest contribution is greater than if we use the traditional competition scheme because if we randomly choose one individual of a set of three individuals, the probability of choosing the worst individual is 1/3. And, if we choose one individual of a set of P individuals, the probability of choosing the worst individual is 1/P. In addition, we only deal with three errors and not with P errors where P is the size of the population. Furthermore, we expect that the contributions of the new solution and its nearest neighbor are different because our idea is to decide if we move the current solution in the population (nearest neighbor of the new solution) to the position of the new solution. Therefore, we expect that the joint contribution of the other solutions is fixed and the contribution of the new solution and its nearest neighbor are different (locality property). We designed an experimental test to validate these last claims. We show experimentally our hypothesis in Section 7. In Algorithm 1, we can see the procedure to approximate the contribution to the hypervolume of one individual in the population. And we can see our alternative selection mechanism in Algorithm 2.

7 Experimental Results

To validate our alternative selection mechanism based on the hypervolume indicator, we incorporated it into the original SMS-EMOA [4] and we called it *"approximate version of improved SMS-EMOA (aviSMS-EMOA)"*. For our experiments, we used problems with up to six objective functions, seven of which were taken from the Deb-Thiele-Laumanns-Zitzler (DTLZ) test suite [14] and seven more were taken from the Walking Fish Group (WFG) toolkit [19]. We used $k = 5$

Input : Current population, $\mathcal{P}_t$, individual, $\mathbf{x}$, reference point, $\mathbf{y}_{ref}$, and the number of samples, $n_{samples}$.
Output: Approximation of the contribution to the hypervolume of individual $\mathbf{x}$, $\tilde{C}_H(\mathbf{x}, \mathcal{P}_t)$.

```
/*Defining the bounding box */
y_box ← y_ref;
foreach x_i ∈ P_t such that x_i ≠ x do
    if x is dominated by x_i in all objective functions
    except in f_k then
        y_box[k] ← x_i[k];
    end
end

/*Calculating the volume of the box BB_x */
volumeBB_x ← 1;
foreach Objective function k do
    volumeBB_x ← (volumeBB_x)(y_box[k] − x[k]);
end

/*Doing sampling */
SuccessSamples ← 0;
for j ← 1 to n_samples do
    Generate a random point x_r, such that x_r ∈ BB_x;
    if Not exists another point x_i ∈ P_t such that x_i
    dominates x_r then
        SuccessSamples ← SuccessSamples + 1;
    end
end

return (SuccessSamples / n_samples)(volumeBB_x);
```

Algorithm 1: Approximating the contribution to the hypervolume of individual $\mathbf{x}$.

Input : Current population, $\mathcal{P}_t$, and the new solution, $\mathbf{x}_{new}$.
Output: The new population, $\mathcal{P}_{t+1}$.

```
/*Calculate the distance of each solution in P_t to
  x_new */
foreach x_i ∈ P_t do
    dist_i ← ||x_i − x_new||;
end

/*Choose the nearest solution to x_new */
x_near | dist_near = min dist_i;

/*Choose one random solution */
Choose randomly x_rand such that x_rand ∈ P_t and
x_rand ≠ x_new;

/*Approximate the contributions to the
  hypervolume, using Algorithm 1 */
C̃_new ← C̃_H(x_new, P_t);
C̃_near ← C̃_H(x_near, P_t);
C̃_rand ← C̃_H(x_rand, P_t);

/*Remove the solution with the worst contribution
  */
x_worst | C̃_worst = min{C̃_new, C̃_near, C̃_rand};
P_{t+1} ← P \ x_worst;
```

Algorithm 2: Alternative selection mechanism based on the hypervolume.

for DTLZ1, DTLZ3 and DTLZ6 and $k = 10$ for the remaining test problems. We used $k_{factor} = 2$ and $l_{factor} = 10$ for the WFG test problems. For each test problem, we performed 30 independent runs. For all algorithms, we adopted the parameters suggested by the authors of NSGA-II: $p_c = 0.9$ (crossover probability), $p_m = 1/n$ (mutation probability), where n is the number of decision variables. For the crossover and mutation operators, we adopted $\eta_c = 15$ and $\eta_m = 20$, respectively. We performed a maximum of 50,000 fitness function evaluations (we used a population size of 100 individuals and we iterated for 500 generations). Only in DTLZ3 we performed 100,000 evaluations (we used a population size of 100 individuals and we iterated for 1000 generations). However, *we adopted four hours as our maximum running time* because we know that the computation of the exact hypervolume contribution has a high computational cost. All MOEAs considered in our experiments were compiled using the GNU C compiler and they were executed on a computer with a 2.66GHz processor and 4GB in RAM.

7.1 Performance Indicators

We adopted I_H to validate our results because it rewards both convergence towards the Pareto front as well as the maximum spread of the solutions obtained. Also, most of the algorithms used in this work have as their aim to maximize the hypervolume and, therefore, it makes sense to use this indicator to assess their performance.[4] To calculate the hypervolume indicator, we normalized the approximations of the Pareto optimal set, generated by the MOEAs, and we used $y_{ref} = [y_1, \cdots, y_k]$ such that $y_i = 1.1$ as our reference point. The normalization was performed considering all approximations generated by the different MOEAs (i.e., we place, in one set, all nondominated solutions found and from this set we calculate the maximum and minimum for each objective function).

Only in some experiments, we consider other two quality indicators. The first one is called "two set coverage (I_{SC})" with the aim of assessing only the convergence of the MOEAs. I_{SC} was proposed by Zitzler et al. [29] and it is a Pareto compliant binary indicator. Let $\mathcal{A}, \mathcal{B}$ two approximations of the Pareto optimal set,

[4] MOEA/D is not based on I_H, but in this case, we also used the "two set coverage" indicator.

I_{SC} is defined as follows:

$$I_{SC}(\mathcal{A},\mathcal{B}) = \frac{|\mathbf{b} \in \mathcal{B} \text{ such that } \exists \mathbf{a} \in \mathcal{A} \text{ with } \mathbf{a} \prec \mathbf{b}|}{|\mathcal{B}|}$$

If all points in $\mathcal{A}$ dominate or are equal to all points in $\mathcal{B}$, then by definition $I_{SC} = 1$. $I_{SC} = 0$ implies that no element in $\mathcal{B}$ is dominanted by any element of $\mathcal{A}$. In general, both $I_{SC}(\mathcal{A},\mathcal{B})$ and $I_{SC}(\mathcal{B},\mathcal{A})$ have to be considered.

The second one is called "inverted generational distance" indicator (I_{IGD}) [10]. I_{IGD} reports how far, on average, $\mathcal{PF}$ is from $\mathcal{A}$, where $\mathcal{PF}$ is the true Pareto front and $\mathcal{A}$ is an approximation of the true Pareto front. I_{IGD} is Pareto non-compliant and it is defined as:

$$I_{IGD}(\mathcal{A}) = \frac{1}{|\mathcal{PF}|}\left(\sum_{i=1}^{|\mathcal{PF}|} d_i^p\right)^{\frac{1}{p}}$$

where $|\mathcal{PF}|$ is the number of vectors in $\mathcal{PF}$, $p = 2$ and d_i is the Euclidean phenotypic distance between each member, i, of $\mathcal{PF}$ and the closest member in $\mathcal{A}$ to that member, i. I_{IGD} measures both convergence and distribution. Since we cannot use the true Pareto front because in most cases we cannot obtain it, we need to use a reference set that provides a reasonably good approximation of the true Pareto front. The result of this indicator depends of this reference set.

7.2 Approximate version of original SMS-EMOA vs approximate version of improved SMS-EMOA

As we saw in the previous section, we propose to approximate the contributions to the hypervolume used by the selection mechanism proposed in [25]. However, this gives rise to the following question: *why don't we approximate the contributions in the original version of the SMS-EMOA?* Our hypothesis is that it is better to use the improved SMS-EMOA for two reasons: First, we need to do sampling for each solution for which we need to know its contribution. Therefore, if we use a large number of samples, the running time drastically increases when employing the traditional selection mechanism adopted by the original SMS-EMOA. This is because in this case, it is required to know the contributions of all the individuals at each iteration, unlike the selection mechanism used by the improved SMS-EMOA that only requires to know the contribution of three individuals per iteration. And second, if we use the improved SMS-EMOA, we can decrease the probability of not choosing the worst individual because we only deal with three errors instead of dealing with P errors (P is the population size). Also, it is important to consider that the selection mechanism used in the improved SMS-EMOA exploits the locality property of the hypervolume. Thus, we estimate that the contributions of the new individual and its nearest neighbor will be different and that the joint contribution of the remaining individuals is fixed.

In order to validate our hypothesis, we compare our "aviSMS-EMOA" with respect to a version of the original SMS-EMOA that approximates the contributions to the hypervolume using Algorithm 1; this version is called "approximate version of the original SMS-EMOA (avoSMS-EMOA)." We ran tests with up to six objective functions because the avoSMS-EMOA algorithm has a manageable running time up to this dimensionality. We used $k(10^3)$ as our number of samples for both algorithms, where k is the number of objective functions (e.g., if we have a MOP with three objective functions, we use $3(10^3) = 3000$ samples). We decided to use this number of samples with the aim that both MOEAs can finish the search or they can execute the largest possible number of generations in the allowable time (we must remember that we adopted four hours as our maximum running time). However, in Section 7.6, we study the behavior of our aviSMS-EMOA with respect to the number of samples. Table 1 shows the results of the DTLZ and WFG test problems with respect to the hypervolume indicator. This table also shows the statistical analysis applied to the experiments using Wilcoxon's rank sum. In this table, we can see that our aviSMS-EMOA obtains better results in most cases, avoSMS-EMOA obtained better results only in nine cases. However, if we check the statistical analysis, we can see that avoSMS-EMOA outperforms our aviSMS-EMOA only in four cases because only in these cases the hypothesis that "medians are equal" can be rejected. Moreover, our aviSMS-EMOA obtains better results than avoSMS-EMOA in forty-seven cases and the hypothesis that "medians are equal" can be rejected in forty-three cases. In summary, our aviSMS-EMOA outperforms avoSMS-EMOA in 43 problems, it is outperformed in 4 problems, and both algorithms obtain similar results in 9 problems. Now, let's check Table 2, which shows the running time required by the two algorithms to obtain the approximation of the Pareto front of each test problem. In this table, we can see that our proposed aviSMS-EMOA is significantly better in all cases (we can say that, on average, it is ten times faster than avoSMS-EMOA).[5] This validates, experi-

[5] It is important to clarify that aviSMS-EMOA is not 33 times faster than avoSMS-EMOA because avoSMS-EMOA does not always calculate 100 contributions to I_H (only when after applying Pareto ranking, a single front is ob-

mentally, the first part of our hypothesis with respect to the running time. For validating the second part, we executed both algorithms, avoSMS-EMOA and aviSMS-EMOA, but we also calculate the exact contribution to the hypervolume and we verify if the algorithm chooses the correct individual to be deleted (worst individual) per iteration. With the aim of calculating the success percentage of each selection mechanism, we consider a success when the algorithm deletes the worst individual. Also, we calculate the percentage in which the new solution and its nearest neighbor had different values in their contribution to the hypervolume because we expect that most of the time these two contributions will be different. For this experiment, we only used problems with three and four objective functions, in order to keep the running times within manageable values. In Table 3, we can see that the alternative selection mechanism used in aviSMS-EMOA achieves a high success rate (above 98% in all cases). This does not happen with avoSMS-EMOA, which cannot even reach a success rate of 1%. Thus, we can conclude that our proposed aviSMS-EMOA is better than avoSMS-EMOA. This is an important result because one of the objectives of this paper is to show that we can significantly improve the current MOEAs based on approximations of I_H if we use the selection mechanism proposed in [25]. Also, we can see that most of the time the new solution and its nearest neighbor had a different contribution (above 95% in most cases).

7.3 HyPE version of SMS-EMOA vs approximate version of improved SMS-EMOA

In this section, we adopt a version of the original SMS-EMOA that uses the fitness assignment scheme proposed in [3] instead of calculating the exact contributions,[6] and which we called "HyPE version of SMS-EMOA (hypeSMS-EMOA)". For hypeSMS-EMOA, we used the source code of HyPE, which is available in the public domain. In both techniques, we used $k(10^3)$ as our number of samples, where k is the number of objective functions. Table 4 shows the results for the DTLZ and WFG test problems, with up to six objective functions, with respect to the hypervolume indicator. This table also presents the statistical analysis of our experiments using Wilcoxon's rank sum. In this table, we can see that our aviSMS-EMOA obtains better results than hypeSMS-EMOA in forty-eight problems and in forty-six of these problems, the hypothesis "medians are equal" can be rejected. Only in eight problems, hypeSMS-EMOA obtains better results than our aviSMS-EMOA and only in four cases, we can say that it outperforms our aviSMS-EMOA because the hypothesis can be rejected. Summarizing, our aviSMS-EMOA outperforms hypeSMS-EMOA in 46 problems, it is outperformed in 4 problems and in 6 problems both algorithms obtain similar results. In Table 5, we can see the running time required by the two algorithms and we can note that hypeSMS-EMOA is better than our aviSMS-EMOA in all cases. However, as we saw in Table 4 the hypeSMS-EMOA algorithm loses quality in its solutions and although our aviSMS-EMOA is slower than hypeSMS-EMOA, its time requirements are still manageable (in the worst case, it requires approximately twenty-four minutes to solve problems with up to six objective functions). In Figures 8, 9 and 10, we can see the Pareto fronts obtained by the algorithms hypeSMS-EMOA, aviSMS-EMOA and the original SMS-EMOA, in the median of thirty independent runs (with respect to the hypervolume indicator) for some of the problems used. In this figure, we can see that hypeSMS-EMOA loses quality in the distribution of the solutions and, in some problems, it cannot even generate the entire Pareto front (for example, in DTLZ6, WFG1 and WFG7). On the other hand, our aviSMS-EMOA obtains a good distribution in all cases, similar to those obtained by the original SMS-EMOA but at a much lower computational cost.

7.4 Approximate version of the improved SMS-EMOA vs the original SMS-EMOA

In this section, we compare our aviSMS-EMOA with respect to the original SMS-EMOA. We tested it only with up to five objective functions because the original SMS-EMOA already exceeds the allowable time in problems with five objective functions. Tables 6, 7 and 8 show that our aviSMS-EMOA is competitive with respect to the original SMS-EMOA. In Table 7, we present the results with respect to the hypervolume indicator and we also present the results of the statistical analysis that we made to validate our experiments, using Wilcoxon's rank sum. Although, in most problems, the original SMS-EMOA obtains better results than our aviSMS-EMOA, the aim of this work was to design a new MOEA based on the approximations of I_H which can significantly reduce the computational cost of MOEAs based on the exact calculation of I_H

tained). Also, aviSMS-EMOA only uses the selection mechanism based on I_H and its locality property when, after applying Pareto ranking, we only obtain a single front. Otherwise, both algorithms use the number of solutions that dominate certain solution as suggested by Beume et al. in [4].

[6] It is important to mention that our aim was to validate the selection mechanism. Therefore, we decided to use the same MOEA in all cases and we only changed the selection mechanism. For this reason, we did not use the original HyPE.

in many-objective optimization problems but without losing much quality. In Table 8, we can see that our algorithm requires at most 24 minutes to solve problems with five objective functions (9.5% of the allowable time), while the original SMS-EMOA spends all the allowable time (four hours) and it is unable to finish the search. It is important to note that our aviSMS-EMOA outperformed the original SMS-EMOA in four problems (DTLZ6, WFG1, WFG4 and WFG7 with five objective functions) in spite of the fact that it requires much less running time.

Finally, in Table 6, we show the results corresponding to the "two set coverage" indicator I_{SC}. To calculate it, we merged all solutions found by our aviSMS-EMOA in a set called $\mathcal{A}$, considering all the 30 independent runs, and we merged all solutions found by the original SMS-EMOA in a set called $\mathcal{B}$. From this table, we can say that in only eleven problems SMS-EMOA covered some solutions generated by aviSMS-EMOA and that aviSMS-EMOA could not cover any solution generated by SMS-EMOA, i.e., in these eleven problems, SMS-EMOA was better than our aviSMS-EMOA in terms of convergence. However, in the remaining thirty-one problems, our aviSMS-EMOA covered some solutions generated by SMS-EMOA and, therefore, we cannot say which algorithm is better. There were no cases in which SMS-EMOA was able to cover all solutions generated by our aviSMS-EMOA and in which aviSMS-EMOA was unable to cover any solution generated by SMS-EMOA. Therefore, we can say that only in 26% of the problems (eleven cases) SMS-EMOA outperforms our aviSMS-EMOA in terms of convergence. In the other 74% of the problems (thirty-one cases) both algorithms had a similar performance in terms of this indicator.

7.5 Approximate version of the improved SMS-EMOA vs MOEA/D

Finally, in this section, we compare our aviSMS-EMOA with respect to another well-known MOEA which is called MOEA/D. We chose this MOEA because it has been an alternative to deal with many-objective optimization problems in recent years and its computational cost is very low. MOEA/D [28] decomposes the MOP into N scalar optimization subproblems and then it solves these subproblems simultaneously using an evolutionary algorithm. For our experiments, we used the version in which MOEA/D adopts PBI (Penalty Boundary Intersection) to decompose the MOP[7] and we generated the convex weights using the technique proposed in [12] and after that, we applied clustering (k-means) to obtain a specific number of weights.

Table 10 shows the results for the DTLZ and WFG test problems, with up to six objective functions, with respect to the hypervolume indicator. This table also presents the statistical analysis of our experiments using Wilcoxon's rank sum. From these results, we can say that our aviSMS-EMOA outperforms MOEA/D in fifty-one problems, it is outperformed by MOEA/D in one problem and both algorithms have a similar behavior in the remaining four problems. Table 9 shows that our aviSMS-EMOA was able to cover a big percentage of the solutions generated by MOEA/D in some problems and MOEA/D did not cover any solutions generated by our aviSMS-EMOA in many cases. Then, we can say that the convergence of our aviSMS-EMOA is better than the convergence of MOEA/D. Table 11 shows the results with respect to I_{IGD} in problems with three objective functions,[8] the reference sets that we adopted were taken from [11]. In this table, we can see that our aviSMS-EMOA is better than MOEA/D in most cases (ten out of fourteen) because it obtained a better result according to I_{IGD} and also the statistical analysis says that we can reject the null hypothesis ("medians are equal"). In one case, both MOEAs have a similar behavior because the null hypothesis cannot be rejected and only in three cases MOEA/D outperformed our aviSMS-EMOA. Regarding I_{IGD} and the DTLZ test problems, we can say that MOEA/D is better in MOPs with concave Pareto fronts (DTLZ2, DTLZ3 and DTLZ4), both MOEAs had a similar behavior in a MOP with a linear Pareto front (DTLZ1) and our aviSMS-EMOA is better in MOPs with degenerate Pareto fronts (DTLZ5 and DTLZ6) and with disconnected Pareto fronts (DTLZ7). These results are logical because we know that in linear Pareto fronts both MOEAs converge to a uniformly distribution, therefore, we expect to both MOEAS have a similar behavior; in concave Pareto fronts we know that both MOEAs converge to a different distribution, therefore, we expect to obtain different values in the indicator I_{IGD}. Finally, we know that MOEA/D has difficulties in MOPs with disconnected Pareto fronts or degenerate Pareto fronts, since it uses a set of well distributed weights (in the whole objective space) to guide the search. With respect to the running time required by each MOEA, MOEA/D outperforms our aviSMS-

[7] We decided to use PBI because the resulting optimal solutions with PBI are normally much better distributed than those obtained by the Tchebycheff approach [28].

[8] We decided to use I_{IGD} only in MOPs with three objective functions because the results of the indicator depend of the reference set that we use and we know that generating a good reference set in MOPs with many objective functions is a difficult task.

EMOA because it only needs a maximum of one second to solve MOPs with 6 objective functions while aviSMS-EMOA needs twenty-two minutes.

Finally, we present a brief study on the effect of the population size on the performance of these two MOEAs. It is normally assumed that if we increase the number of objective functions, we should increase the population size as well. However, MOEAs based on I_H are not practical when we use large populations because their computational costs increase rapidly (we need to compute more times the contribution to I_H). Our aviSMS-EMOA is more practical in this sense for two reasons: (i) it only needs to calculate three contributions to I_H per iteration regardless of the population size and (ii) it does not compute the exact contributions to I_H, it only approximates them. For our study, we only used the DTLZ2 test problem with 3, 4 and 5 objective functions and we used a population size equal to 300, 350 and 400 individuals, respectively. Table 12 shows the results. In (a), we can see that our aviSMS-EMOA is better than MOEA/D regarding I_H because it obtains better results and we can also reject the null hypothesis in all three cases. In (b), we can see that our aviSMS-EMOA is better than MOEA/D in terms of convergence in two cases because aviSMS-EMOA was able to cover some solutions found by MOEA/D and MOEA/D could not cover any solution found by aviSMS-EMOA. Only in DTLZ2 with three objective functions both MOEAs have a similar behavior because MOEA/D was able to cover some solutions found by aviSMS-EMOA and, therefore, we cannot say if one of these MOEAs is better. With respect to the running time, we can see in (c) that MOEA/D is much faster than our aviSMS-EMOA because it only needs two seconds to solve problems with 3, 4 or 5 objective functions while our aviSMS-EMOA consumes all the allowable running time (4 hours). Although MOEA/D is very fast, it is important to keep in mind that MOEA/D needs to generate a well-distributed set of convex weights and this task is not easy when we increase the number of objective functions.

7.6 Study: aviSMS-EMOA and the number of samples

As we mentioned in Section 4, Bringmann and Friedrich proposed a method to approximate the contribution to I_H of one solution. However, their main goal was to find the solution from a set of solutions with the least contribution to I_H. Thus, they present a way in which we can determine the number of samples to guarantee that for any given δ and $\epsilon \geq 0$ the obtained solution is with a probability of $(1-\delta)$, larger by at most a factor of $(1+\epsilon)$ than the least contributor. Also, Nowak et al. [27] made an empirical study about the number of samples in the approximation method proposed by Bringmann and Friedrich to find the least contributor. They used $\delta = 10^{-6}$ and $\epsilon = 10^{-2}$. In one of their experiments in which they used 200 points, the number of samples was: 10^4 for points with dimension equal to 3, between 10^4 and 10^5 for points with dimensions equal to 4 and 5, 10^5 for points with dimension equal to 6, between 10^5 and 10^6 for points with dimension equal to 7, between 10^6 and 10^7 for points with dimension equal to 8, 10^7 and 10^8 for points with dimensions equal to 9 and 10, 10^8 for point with dimension equal to 11, 10^9 for points with dimension equal to 12. And, an interesting thing is that for points with dimension greater than 12 the number of samples can significantly decrease, e.g., for points with dimension equal to 90 the number of samples was between 10^4 and 10^5. Some of their conclusions were the following: (i) there is a dependence between the number of samples and the dimension, (ii) the number of samples increases when two or more points differ very little in their contribution (this was called "hardness of approximation" in the original work by Bringmann and Friedrich) and, (iii) the effect "hardness of approximation" seems to be inversely proportional to the dimension when the number of points is fixed. The authors mentioned that this can be attributed to relatively sparse distribution of points as the dimension increases, leading to fewer occurrences of hard cases.

We do not use the complete algorithm proposed by Bringmann and Friedrich in which they want to find the point with the least contribution to I_H from a set of points. Instead, we only use the way in which the contribution is approximated. We decided to conduct a study, in which we use 10^2, 10^3, 10^4 and 10^5 samples to approximate the contribution to I_H in our aviSMS-EMOA and we consider the DTLZ2 test problem with 3, 4, 5, 6, 7, 8, 9 and 10 objective functions. We made our experiments using a population size equal to 100. We decided to use up to 10^5 samples because according to the empirical results presented by Nowak et al. we need at most 10^5 samples for solving problems with dimension greater or equal than 2 and less or equal than 7 and problems with dimension greater or equal than 40 and less or equal than 90 to obtain a good approximation (when $\delta = 10^{-6}$ and $\epsilon = 10^{-2}$ as suggested by Bringmann and Friedrich). And also, for this number of samples our aviSMS-EMOA requires less than four hours (maximum allowable running time) to obtain the approximation of the Pareto front. Therefore, we can think that this is a good number of samples that our aviSMS-EMOA could adopt. As in the above Sections, to calculate the hypervolume indicator, we normalized

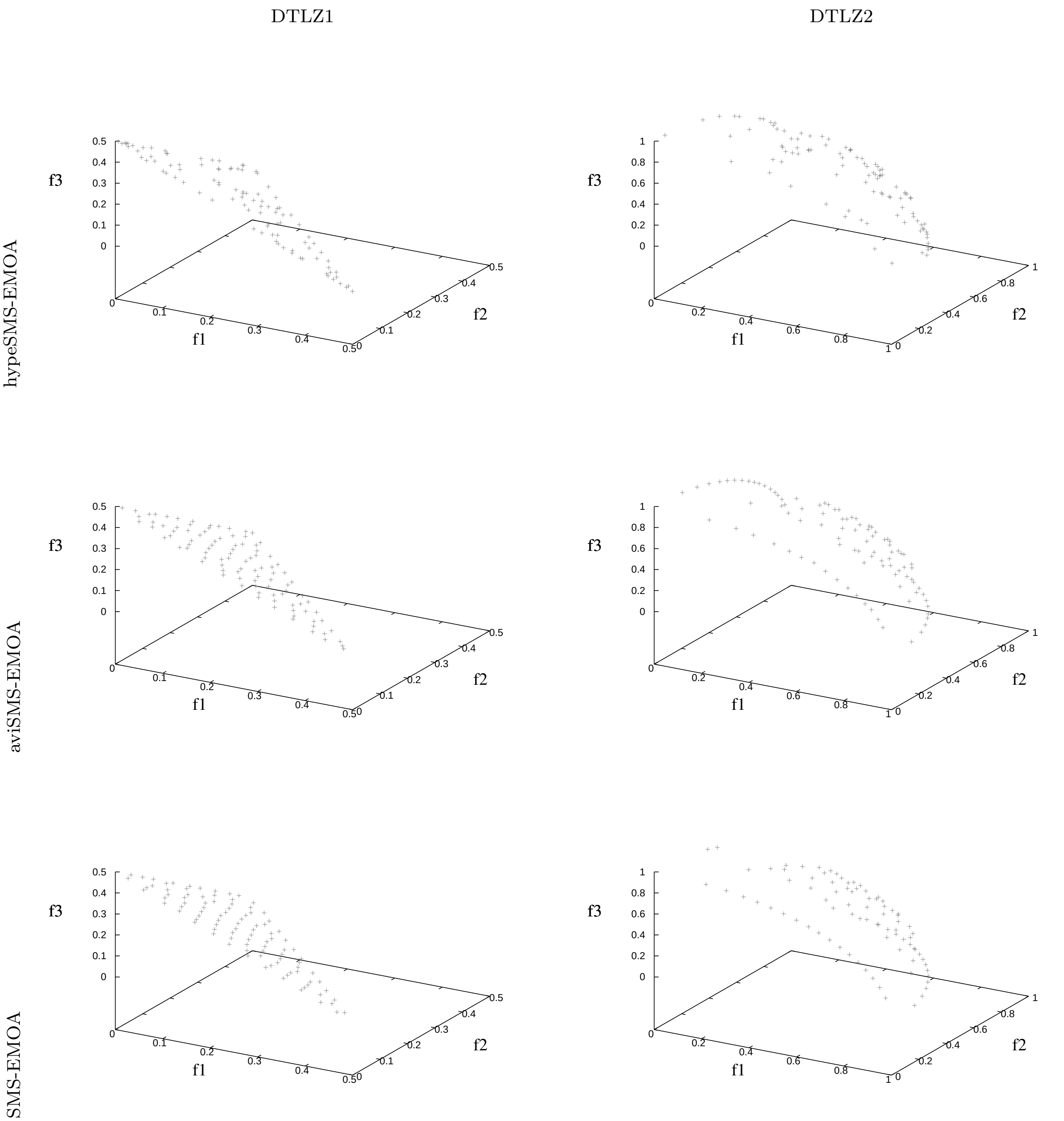


Fig. 8 Pareto fronts obtained by the three algorithms in the median (with respect to the hypervolume indicator) of their independent runs for the test problems DTLZ1 and DTLZ2.

the approximations of the Pareto optimal set, generated by aviSMS-EMOA and we used $y_{ref} = [y_1, \cdots, y_k]$ such that $y_i = 1.1$ as our reference point. The normalization was performed considering all approximations generated by aviSMS-EMOA using a different number of samples (i.e., we place, in one set, all nondominated solutions found and from this set we calculate the maximum and minimum for each objective function). In Table 13, we can see that aviSMS-EMOA significantly improved the quality in its solutions when we increased the number of samples. And, an interesting thing is that in Table 14 we can see that our aviSMS-EMOA is still faster than avoSMS-EMOA (avoSMS-EMOA required 8418 seconds to solve the DTLZ2 test problems with six objective funtions using 6000 samples while aviSMS-EMOA required 3546 seconds to solve the same problem

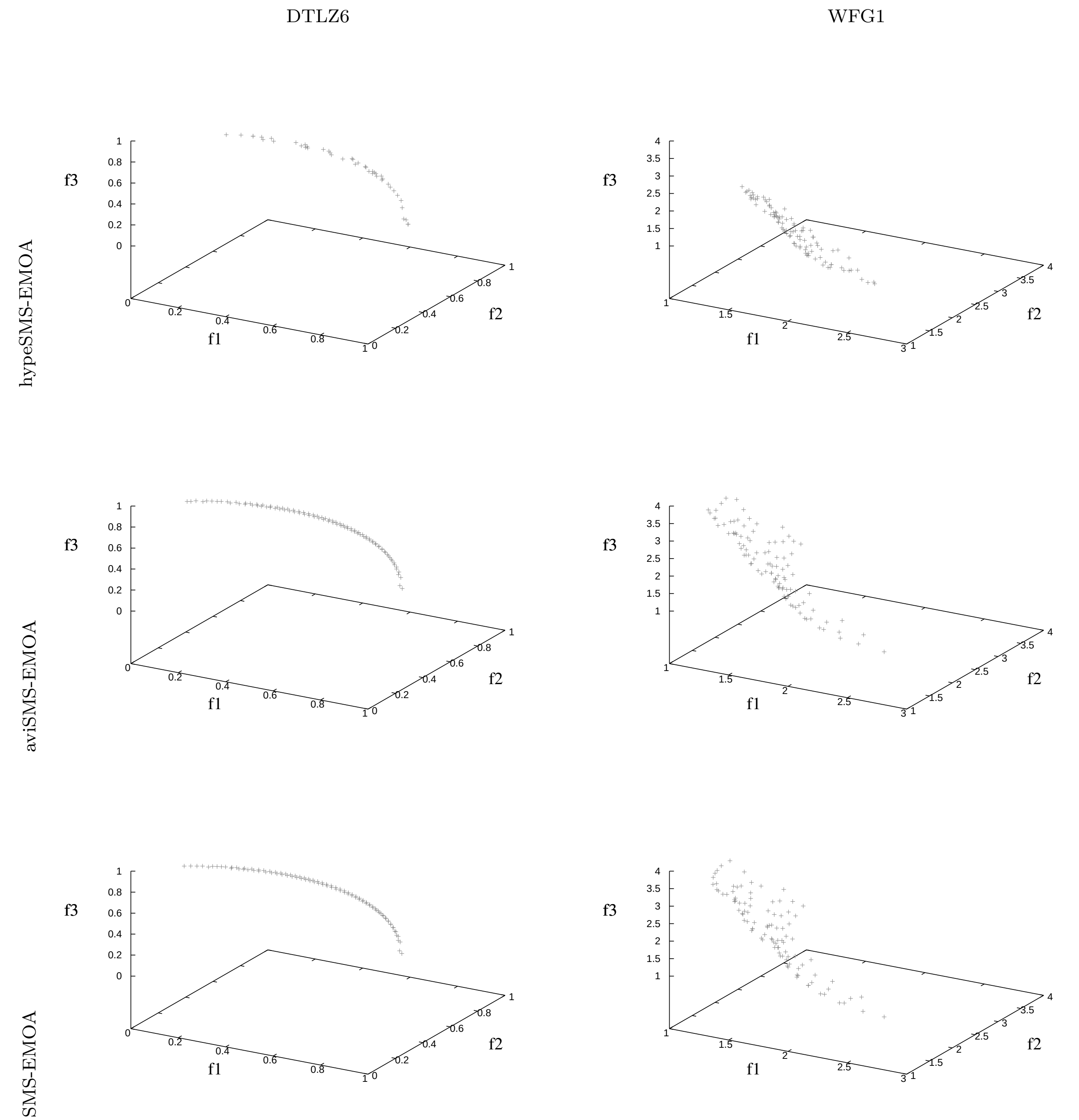


Fig. 9 Pareto fronts obtained by the three algorithms in the median (with respect to the hypervolume indicator) of their independent runs for the test problems DTLZ6 and WFG1.

using 10^5 samples). It is important to mention that the running time decreases as we increase the number of objective functions and this is because aviSMS-EMOA only uses the selection mechanism based on I_H and its locality property when, after applying Pareto ranking, we only obtain one front. Therefore, we can claim that our aviSMS-EMOA is a good option to solve MOPs with low or high dimensionality in objective function space.

8 Conclusions and Future Work

We have studied some MOEAs based on the hypervolume indicator, finding that in most cases, they use a traditional competition scheme. The exception is the im-

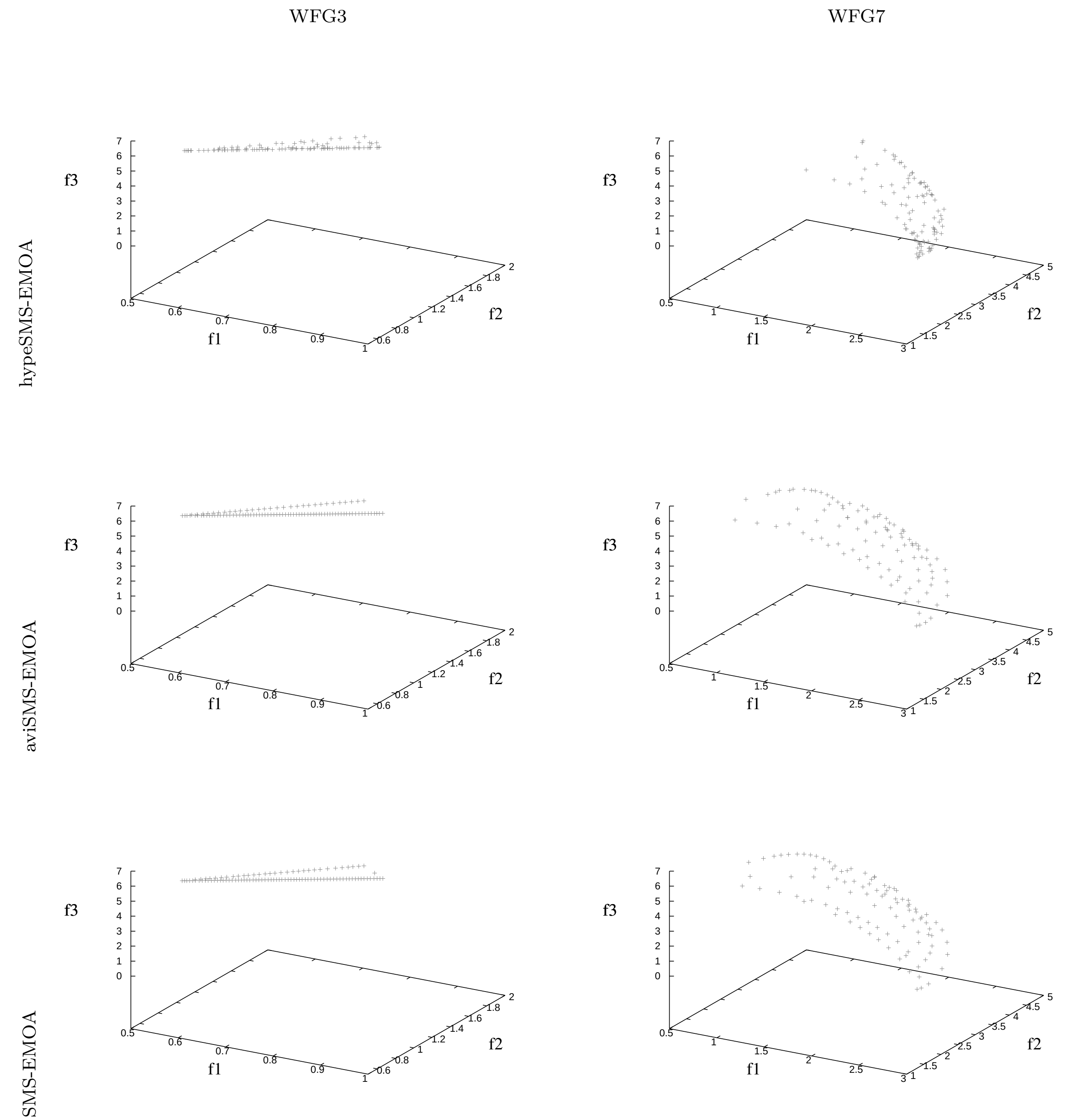


Fig. 10 Pareto fronts obtained by the three algorithms in the median (with respect to the hypervolume indicator) of their independent runs for the test problems WFG3 and WFG7.

proved SMS-EMOA proposed by Menchaca and Coello [25], which uses a different competition scheme that exploits the locality property of the hypervolume. Also, we have studied different techniques to approximate the hypervolume, and we found out that the technique proposed by Bringmann and Friedrich [7] is an excellent choice to be incorporated into the competition scheme of a MOEA that exploits the locality property of the hypervolume. This assumption is based on the following hypothesis: Since the selection mechanism proposed by Menchaca and Coello needs to calculate the contribution of only three individuals, we can reduce the error of the approximation in two ways: (i) by increasing the number of samples without excessively increasing the running time and (ii) by considering that the probability of deleting the individual with the lowest contribution is greater than if we use the traditional competition scheme, because in this case we only deal with three errors and not with P errors (where P is the population size). This hypothesis has been empirically validated in this paper. Our results showed that our proposed selection scheme is a viable alternative for solving MOPs with many objective functions, since it provides reasonably good solutions at a very affordable computational cost.

We also proposed an approximate version of the improved SMS-EMOA, which was called "aviSMS-EMOA". This approach incorporates our proposed selection mechanism into the original SMS-EMOA [4]. We compared our proposed aviSMS-EMOA with respect to different versions of the original SMS-EMOA: avoSMS-EMOA (which uses the technique proposed by Bringmann and Friedrich [7] in a traditional competition scheme), hypeSMS-EMOA (which assigns fitness to each individual in the population, using the technique proposed by Bader and Zitzler [3]) and SMS-EMOA. Also, we compared our aviSMS-EMOA with respect to MOEA/D using PBI. We showed that our proposed aviSMS-EMOA outperforms avoSMS-EMOA, hypeSMS-EMOA and MOEA/D. Moreover, we can say that our aviSMS-EMOA outperforms SMS-EMOA, if we consider both the quality of the approximation of the Pareto front and the computational cost required to obtain that approximation. This is because our proposed aviSMS-EMOA obtains competitive results with respect to SMS-EMOA but at a much lower computational cost. Although with respect to I_H, SMS-EMOA was better than aviSMS-EMOA in most problems, regarding I_{SC}, in thirty-one problems (73% of the total problems), we saw that aviSMS-EMOA was able to generate solutions that no solution found by SMS-EMOA can dominate. Since we cannot say if the nondominated solutions found by SMS-EMOA are better than the nondominated solutions found by aviSMS-EMOA, we claim that they are both competitive in these thirty-one problems. Finally, we also conducted a study about the number of samples that our aviSMS-EMOA should use to increase the quality of the solutions but without exceeding our maximum running time of four hours and we concluded that 10^5 is a good choice. However, we should not forget that aviSMS-EMOA allows us to balance the quality of the solutions and the running time required to obtain them.

As part of our future work, we plan to study other techniques to approximate the contribution of the hypervolume with the aim of reducing even more the running time of our proposed scheme, as well as its approximation error. We are also interested in studying other performance indicators, such as the ϵ indicator [30], which is also Pareto compliant [32]. The aim would be to use the ϵ indicator to select solutions and the hypervolume to distribute them, with the goal of having a hybrid selection scheme that is more effective and efficient than any of the hypervolume-based selection schemes currently available. Finally, we plan to design a version of aviSMS-EMOA which is able to use large population sizes. The idea is to start the search using a small population size and to increase its size over time. In this way, we can obtain more accurate knowledge about the Pareto front in many-objective problems but saving both evaluations of the objective functions and calculations of the contributions to I_H.

Acknowledgements The first author acknowledges support from CONACyT for pursuing graduate studies in Computer Science at CINVESTAV-IPN. The second author acknowledges support from a 2014 Cátedra Marcos Moshinsky in Mathematics and from CONACyT project no. 221551.

References

1. Auger, A., Bader, J., Brockhoff, D.: Theoretically Investigating Optimal μ-Distributions for the Hypervolume Indicator: First Results for Three Objectives. In: R. Schaefer, C. Cotta, J. Kołodziej, G. Rudolph (eds.) Parallel Problem Solving from Nature–PPSN XI, 11th International Conference, Proceedings, Part I, pp. 586–596. Springer, Lecture Notes in Computer Science Vol. 6238, Kraków, Poland (2010)
2. Auger, A., Bader, J., Brockhoff, D., Zitzler, E.: Theory of the Hypervolume Indicator: Optimal $\{\mu\}$-Distributions and the Choice Of The Reference Point. In: FOGA '09: Proceedings of the tenth ACM SIGEVO workshop on Foundations of genetic algorithms, pp. 87–102. ACM, Orlando, Florida, USA (2009)
3. Bader, J., Zitzler, E.: HypE: An Algorithm for Fast Hypervolume-Based Many-Objective Optimization. Evolutionary Computation **19**(1), 45–76 (Spring, 2011)
4. Beume, N., Naujoks, B., Emmerich, M.: SMS-EMOA: Multiobjective selection based on dominated hypervolume. European Journal of Operational Research **181**(3), 1653–1669 (2007)
5. Bradstreet, L., Barone, L., While, L.: Updating Exclusive Hypervolume Contributions Cheaply. In: 2009 IEEE Congress on Evolutionary Computation (CEC'2009), pp. 538–544. IEEE Press, Trondheim, Norway (2009)
6. Bringmann, K., Friedrich, T.: Approximating the volume of unions and intersections of high-dimensional geometric objects. Computational Geometry **43**(67), 601 – 610 (2010)
7. Bringmann, K., Friedrich, T.: Approximating the least hypervolume contributor: NP-hard in general, but fast in practice. Theoretical Computer Science **425**, 104–116 (2012)
8. Bringmann, K., Friedrich, T.: Convergence of Hypervolume-Based Archiving Algorithms II: Competitiveness. In: 2012 Genetic and Evolutionary Computation Conference (GECCO'2012), pp. 457–464. ACM Press, Philadelphia, USA (2012). ISBN: 978-1-4503-1177-9

9. Brockhoff, D., Zitzler, E.: Improving Hypervolume-based Multiobjective Evolutionary Algorithms by Using Objective Reduction Methods. In: 2007 IEEE Congress on Evolutionary Computation (CEC'2007), pp. 2086–2093. IEEE Press, Singapore (2007)
10. Coello Coello, C.A., Cruz Cortés, N.: Solving Multiobjective Optimization Problems using an Artificial Immune System. Genetic Programming and Evolvable Machines **6**(2), 163–190 (2005)
11. Coello Coello, C.A., Lamont, G.B., Van Veldhuizen, D.A.: Evolutionary Algorithms for Solving Multi-Objective Problems, second edn. Springer, New York (2007)
12. Das, I., Dennis, J.E.: Normal-boundary intersection: A new method for generating the pareto surface in nonlinear multicriteria optimization problems. SIAM J. on Optimization **8**(3), 631–657 (1998)
13. Deb, K., Pratap, A., Agarwal, S., Meyarivan, T.: A Fast and Elitist Multiobjective Genetic Algorithm: NSGA–II. IEEE Transactions on Evolutionary Computation **6**(2), 182–197 (2002)
14. Deb, K., Thiele, L., Laumanns, M., Zitzler, E.: Scalable Test Problems for Evolutionary Multiobjective Optimization. In: A. Abraham, L. Jain, R. Goldberg (eds.) Evolutionary Multiobjective Optimization. Theoretical Advances and Applications, pp. 105–145. Springer, USA (2005)
15. Emmerich, M., Beume, N., Naujoks, B.: An EMO Algorithm Using the Hypervolume Measure as Selection Criterion. In: C.A. Coello Coello, A. Hernández Aguirre, E. Zitzler (eds.) Evolutionary Multi-Criterion Optimization. Third International Conference, EMO 2005, pp. 62–76. Springer. Lecture Notes in Computer Science Vol. 3410, Guanajuato, México (2005)
16. Emmerich, M.T., Fonseca, C.M.: Computing Hypervolume Contributions in Low Dimensions: Asymptotically Optimal Algorithm and Complexity Results. In: R.H. Takahashi, K. Deb, E.F. Wanner, S. Grecco (eds.) Evolutionary Multi-Criterion Optimization, 6th International Conference, EMO 2011, pp. 121–135. Springer. Lecture Notes in Computer Science Vol. 6576, Ouro Preto, Brazil (2011)
17. Farina, M., Amato, P.: On the Optimal Solution Definition for Many-criteria Optimization Problems. In: Proceedings of the NAFIPS-FLINT International Conference'2002, pp. 233–238. IEEE Service Center, Piscataway, New Jersey (2002)
18. Fleischer, M.: The Measure of Pareto Optima. Applications to Multi-objective Metaheuristics. In: C.M. Fonseca, P.J. Fleming, E. Zitzler, K. Deb, L. Thiele (eds.) Evolutionary Multi-Criterion Optimization. Second International Conference, EMO 2003, pp. 519–533. Springer. Lecture Notes in Computer Science. Volume 2632, Faro, Portugal (2003)
19. Huband, S., Hingston, P., Barone, L., While, L.: A Review of Multiobjective Test Problems and a Scalable Test Problem Toolkit. IEEE Transactions on Evolutionary Computation **10**(5), 477–506 (2006)
20. Huband, S., Hingston, P., White, L., Barone, L.: An Evolution Strategy with Probabilistic Mutation for Multi-Objective Optimisation. In: Proceedings of the 2003 Congress on Evolutionary Computation (CEC'2003), vol. 3, pp. 2284–2291. IEEE Press, Canberra, Australia (2003)
21. Igel, C., Hansen, N., Roth, S.: Covariance Matrix Adaptation for Multi-objective Optimization. Evolutionary Computation **15**(1), 1–28 (2007)
22. Ishibuchi, H., Tsukamoto, N., Sakane, Y., Nojima, Y.: Hypervolume Approximation Using Achievement Scalarizing Functions for Evolutionary Many-Objective Optimization. In: 2009 IEEE Congress on Evolutionary Computation (CEC'2009), pp. 530–537. IEEE Press, Trondheim, Norway (2009)
23. Ishibuchi, H., Tsukamoto, N., Sakane, Y., Nojima, Y.: Indicator-Based Evolutionary Algorithm with Hypervolume Approximation by Achievement Scalarizing Functions. In: Proceedings of the 12th annual conference on Genetic and Evolutionary Computation (GECCO'2010), pp. 527–534. ACM Press, Portland, Oregon, USA (2010). ISBN 978-1-4503-0072-8
24. Knowles, J., Corne, D.: Properties of an Adaptive Archiving Algorithm for Storing Nondominated Vectors. IEEE Transactions on Evolutionary Computation **7**(2), 100–116 (2003)
25. Menchaca-Mendez, A., Coello Coello, C.A.: A New Selection Mechanism Based on Hypervolume and its Locality Property. In: 2013 IEEE Congress on Evolutionary Computation (CEC'2013), pp. 924–931. IEEE Press, Cancún, México (2013)
26. Mostaghim, S., Branke, J., Schmeck, H.: Multi-Objective Particle Swarm Optimization on Computer Grids. In: D. Thierens (ed.) 2007 Genetic and Evolutionary Computation Conference (GECCO'2007), vol. 1, pp. 869–875. ACM Press, London, UK (2007)
27. Nowak, K., Märtens, M., Izzo, D.: Empirical Performance of the Approximation of the Least Hypervolume Contributor. In: T. Bartz-Beielstein, J. Branke, B. Filipič, J. Smith (eds.) Parallel Problem Solving from Nature PPSN XIII, 13th International Conference, pp. 662–671. Springer. Lecture Notes in Computer Science Vol. 8672, Ljubljana, Slovenia (2014)
28. Zhang, Q., Li, H.: MOEA/D: A Multiobjective Evolutionary Algorithm Based on Decomposition. IEEE Transactions on Evolutionary Computation **11**(6), 712–731 (2007)
29. Zitzler, E., Deb, K., Thiele, L.: Comparison of Multiobjective Evolutionary Algorithms: Empirical Results. Evolutionary Computation **8**(2), 173–195 (2000)
30. Zitzler, E., Künzli, S.: Indicator-based Selection in Multiobjective Search. In: X.Y. et al. (ed.) Parallel Problem Solving from Nature - PPSN VIII, pp. 832–842. Springer-Verlag. Lecture Notes in Computer Science Vol. 3242, Birmingham, UK (2004)
31. Zitzler, E., Thiele, L.: Multiobjective Optimization Using Evolutionary Algorithms—A Comparative Study. In: A.E. Eiben (ed.) Parallel Problem Solving from Nature V, pp. 292–301. Springer-Verlag, Amsterdam (1998)
32. Zitzler, E., Thiele, L., Laumanns, M., Fonseca, C.M., da Fonseca, V.G.: Performance Assessment of Multiobjective Optimizers: An Analysis and Review. IEEE Transactions on Evolutionary Computation **7**(2), 117–132 (2003)

f	avo SMS-EMOA I_H	avi SMS-EMOA I_H	$P(H)$	f	avo SMS-EMOA I_H	avi SMS-EMOA I_H	$P(H)$
DTLZ1 (3)	1.119355 (0.000752)	**1.122217 (0.000494)**	**0.000 (1)**	WFG1 (3)	**1.213067 (0.017999)**	1.205361 (0.024067)	0.067 (0)
DTLZ2 (3)	0.752558 (0.000827)	**0.758602 (0.000340)**	**0.000 (1)**	WFG2 (3)	0.773147 (0.090704)	**0.799503 (0.074819)**	0.318 (0)
DTLZ3 (3)	**1.328649 (0.000456)**	1.328265 (0.000700)	**0.000 (1)**	WFG3 (3)	0.634272 (0.000975)	**0.636772 (0.003130)**	**0.000 (1)**
DTLZ4 (3)	0.869511 (0.000606)	**0.873943 (0.000250)**	**0.000 (1)**	WFG4 (3)	0.745912 (0.001248)	**0.752357 (0.001500)**	**0.000 (1)**
DTLZ5 (3)	0.266479 (0.000073)	**0.266585 (0.000048)**	**0.000 (1)**	WFG5 (3)	0.554767 (0.001288)	**0.557581 (0.001661)**	**0.000 (1)**
DTLZ6 (3)	1.093792 (0.003768)	**1.094423 (0.004341)**	0.739 (0)	WFG6 (3)	0.562236 (0.001805)	**0.565928 (0.001409)**	**0.000 (1)**
DTLZ7 (3)	0.550945 (0.040583)	**0.552256 (0.048770)**	**0.000 (1)**	WFG7 (3)	0.736256 (0.004506)	**0.748956 (0.003767)**	**0.000 (1)**
DTLZ1 (4)	1.359120 (0.003350)	**1.366684 (0.001205)**	**0.000 (1)**	WFG1 (4)	**1.416596 (0.009155)**	1.414339 (0.008604)	0.245 (0)
DTLZ2 (4)	1.020956 (0.002523)	**1.035152 (0.001397)**	**0.000 (1)**	WFG2 (4)	0.001735 (0.008153)	**0.300548 (0.254003)**	**0.000 (1)**
DTLZ3 (4)	1.462183 (0.002832)	**1.463597 (0.000130)**	**0.000 (1)**	WFG3 (4)	0.583248 (0.003331)	**0.592232 (0.006450)**	**0.000 (1)**
DTLZ4 (4)	1.021915 (0.001837)	**1.035100 (0.000920)**	**0.000 (1)**	WFG4 (4)	1.014467 (0.003034)	**1.030276 (0.003034)**	**0.000 (1)**
DTLZ5 (4)	0.542823 (0.000578)	**0.545064 (0.000508)**	**0.000 (1)**	WFG5 (4)	0.591930 (0.001643)	**0.598550 (0.001842)**	**0.000 (1)**
DTLZ6 (4)	1.204105 (0.005438)	**1.206271 (0.004888)**	0.057 (0)	WFG6 (4)	0.589722 (0.007086)	**0.610031 (0.006558)**	**0.000 (1)**
DTLZ7 (4)	**0.558208 (0.045049)**	0.550050 (0.061062)	0.228 (0)	WFG7 (4)	0.901884 (0.007163)	**0.915375 (0.006581)**	**0.000 (1)**
DTLZ1 (5)	0.000157 (0.000847)	**1.541811 (0.005706)**	**0.000 (1)**	WFG1 (5)	**1.548501 (0.006609)**	1.547656 (0.006617)	0.795 (0)
DTLZ2 (5)	1.283019 (0.007232)	**1.300072 (0.004737)**	**0.000 (1)**	WFG2 (5)	0.022475 (0.038981)	**0.407945 (0.205793)**	**0.000 (1)**
DTLZ3 (5)	1.602555 (0.003153)	**1.608407 (0.003140)**	**0.000 (1)**	WFG3 (5)	0.568612 (0.017204)	**0.576101 (0.022876)**	0.093 (0)
DTLZ4 (5)	1.242106 (0.006623)	**1.264793 (0.004457)**	**0.000 (1)**	WFG4 (5)	1.248211 (0.005244)	**1.270178 (0.004494)**	**0.000 (1)**
DTLZ5 (5)	0.928496 (0.001337)	**0.930685 (0.001020)**	**0.000 (1)**	WFG5 (5)	0.644069 (0.003874)	**0.655659 (0.002092)**	**0.000 (1)**
DTLZ6 (5)	**1.519338 (0.002185)**	1.519103 (0.001631)	0.853 (0)	WFG6 (5)	0.516213 (0.037908)	**0.598269 (0.027623)**	**0.000 (1)**
DTLZ7 (5)	0.585090 (0.009424)	**0.589229 (0.020254)**	**0.001 (1)**	WFG7 (5)	1.023308 (0.008889)	**1.032123 (0.012809)**	**0.004 (1)**
DTLZ1 (6)	0.000000 (0.000000)	**1.549649 (0.202423)**	**0.000 (1)**	WFG1 (6)	**1.703600 (0.010057)**	1.691179 (0.014593)	**0.000 (1)**
DTLZ2 (6)	1.643348 (0.006121)	**1.655438 (0.003776)**	**0.000 (1)**	WFG2 (6)	0.017461 (0.036667)	**0.487496 (0.254908)**	**0.000 (1)**
DTLZ3 (6)	1.765905 (0.003950)	**1.770941 (0.000163)**	**0.000 (1)**	WFG3 (6)	0.550745 (0.034929)	**0.590424 (0.049438)**	**0.000 (1)**
DTLZ4 (6)	1.551908 (0.011219)	**1.572087 (0.006064)**	**0.000 (1)**	WFG4 (6)	1.468713 (0.005597)	**1.492717 (0.007779)**	**0.000 (1)**
DTLZ5 (6)	1.023163 (0.001465)	**1.031885 (0.001188)**	**0.000 (1)**	WFG5 (6)	0.700614 (0.003190)	**0.720380 (0.002684)**	**0.000 (1)**
DTLZ6 (6)	**1.634232 (0.003208)**	1.631369 (0.002293)	**0.000 (1)**	WFG6 (6)	0.526582 (0.042342)	**0.637238 (0.050036)**	**0.000 (1)**
DTLZ7 (6)	0.760641 (0.009346)	**0.773352 (0.010180)**	**0.000 (1)**	WFG7 (6)	**1.014516 (0.061564)**	0.907848 (0.066396)	**0.000 (1)**

Table 1 Results obtained in the DTLZ and WFG test problems by avoSMS-EMOA and aviSMS-EMOA, using the hypervolume indicator. We show average values over 30 independent runs. The values in parentheses correspond to the standard deviations. The third column shows the results of the statistical analysis applied to our experiments using Wilcoxon's rank sum. P is the probability of observing the given result (the null hypothesis is true). Small values of P cast doubt on the validity of the null hypothesis. $H = 0$ indicates that the null hypothesis ("medians are equal") cannot be rejected at the 5% level. $H = 1$ indicates that the null hypothesis can be rejected at the 5% level.

f	avo SMS-EMOA time	avi SMS-EMOA time
DTLZ1 (3)	≈ 4593 s	≈ **385 s**
DTLZ2 (3)	≈ 10349 s	≈ **889 s**
DTLZ3 (3)	≈ 14300 s	≈ **1298 s**
DTLZ4 (3)	≈ 10482 s	≈ **888 s**
DTLZ5 (3)	≈ 897 s	≈ **172 s**
DTLZ6 (3)	≈ 5203 s	≈ **409 s**
DTLZ7 (3)	≈ 8469 s	≈ **738 s**
DTLZ1 (4)	≈ 4215 s	≈ **394 s**
DTLZ2 (4)	≈ 12071 s	≈ **961 s**
DTLZ3 (4)	≈ 11419 s	≈ **1399 s**
DTLZ4 (4)	≈ 12062 s	≈ **948 s**
DTLZ5 (4)	≈ 4679 s	≈ **495 s**
DTLZ6 (4)	≈ 5794 s	≈ **567 s**
DTLZ7 (4)	≈ 11439 s	≈ **974 s**
DTLZ1 (5)	≈ 5488 s	≈ **350 s**
DTLZ2 (5)	≈ 9542 s	≈ **798 s**
DTLZ3 (5)	≈ 12807 s	≈ **1346 s**
DTLZ4 (5)	≈ 9558 s	≈ **776 s**
DTLZ5 (5)	≈ 5950 s	≈ **557 s**
DTLZ6 (5)	≈ 6878 s	≈ **649 s**
DTLZ7 (5)	≈ 13077 s	≈ **1135 s**
DTLZ1 (6)	≈ 8364 s	≈ **436 s**
DTLZ2 (6)	≈ 8418 s	≈ **750 s**
DTLZ3 (6)	≈ 14410 s	≈ **1164 s**
DTLZ4 (6)	≈ 8426 s	≈ **726 s**
DTLZ5 (6)	≈ 6913 s	≈ **662 s**
DTLZ6 (6)	≈ 7954 s	≈ **754 s**
DTLZ7 (6)	≈ 13879 s	≈ **1324 s**

f	avo SMS-EMOA time	avi SMS-EMOA time
WFG1 (3)	≈ 13915 s	≈ **1176 s**
WFG2 (3)	≈ 4706 s	≈ **474 s**
WFG3 (3)	≈ 6627 s	≈ **554 s**
WFG4 (3)	≈ 13945 s	≈ **1230 s**
WFG5 (3)	≈ 10697 s	≈ **896 s**
WFG6 (3)	≈ 9574 s	≈ **736 s**
WFG7 (3)	≈ 14415 s	≈ **1351 s**
WFG1 (4)	≈ 11621 s	≈ **882 s**
WFG2 (4)	≈ 6353 s	≈ **693 s**
WFG3 (4)	≈ 7828 s	≈ **628 s**
WFG4 (4)	≈ 14419 s	≈ **1238 s**
WFG5 (4)	≈ 10827 s	≈ **878 s**
WFG6 (4)	≈ 10351 s	≈ **799 s**
WFG7 (4)	≈ 14319 s	≈ **1129 s**
WFG1 (5)	≈ 7432 s	≈ **614 s**
WFG2 (5)	≈ 9955 s	≈ **1007 s**
WFG3 (5)	≈ 9945 s	≈ **815 s**
WFG4 (5)	≈ 13473 s	≈ **1091 s**
WFG5 (5)	≈ 13090 s	≈ **1123 s**
WFG6 (5)	≈ 12392 s	≈ **1000 s**
WFG7 (5)	≈ 12685 s	≈ **992 s**
WFG1 (6)	≈ 5974 s	≈ **558 s**
WFG2 (6)	≈ 13436 s	≈ **1339 s**
WFG3 (6)	≈ 11710 s	≈ **1058 s**
WFG4 (6)	≈ 12737 s	≈ **1042 s**
WFG5 (6)	≈ 14413 s	≈ **1332 s**
WFG6 (6)	≈ 14416 s	≈ **1359 s**
WFG7 (6)	≈ 12346 s	≈ **992 s**

Table 2 Time required by avoSMS-EMOA and aviSMS-EMOA for the test problems adopted. s = seconds. Both algorithms were compiled using the GNU C compiler and they were executed on a computer with a 2.66GHz processor and 4GB in RAM.

f	avo SMS-EMOA success	avi SMS-EMOA success	avi SMS-EMOA diff
DTLZ1 (3)	0.010551 (0.003024)	**0.973352 (0.002994)**	**0.898300 (0.002532)**
DTLZ2 (3)	0.010578 (0.003577)	**0.976261 (0.001304)**	**0.946450 (0.001322)**
DTLZ3 (3)	0.008350 (0.003084)	**0.953463 (0.011867)**	**0.942000 (0.005657)**
DTLZ4 (3)	0.009864 (0.002434)	**0.968088 (0.001255)**	**0.946100 (0.001700)**
DTLZ5 (3)	0.009462 (0.001427)	**0.869048 (0.009573)**	**0.920750 (0.001757)**
DTLZ6 (3)	0.011180 (0.001651)	**0.967014 (0.015320)**	**0.714150 (0.085841)**
DTLZ7 (3)	0.009721 (0.003322)	**0.990437 (0.001576)**	**0.937000 (0.002449)**
DTLZ1 (4)	0.009780 (0.002375)	**0.929687 (0.004549)**	**0.899200 (0.002768)**
DTLZ2 (4)	0.009171 (0.002870)	**0.958048 (0.001183)**	**0.946500 (0.000975)**
DTLZ3 (4)	0.007843 (0.002046)	**0.871917 (0.022897)**	**0.923600 (0.009324)**
DTLZ4 (4)	0.009949 (0.002545)	**0.937088 (0.001944)**	**0.942850 (0.001621)**
DTLZ5 (4)	0.009950 (0.002343)	**0.945747 (0.002642)**	**0.943850 (0.002725)**
DTLZ6 (4)	0.010536 (0.002724)	**0.963123 (0.002079)**	**0.930850 (0.003198)**
DTLZ7 (4)	0.009045 (0.002350)	**0.983524 (0.004796)**	**0.950350 (0.001108)**

f	avo SMS-EMOA success	avi SMS-EMOA success	avi SMS-EMOA diff
WFG1 (3)	0.009368 (0.002253)	**0.981253 (0.001282)**	**0.954100 (0.000768)**
WFG2 (3)	0.010965 (0.002512)	**0.976954 (0.004540)**	**0.930550 (0.005005)**
WFG3 (3)	0.009335 (0.001776)	**0.986898 (0.000819)**	**0.952900 (0.001221)**
WFG4 (3)	0.009588 (0.003385)	**0.997338 (0.000251)**	**0.955500 (0.000742)**
WFG5 (3)	0.009214 (0.003495)	**0.981864 (0.000761)**	**0.958000 (0.001000)**
WFG6 (3)	0.009902 (0.002373)	**0.986261 (0.000806)**	**0.951150 (0.000963)**
WFG7 (3)	0.009955 (0.003873)	**0.986393 (0.000630)**	**0.963250 (0.000887)**
WFG1 (4)	0.008880 (0.003976)	**0.970744 (0.001421)**	**0.953550 (0.001396)**
WFG2 (4)	0.011041 (0.002897)	**0.930375 (0.010810)**	**0.931550 (0.003471)**
WFG3 (4)	0.010296 (0.004187)	**0.979382 (0.001668)**	**0.952700 (0.001453)**
WFG4 (4)	0.009413 (0.002984)	**0.994394 (0.000436)**	**0.959100 (0.000943)**
WFG5 (4)	0.009564 (0.003095)	**0.981475 (0.000773)**	**0.958550 (0.000865)**
WFG6 (4)	0.010444 (0.002686)	**0.971220 (0.002110)**	**0.951650 (0.001152)**
WFG7 (4)	0.008602 (0.004414)	**0.984837 (0.000790)**	**0.961850 (0.000726)**

Table 3 Success rate (column called "success") achieved by both, avoSMS-EMOA and aviSMS-EMOA. Since the two selection mechanisms delete the individual with the worst contribution, we define success when the following occurs: When the algorithm deletes the true worst individual (in order to know which is the true worst individual, we compute the exact contribution). In the case of aviSMS-EMOA, we consider the worst individual among three individuals (new, near and rand). In the case of avoSMS-EMOA, we consider the worst individual among all individuals in the population. The column called "diff" shows the percentage in which the new solution and its nearest neighbor had different values in their contribution to I_H (this column can only be applied to aviSMS-EMOA).

f	hype SMS-EMOA I_H	avi SMS-EMOA I_H	$P(H)$
DTLZ1 (3)	1.101012 (0.006067)	**1.122217 (0.000494)**	**0.000 (1)**
DTLZ2 (3)	0.743232 (0.002097)	**0.758602 (0.000340)**	**0.000 (1)**
DTLZ3 (3)	**1.328533 (0.000169)**	1.328265 (0.000700)	0.487 (0)
DTLZ4 (3)	0.863879 (0.001943)	**0.873943 (0.000250)**	**0.000 (1)**
DTLZ5 (3)	0.265428 (0.000238)	**0.266585 (0.000048)**	**0.000 (1)**
DTLZ6 (3)	1.082554 (0.015066)	**1.094423 (0.004341)**	**0.000 (1)**
DTLZ7 (3)	0.534802 (0.033129)	**0.552256 (0.048770)**	**0.000 (1)**
DTLZ1 (4)	1.258490 (0.056806)	**1.366684 (0.001205)**	**0.000 (1)**
DTLZ2 (4)	1.008560 (0.003414)	**1.035152 (0.001397)**	**0.000 (1)**
DTLZ3 (4)	**1.463624 (0.000067)**	1.463597 (0.000130)	0.695 (0)
DTLZ4 (4)	1.014310 (0.004246)	**1.035100 (0.000920)**	**0.000 (1)**
DTLZ5 (4)	0.518095 (0.004575)	**0.545064 (0.000508)**	**0.000 (1)**
DTLZ6 (4)	1.056882 (0.017816)	**1.206271 (0.004888)**	**0.000 (1)**
DTLZ7 (4)	0.507082 (0.030202)	**0.550050 (0.061062)**	**0.002 (1)**
DTLZ1 (5)	1.237122 (0.348331)	**1.541811 (0.005706)**	**0.000 (1)**
DTLZ2 (5)	1.281626 (0.004608)	**1.300072 (0.004737)**	**0.000 (1)**
DTLZ3 (5)	**1.608858 (0.000222)**	1.608407 (0.003140)	**0.004 (1)**
DTLZ4 (5)	1.254611 (0.005040)	**1.264793 (0.004457)**	**0.000 (1)**
DTLZ5 (5)	0.871195 (0.008001)	**0.930685 (0.001020)**	**0.000 (1)**
DTLZ6 (5)	1.433997 (0.008634)	**1.519103 (0.001631)**	**0.000 (1)**
DTLZ7 (5)	0.473364 (0.056567)	**0.589229 (0.020254)**	**0.000 (1)**
DTLZ1 (6)	1.500805 (0.234972)	**1.549649 (0.202423)**	0.162 (0)
DTLZ2 (6)	**1.658514 (0.002679)**	1.655438 (0.003776)	**0.000 (1)**
DTLZ3 (6)	**1.771045 (0.000047)**	1.770941 (0.000163)	**0.000 (1)**
DTLZ4 (6)	**1.583394 (0.003860)**	1.572087 (0.006064)	**0.000 (1)**
DTLZ5 (6)	0.941583 (0.011495)	**1.031885 (0.001188)**	**0.000 (1)**
DTLZ6 (6)	1.460519 (0.022557)	**1.631369 (0.002293)**	**0.000 (1)**
DTLZ7 (6)	0.450393 (0.124927)	**0.773352 (0.010180)**	**0.000 (1)**

f	hype SMS-EMOA I_H	avi SMS-EMOA I_H	$P(H)$
WFG1 (3)	1.017526 (0.067671)	**1.205361 (0.024067)**	**0.000 (1)**
WFG2 (3)	0.647049 (0.054860)	**0.799503 (0.074819)**	**0.000 (1)**
WFG3 (3)	0.606017 (0.006630)	**0.636772 (0.003130)**	**0.000 (1)**
WFG4 (3)	0.701693 (0.005039)	**0.752357 (0.001500)**	**0.000 (1)**
WFG5 (3)	0.537158 (0.002760)	**0.557581 (0.001661)**	**0.000 (1)**
WFG6 (3)	0.547413 (0.003932)	**0.565928 (0.001409)**	**0.000 (1)**
WFG7 (3)	0.558748 (0.028965)	**0.748956 (0.003767)**	**0.000 (1)**
WFG1 (4)	1.147478 (0.026432)	**1.414339 (0.008604)**	**0.000 (1)**
WFG2 (4)	**0.422296 (0.256235)**	0.300548 (0.254003)	0.084 (0)
WFG3 (4)	0.527767 (0.016347)	**0.592232 (0.006450)**	**0.000 (1)**
WFG4 (4)	0.932772 (0.008365)	**1.030276 (0.003034)**	**0.000 (1)**
WFG5 (4)	0.558728 (0.005313)	**0.598550 (0.001842)**	**0.000 (1)**
WFG6 (4)	0.562704 (0.011569)	**0.610031 (0.006558)**	**0.000 (1)**
WFG7 (4)	0.417145 (0.032298)	**0.915375 (0.006581)**	**0.000 (1)**
WFG1 (5)	1.245126 (0.027633)	**1.547656 (0.006617)**	**0.000 (1)**
WFG2 (5)	**0.496898 (0.219141)**	0.407945 (0.205793)	0.176 (0)
WFG3 (5)	0.424716 (0.034430)	**0.576101 (0.022876)**	**0.000 (1)**
WFG4 (5)	1.116985 (0.018688)	**1.270178 (0.004494)**	**0.000 (1)**
WFG5 (5)	0.564973 (0.012296)	**0.655659 (0.002092)**	**0.000 (1)**
WFG6 (5)	0.437411 (0.035684)	**0.598269 (0.027623)**	**0.000 (1)**
WFG7 (5)	0.309892 (0.022794)	**1.032123 (0.012809)**	**0.000 (1)**
WFG1 (6)	1.356902 (0.031802)	**1.691179 (0.014593)**	**0.000 (1)**
WFG2 (6)	0.369049 (0.247980)	**0.487496 (0.254908)**	0.072 (0)
WFG3 (6)	0.353758 (0.043983)	**0.590424 (0.049438)**	**0.000 (1)**
WFG4 (6)	1.282154 (0.024684)	**1.492717 (0.007779)**	**0.000 (1)**
WFG5 (6)	0.523809 (0.020982)	**0.720380 (0.002684)**	**0.000 (1)**
WFG6 (6)	0.369149 (0.052829)	**0.637238 (0.050036)**	**0.000 (1)**
WFG7 (6)	0.264190 (0.017844)	**0.907848 (0.066396)**	**0.000 (1)**

Table 4 Comparison of the results obtained in the DTLZ and WFG test problems by hypeSMS-EMOA and aviSMS-EMOA, with respect to the hypervolume indicator. We show average values over 30 independent runs. The values in parentheses correspond to the standard deviations. The third column shows the results of the statistical analysis applied to our experiments using Wilcoxon's rank sum. P is the probability of observing the given result (the null hypothesis is true). Small values of P cast doubt on the validity of the null hypothesis. $H = 0$ indicates that the null hypothesis ("medians are equal") cannot be rejected at the 5% level. $H = 1$ indicates that the null hypothesis can be rejected at the 5% level.

f	hype sms-emoa time	avi SMS-EMOA time	**f**	hype sms-emoa time	avi SMS-EMOA time
DTLZ1 (3)	≈ **47 s**	≈ 385 s	WFG1 (3)	≈ **147 s**	≈ 1176 s
DTLZ2 (3)	≈ **106 s**	≈ 889 s	WFG2 (3)	≈ **98 s**	≈ 474 s
DTLZ3 (3)	≈ **135 s**	≈ 1298 s	WFG3 (3)	≈ **148 s**	≈ 554 s
DTLZ4 (3)	≈ **107 s**	≈ 888 s	WFG4 (3)	≈ **107 s**	≈ 1230 s
DTLZ5 (3)	≈ **64 s**	≈ 172 s	WFG5 (3)	≈ **153 s**	≈ 896 s
DTLZ6 (3)	≈ **59 s**	≈ 409 s	WFG6 (3)	≈ **168 s**	≈ 736 s
DTLZ7 (3)	≈ **98 s**	≈ 738 s	WFG7 (3)	≈ **151 s**	≈ 1351 s
DTLZ1 (4)	≈ **59 s**	≈ 394 s	WFG1 (4)	≈ **233 s**	≈ 882 s
DTLZ2 (4)	≈ **156 s**	≈ 961 s	WFG2 (4)	≈ **170 s**	≈ 693 s
DTLZ3 (4)	≈ **165 s**	≈ 1399 s	WFG3 (4)	≈ **247 s**	≈ 628 s
DTLZ4 (4)	≈ **157 s**	≈ 948 s	WFG4 (4)	≈ **157 s**	≈ 1238 s
DTLZ5 (4)	≈ **143 s**	≈ 495 s	WFG5 (4)	≈ **206 s**	≈ 878 s
DTLZ6 (4)	≈ **129 s**	≈ 567 s	WFG6 (4)	≈ **216 s**	≈ 799 s
DTLZ7 (4)	≈ **185 s**	≈ 974 s	WFG7 (4)	≈ **252 s**	≈ 1129 s
DTLZ1 (5)	≈ **79 s**	≈ 350 s	WFG1 (5)	≈ **335 s**	≈ 614 s
DTLZ2 (5)	≈ **188 s**	≈ 798 s	WFG2 (5)	≈ **269 s**	≈ 1007 s
DTLZ3 (5)	≈ **177 s**	≈ 1346 s	WFG3 (5)	≈ **378 s**	≈ 815 s
DTLZ4 (5)	≈ **190 s**	≈ 776 s	WFG4 (5)	≈ **220 s**	≈ 1091 s
DTLZ5 (5)	≈ **229 s**	≈ 557 s	WFG5 (5)	≈ **276 s**	≈ 1123 s
DTLZ6 (5)	≈ **225 s**	≈ 649 s	WFG6 (5)	≈ **274 s**	≈ 1000 s
DTLZ7 (5)	≈ **296 s**	≈ 1135 s	WFG7 (5)	≈ **358 s**	≈ 992 s
DTLZ1 (6)	≈ **98 s**	≈ 436 s	WFG1 (6)	≈ **383 s**	≈ 558 s
DTLZ2 (6)	≈ **233 s**	≈ 750 s	WFG2 (6)	≈ **377 s**	≈ 1339 s
DTLZ3 (6)	≈ **185 s**	≈ 1164 s	WFG3 (6)	≈ **445 s**	≈ 1058 s
DTLZ4 (6)	≈ **234 s**	≈ 726 s	WFG4 (6)	≈ **316 s**	≈ 1042 s
DTLZ5 (6)	≈ **336 s**	≈ 662 s	WFG5 (6)	≈ **246 s**	≈ 1332 s
DTLZ6 (6)	≈ **340 s**	≈ 754 s	WFG6 (6)	≈ **259 s**	≈ 1359 s
DTLZ7 (6)	≈ **377 s**	≈ 1324 s	WFG7 (6)	≈ **408 s**	≈ 992 s

Table 5 Time required by hypeSMS-EMOA and aviSMS-EMOA for the test problems adopted. s = seconds. Both algorithms were compiled using the GNU C compiler and they were executed on a computer with a 2.66GHz processor and 4GB in RAM.

f	$I_{SC}(\mathcal{A},\mathcal{B})$	$I_{SC}(\mathcal{B},\mathcal{A})$	**f**	$I_{SC}(\mathcal{A},\mathcal{B})$	$I_{SC}(\mathcal{B},\mathcal{A})$
DTLZ1 (3)	0.002333	**0.031000**	WFG1 (3)	0.000000	**0.003667**
DTLZ2 (3)	0.000000	**0.063333**	WFG2 (3)	0.426333	**0.634667**
DTLZ3 (3)	0.039333	**0.276000**	WFG3 (3)	**0.379333**	0.186333
DTLZ4 (3)	0.000000	**0.067667**	WFG4 (3)	0.078667	**0.256667**
DTLZ5 (3)	0.002000	**0.087333**	WFG5 (3)	0.004333	**0.061000**
DTLZ6 (3)	0.693667	**0.703333**	WFG6 (3)	0.272333	**0.445333**
DTLZ7 (3)	**0.002333**	0.001667	WFG7 (3)	0.003667	**0.020333**
DTLZ1 (4)	0.000000	**0.013333**	WFG1 (4)	0.000000	**0.001000**
DTLZ2 (4)	0.000000	**0.208000**	WFG2 (4)	0.033000	**0.966000**
DTLZ3 (4)	0.002000	**0.169333**	WFG3 (4)	**0.159333**	0.137333
DTLZ4 (4)	0.000000	**0.192667**	WFG4 (4)	0.027000	**0.228333**
DTLZ5 (4)	0.000667	**0.127333**	WFG5 (4)	0.003333	**0.050667**
DTLZ6 (4)	0.226667	**0.308333**	WFG6 (4)	0.086000	**0.387000**
DTLZ7 (4)	0.000333	**0.002000**	WFG7 (4)	0.001000	**0.002667**
DTLZ1 (5)	0.000000	**0.112667**	WFG1 (5)	**0.000000**	0.000000
DTLZ2 (5)	0.000000	**0.327667**	WFG2 (5)	0.029667	**0.966667**
DTLZ3 (5)	0.091333	**0.191000**	WFG3 (5)	0.106667	**0.207667**
DTLZ4 (5)	0.000000	**0.330667**	WFG4 (5)	**0.419000**	0.000667
DTLZ5 (5)	**0.092333**	0.044000	WFG5 (5)	0.002000	**0.044000**
DTLZ6 (5)	**0.273333**	0.134000	WFG6 (5)	0.023667	**0.393667**
DTLZ7 (5)	0.000000	**0.004667**	WFG7 (5)	**0.000000**	0.000000

Table 6 Results obtained in the DTLZ and WFG test problems by aviSMS-EMOA and SMS-EMOA, using the two set coverage indicator (I_{SC}). In this case, $\mathcal{A}$ is the set composed by all solutions found by aviSMS-EMOA considering all 30 independent runs and $\mathcal{B}$ is the set composed by all solutions found by SMS-EMOA considering all 30 independent runs.

f	avi SMS-EMOA I_H	SMS-EMOA I_H	$P(H)$
DTLZ1 (3)	1.122217 (0.000494)	**1.123180 (0.000283)**	**0.000 (1)**
DTLZ2 (3)	0.758602 (0.000340)	**0.759983 (0.000048)**	**0.000 (1)**
DTLZ3 (3)	**1.328265 (0.000700)**	1.328074 (0.000341)	0.051 (0)
DTLZ4 (3)	0.873943 (0.000250)	**0.875118 (0.000042)**	**0.000 (1)**
DTLZ5 (3)	0.266585 (0.000048)	**0.266762 (0.000021)**	**0.000 (1)**
DTLZ6 (3)	1.094423 (0.004341)	**1.095866 (0.003607)**	0.145 (0)
DTLZ7 (3)	**0.552256 (0.048770)**	0.548923 (0.056148)	**0.045 (1)**
DTLZ1 (4)	1.366684 (0.001205)	**1.373796 (0.000307)**	**0.000 (1)**
DTLZ2 (4)	1.035152 (0.001397)	**1.046741 (0.000063)**	**0.000 (1)**
DTLZ3 (4)	1.463597 (0.000130)	**1.463689 (0.000083)**	**0.000 (1)**
DTLZ4 (4)	1.035100 (0.000920)	**1.044891 (0.000093)**	**0.000 (1)**
DTLZ5 (4)	0.545064 (0.000508)	**0.546139 (0.000159)**	**0.000 (1)**
DTLZ6 (4)	1.206271 (0.004888)	**1.208642 (0.003669)**	**0.003 (1)**
DTLZ7 (4)	0.550050 (0.061062)	**0.579217 (0.046162)**	**0.000 (1)**
DTLZ1 (5)	1.541811 (0.005706)	**1.566729 (0.000759)**	**0.000 (1)**
DTLZ2 (5)	1.300072 (0.004737)	**1.334594 (0.000329)**	**0.000 (1)**
DTLZ3 (5)	1.608407 (0.003140)	**1.609056 (0.000319)**	**0.004 (1)**
DTLZ4 (5)	1.264793 (0.004457)	**1.299259 (0.000224)**	**0.000 (1)**
DTLZ5 (5)	0.930685 (0.001020)	**0.931459 (0.001521)**	**0.000 (1)**
DTLZ6 (5)	**1.519103 (0.001631)**	1.504474 (0.002545)	**0.000 (1)**
DTLZ7 (5)	0.589229 (0.020254)	**0.599077 (0.019488)**	**0.008 (1)**

f	avi SMS-EMOA I_H	SMS-EMOA I_H	$P(H)$
WFG1 (3)	1.205361 (0.024067)	**1.210076 (0.025345)**	**0.029 (1)**
WFG2 (3)	0.799503 (0.074819)	**0.809164 (0.067653)**	0.245 (0)
WFG3 (3)	0.636772 (0.003130)	**0.636873 (0.002070)**	0.646 (0)
WFG4 (3)	0.752357 (0.001500)	**0.754175 (0.001647)**	**0.000 (1)**
WFG5 (3)	0.557581 (0.001661)	**0.557814 (0.001690)**	**0.015 (1)**
WFG6 (3)	0.565928 (0.001409)	**0.567213 (0.001614)**	**0.003 (1)**
WFG7 (3)	0.748956 (0.003767)	**0.750817 (0.003654)**	0.055 (0)
WFG1 (4)	1.414339 (0.008604)	**1.422808 (0.008483)**	**0.000 (1)**
WFG2 (4)	0.300548 (0.254003)	**0.861447 (0.126454)**	**0.000 (1)**
WFG3 (4)	0.592232 (0.006450)	**0.599850 (0.006850)**	**0.000 (1)**
WFG4 (4)	1.030276 (0.003034)	**1.038021 (0.002107)**	**0.000 (1)**
WFG5 (4)	0.598550 (0.001842)	**0.599677 (0.001846)**	**0.000 (1)**
WFG6 (4)	0.610031 (0.006558)	**0.616532 (0.006956)**	**0.000 (1)**
WFG7 (4)	0.915375 (0.006581)	**0.925977 (0.007987)**	**0.000 (1)**
WFG1 (5)	**1.547656 (0.006617)**	1.372422 (0.018408)	**0.000 (1)**
WFG2 (5)	0.407945 (0.205793)	**0.913807 (0.125715)**	**0.000 (1)**
WFG3 (5)	0.576101 (0.022876)	**0.590381 (0.027869)**	**0.000 (1)**
WFG4 (5)	**1.270178 (0.004494)**	1.224230 (0.008055)	**0.000 (1)**
WFG5 (5)	0.655659 (0.002092)	**0.658808 (0.002015)**	**0.000 (1)**
WFG6 (5)	0.598269 (0.027623)	**0.631408 (0.028629)**	**0.000 (1)**
WFG7 (5)	**1.032123 (0.012809)**	0.753463 (0.054363)	**0.000 (1)**

Table 7 Results obtained in the DTLZ and WFG test problems by aviSMS-EMOA and SMS-EMOA, using the hypervolume indicator. We show average values over 30 independent runs. The values in parentheses correspond to the standard deviations. The third column shows the results of the statistical analysis applied to our experiments using Wilcoxon's rank sum. P is the probability of observing the given result (the null hypothesis is true). Small values of P cast doubt on the validity of the null hypothesis. $H = 0$ indicates that the null hypothesis ("medians are equal") cannot be rejected at the 5% level. $H = 1$ indicates that the null hypothesis can be rejected at the 5% level.

f	avi SMS-EMOA time	sms-emoa time	f	avi SMS-EMOA time	sms-emoa time
DTLZ1 (3)	≈ 385 s	≈ **197 s**	WFG1 (3)	≈ 1176 s	≈ **369 s**
DTLZ2 (3)	≈ 889 s	≈ **302 s**	WFG2 (3)	≈ 474 s	≈ **236 s**
DTLZ3 (3)	≈ 1298 s	≈ **502 s**	WFG3 (3)	≈ 554 s	≈ **288 s**
DTLZ4 (3)	≈ 888 s	≈ **303 s**	WFG4 (3)	≈ 1230 s	≈ **340 s**
DTLZ5 (3)	≈ **172 s**	≈ 201 s	WFG5 (3)	≈ 896 s	≈ **342 s**
DTLZ6 (3)	≈ 409 s	≈ **236 s**	WFG6 (3)	≈ 736 s	≈ **298 s**
DTLZ7 (3)	≈ 738 s	≈ **270 s**	WFG7 (3)	≈ 1351 s	≈ **388 s**
DTLZ1 (4)	≈ **394 s**	≈ 1422 s	WFG1 (4)	≈ **882 s**	≈ 3471 s
DTLZ2 (4)	≈ **961 s**	≈ 2527 s	WFG2 (4)	≈ **693 s**	≈ 751 s
DTLZ3 (4)	≈ **1399 s**	≈ 6093 s	WFG3 (4)	≈ **628 s**	≈ 783 s
DTLZ4 (4)	≈ **948 s**	≈ 2589 s	WFG4 (4)	≈ **1238 s**	≈ 2809 s
DTLZ5 (4)	≈ **495 s**	≈ 1695 s	WFG5 (4)	≈ **878 s**	≈ 1067 s
DTLZ6 (4)	≈ **567 s**	≈ 2157 s	WFG6 (4)	≈ **799 s**	≈ 939 s
DTLZ7 (4)	≈ **974 s**	≈ 1402 s	WFG7 (4)	≈ **1129 s**	≈ 2948 s
DTLZ1 (5)	≈ **350 s**	≈ 14431 s	WFG1 (5)	≈ **614 s**	≈ 14463 s
DTLZ2 (5)	≈ **798 s**	≈ 14449 s	WFG2 (5)	≈ **1007 s**	≈ 2486 s
DTLZ3 (5)	≈ **1346 s**	≈ 14474 s	WFG3 (5)	≈ **815 s**	≈ 1424 s
DTLZ4 (5)	≈ **776 s**	≈ 14440 s	WFG4 (5)	≈ **1091 s**	≈ 14456 s
DTLZ5 (5)	≈ **557 s**	≈ 14433 s	WFG5 (5)	≈ **1123 s**	≈ 2742 s
DTLZ6 (5)	≈ **649 s**	≈ 14444 s	WFG6 (5)	≈ **1000 s**	≈ 2738 s
DTLZ7 (5)	≈ **1135 s**	≈ 13256 s	WFG7 (5)	≈ **992 s**	≈ 14445 s

Table 8 Time required by aviSMS-EMOA and SMS-EMOA for the test problems adopted. s = seconds. Both algorithms were compiled using the GNU C compiler and they were executed on a computer with a processor running at 2.66GHz and with 4GB in RAM.

f	$I_{SC}(\mathcal{A},\mathcal{B})$	$I_{SC}(\mathcal{B},\mathcal{A})$	f	$I_{SC}(\mathcal{A},\mathcal{B})$	$I_{SC}(\mathcal{B},\mathcal{A})$
DTLZ1 (3)	**0.036333**	0.000000	WFG1 (3)	**0.001667**	0.000000
DTLZ2 (3)	**0.002667**	0.000333	WFG2 (3)	**1.000000**	0.004000
DTLZ3 (3)	**0.167667**	0.003333	WFG3 (3)	**0.922000**	0.014667
DTLZ4 (3)	**0.001333**	0.000000	WFG4 (3)	**0.989333**	0.000000
DTLZ5 (3)	**0.501667**	0.020333	WFG5 (3)	**0.085333**	0.050000
DTLZ6 (3)	**0.983333**	0.453667	WFG6 (3)	**0.621000**	0.162667
DTLZ7 (3)	**0.501333**	0.000000	WFG7 (3)	**0.165667**	0.002000
DTLZ1 (4)	**0.000000**	0.000000	WFG1 (4)	**0.000000**	0.000000
DTLZ2 (4)	**0.002667**	0.000000	WFG2 (4)	**1.000000**	0.009333
DTLZ3 (4)	**0.060333**	0.004667	WFG3 (4)	**0.865000**	0.005000
DTLZ4 (4)	**0.001667**	0.000000	WFG4 (4)	**0.811667**	0.000000
DTLZ5 (4)	**0.285667**	0.062333	WFG5 (4)	**0.040667**	0.010000
DTLZ6 (4)	**0.811667**	0.057333	WFG6 (4)	**0.571667**	0.014667
DTLZ7 (4)	**0.124667**	0.000000	WFG7 (4)	**0.002667**	0.000000
DTLZ1 (5)	0.000000	**0.000333**	WFG1 (5)	**0.000000**	0.000000
DTLZ2 (5)	**0.002333**	0.000000	WFG2 (5)	**0.993333**	0.020667
DTLZ3 (5)	**0.052000**	0.008667	WFG3 (5)	**0.936333**	0.000000
DTLZ4 (5)	**0.002333**	0.000000	WFG4 (5)	**0.627333**	0.000000
DTLZ5 (5)	**0.159000**	0.055000	WFG5 (5)	**0.007667**	0.000000
DTLZ6 (5)	**0.423667**	0.133333	WFG6 (5)	**0.381333**	0.010667
DTLZ7 (5)	**0.011333**	0.000000	WFG7 (5)	**0.000000**	0.000000
DTLZ1 (6)	0.000000	**0.006000**	WFG1 (6)	**0.000000**	0.000000
DTLZ2 (6)	**0.000333**	0.000000	WFG2 (6)	**1.000000**	0.002333
DTLZ3 (6)	**0.025000**	0.007667	WFG3 (6)	**0.911333**	0.000000
DTLZ4 (6)	**0.000333**	0.000000	WFG4 (6)	**0.500000**	0.010667
DTLZ5 (6)	**0.148667**	0.053000	WFG5 (6)	**0.000000**	0.000000
DTLZ6 (6)	**0.374000**	0.089333	WFG6 (6)	**0.333000**	0.010000
DTLZ7 (6)	**0.000333**	0.000000	WFG7 (6)	**0.000000**	0.000000

Table 9 Results obtained in the DTLZ and WFG test problems by aviSMS-EMOA and MOEA/D, using the two set coverage indicator (I_{SC}). In this case, $\mathcal{A}$ is the set composed by all solutions found by aviSMS-EMOA considering all 30 independent runs and $\mathcal{B}$ is the set composed by all solutions found by MOEA/D considering all 30 independent runs.

f	pbi MOEA/D I_H	avi SMS-EMOA I_H	$P(H)$
DTLZ1 (3)	1.071328 (0.002556)	**1.122217 (0.000494)**	**0.000 (1)**
DTLZ2 (3)	0.718988 (0.000212)	**0.758602 (0.000340)**	**0.000 (1)**
DTLZ3 (3)	1.294000 (0.002000)	**1.328265 (0.000700)**	**0.000 (1)**
DTLZ4 (3)	0.709501 (0.000134)	**0.873943 (0.000250)**	**0.000 (1)**
DTLZ5 (3)	0.246682 (0.000807)	**0.266585 (0.000048)**	**0.000 (1)**
DTLZ6 (3)	0.197818 (0.029819)	**1.094423 (0.004341)**	**0.000 (1)**
DTLZ7 (3)	0.448768 (0.026011)	**0.552256 (0.048770)**	**0.000 (1)**
DTLZ1 (4)	1.311857 (0.003695)	**1.366684 (0.001205)**	**0.000 (1)**
DTLZ2 (4)	0.887228 (0.000914)	**1.035152 (0.001397)**	**0.000 (1)**
DTLZ3 (4)	1.439263 (0.004172)	**1.463597 (0.000130)**	**0.000 (1)**
DTLZ4 (4)	0.878865 (0.001268)	**1.035100 (0.000920)**	**0.000 (1)**
DTLZ5 (4)	0.471816 (0.003958)	**0.545064 (0.000508)**	**0.000 (1)**
DTLZ6 (4)	0.592195 (0.014757)	**1.206271 (0.004888)**	**0.000 (1)**
DTLZ7 (4)	0.337272 (0.008258)	**0.550050 (0.061062)**	0.994 (0)
DTLZ1 (5)	1.506309 (0.008970)	**1.541811 (0.005706)**	**0.000 (1)**
DTLZ2 (5)	0.987833 (0.003838)	**1.300072 (0.004737)**	**0.000 (1)**
DTLZ3 (5)	1.608395 (0.000366)	**1.608407 (0.003140)**	**0.000 (1)**
DTLZ4 (5)	0.982714 (0.003994)	**1.264793 (0.004457)**	**0.000 (1)**
DTLZ5 (5)	0.669155 (0.022872)	**0.930685 (0.001020)**	**0.000 (1)**
DTLZ6 (5)	0.802058 (0.020102)	**1.519103 (0.001631)**	**0.018 (1)**
DTLZ7 (5)	0.075921 (0.070973)	**0.589229 (0.020254)**	0.589 (0)
DTLZ1 (6)	**1.690367 (0.003587)**	1.549649 (0.202423)	**0.000 (1)**
DTLZ2 (6)	0.973263 (0.008725)	**1.655438 (0.003776)**	**0.000 (1)**
DTLZ3 (6)	1.766545 (0.001449)	**1.770941 (0.000163)**	0.661 (0)
DTLZ4 (6)	0.986331 (0.006574)	**1.572087 (0.006064)**	**0.000 (1)**
DTLZ5 (6)	0.585905 (0.017193)	**1.031885 (0.001188)**	**0.000 (1)**
DTLZ6 (6)	0.708188 (0.043127)	**1.631369 (0.002293)**	**0.000 (1)**
DTLZ7 (6)	0.013435 (0.003121)	**0.773352 (0.010180)**	1.000 (0)

f	pbi MOEA/D I_H	avi SMS-EMOA I_H	$P(H)$
WFG1 (3)	0.910507 (0.016598)	**1.205361 (0.024067)**	**0.000 (1)**
WFG2 (3)	0.145574 (0.198499)	**0.799503 (0.074819)**	**0.000 (1)**
WFG3 (3)	0.499214 (0.025639)	**0.636772 (0.003130)**	**0.000 (1)**
WFG4 (3)	0.595609 (0.013100)	**0.752357 (0.001500)**	**0.000 (1)**
WFG5 (3)	0.471079 (0.010426)	**0.557581 (0.001661)**	**0.000 (1)**
WFG6 (3)	0.453757 (0.006661)	**0.565928 (0.001409)**	**0.000 (1)**
WFG7 (3)	0.494583 (0.056148)	**0.748956 (0.003767)**	**0.000 (1)**
WFG1 (4)	1.100204 (0.057651)	**1.414339 (0.008604)**	**0.000 (1)**
WFG2 (4)	0.007223 (0.031709)	**0.300548 (0.254003)**	**0.000 (1)**
WFG3 (4)	0.287483 (0.034365)	**0.592232 (0.006450)**	**0.000 (1)**
WFG4 (4)	0.652634 (0.025612)	**1.030276 (0.003034)**	**0.000 (1)**
WFG5 (4)	0.366984 (0.015366)	**0.598550 (0.001842)**	**0.000 (1)**
WFG6 (4)	0.268060 (0.015468)	**0.610031 (0.006558)**	**0.000 (1)**
WFG7 (4)	0.293433 (0.036496)	**0.915375 (0.006581)**	**0.000 (1)**
WFG1 (5)	1.206775 (0.062432)	**1.547656 (0.006617)**	**0.000 (1)**
WFG2 (5)	0.029223 (0.064926)	**0.407945 (0.205793)**	**0.000 (1)**
WFG3 (5)	0.191112 (0.031531)	**0.576101 (0.022876)**	**0.000 (1)**
WFG4 (5)	0.640835 (0.023757)	**1.270178 (0.004494)**	**0.000 (1)**
WFG5 (5)	0.238371 (0.013922)	**0.655659 (0.002092)**	**0.000 (1)**
WFG6 (5)	0.193513 (0.027379)	**0.598269 (0.027623)**	**0.000 (1)**
WFG7 (5)	0.218223 (0.014294)	**1.032123 (0.012809)**	**0.000 (1)**
WFG1 (6)	1.167832 (0.030511)	**1.691179 (0.014593)**	**0.000 (1)**
WFG2 (6)	0.003634 (0.018966)	**0.487496 (0.254908)**	**0.000 (1)**
WFG3 (6)	0.040186 (0.034593)	**0.590424 (0.049438)**	**0.000 (1)**
WFG4 (6)	0.591344 (0.028666)	**1.492717 (0.007779)**	**0.000 (1)**
WFG5 (6)	0.153001 (0.017135)	**0.720380 (0.002684)**	**0.000 (1)**
WFG6 (6)	0.152732 (0.038473)	**0.637238 (0.050036)**	**0.000 (1)**
WFG7 (6)	0.189978 (0.014303)	**0.907848 (0.066396)**	**0.000 (1)**

Table 10 Results obtained in the DTLZ and WFG test problems by MOEA/D and aviSMS-EMOA, using the hypervolume indicator. We show average values over 30 independent runs. The values in parentheses correspond to the standard deviations. The third column shows the results of the statistical analysis applied to our experiments using Wilcoxon's rank sum. P is the probability of observing the given result (the null hypothesis is true). Small values of P cast doubt on the validity of the null hypothesis. $H = 0$ indicates that the null hypothesis ("medians are equal") cannot be rejected at the 5% level. $H = 1$ indicates that the null hypothesis can be rejected at the 5% level.

f	pbi MOEA/D I_{IGD}	avi SMS-EMOA I_{IGD}	$P(H)$
DTLZ1 (3)	0.0004 (0.000)	0.0004 (0.000)	**0.945 (0)**
DTLZ2 (3)	**0.0009 (0.000)**	0.0012 (0.000)	**0.000 (1)**
DTLZ3 (3)	**0.0009 (0.000)**	0.0038 (0.002)	**0.000 (1)**
DTLZ4 (3)	**0.0009 (0.000)**	0.0012 (0.000)	**0.000 (1)**
DTLZ5 (3)	0.0002 (0.000)	**0.0001 (0.000)**	**0.000 (1)**
DTLZ6 (3)	0.0007 (0.000)	**0.0003 (0.000)**	**0.000 (1)**
DTLZ7 (3)	0.0026 (0.003)	**0.0024 (0.002)**	**0.000 (1)**

f	pbi MOEA/D I_{IGD}	avi SMS-EMOA I_{IGD}	$P(H)$
WFG1 (3)	0.0066 (0.000)	**0.0051 (0.000)**	**0.000 (1)**
WFG2 (3)	0.0138 (0.000)	**0.0124 (0.001)**	**0.000 (1)**
WFG3 (3)	0.0056 (0.000)	**0.0055 (0.000)**	**0.000 (1)**
WFG4 (3)	0.0009 (0.000)	**0.0007 (0.000)**	**0.000 (1)**
WFG5 (3)	0.0022 (0.000)	**0.0021 (0.000)**	**0.000 (1)**
WFG6 (3)	0.0155 (0.000)	**0.0151 (0.000)**	**0.000 (1)**
WFG7 (3)	0.0038 (0.000)	**0.0035 (0.000)**	**0.000 (1)**

Table 11 Results obtained in the DTLZ and WFG test problems with three objective functions by MOEA/D and aviSMS-EMOA, using the inverted generational distance indicator (I_{IGD}). We show average values over 30 independent runs. The values in parentheses correspond to the standard deviations. The third column shows the results of the statistical analysis applied to our experiments using Wilcoxon's rank sum. P is the probability of observing the given result (the null hypothesis is true). Small values of P cast doubt on the validity of the null hypothesis. $H = 0$ indicates that the null hypothesis ("medians are equal") cannot be rejected at the 5% level. $H = 1$ indicates that the null hypothesis can be rejected at the 5% level.

f	$\lvert\mathcal{P}\rvert$	pbi MOEA/D I_H	avi SMS-EMOA I_H	$P(H)$
DTLZ2 (3)	300	0.7678(0.000)	**0.7935 (0.000)**	0.000 (1)
DTLZ2 (4)	350	0.9913(0.001)	**1.1046 (0.001)**	0.000 (1)
DTLZ2 (5)	400	1.2083(0.002)	**1.4338 (0.001)**	0.000 (1)

(a)

$I_{SC}(\mathcal{A},\mathcal{B})$	$I_{SC}(\mathcal{B},\mathcal{A})$
0.000222	**0.000667**
0.000000	**0.001143**
0.000000	**0.001583**

(b)

pbi MOEA/D time	avi SMS-EMOA time
≈**2.38**	≈2935.09
≈**2.36**	≈4384.97
≈**2.74**	≈4551.01

(c)

Table 12 Results obtained in the DTLZ2 test problem by MOEA/D and aviSMS-EMOA. $|\mathcal{P}|$ is the population size. In the case of the hypervolume indicator (I_H), we show average values over 30 independent runs. The values in parentheses correspond to the standard deviations. Also, in the case of I_H, we present the results of the statistical analysis applied to our experiments using Wilcoxon's rank sum. P is the probability of observing the given result (the null hypothesis is true). Small values of P cast doubt on the validity of the null hypothesis. $H = 1$ indicates that the null hypothesis can be rejected at the 5% level. In the case of the two set coverage indicator, all solutions found by our MOEA/D were merged in a set called $\mathcal{A}$, considering all the 30 independent runs, and all solutions found by the original aviSMS-EMOA are merged in a set called $\mathcal{B}$. In the case of running time (*time*), we present the time required by both MOEAs in seconds. Both algorithms were compiled using the GNU C compiler and they were executed on a computer with a processor running at 2.66GHz and with 4GB in RAM.

f	100-samples I_H	1000-samples I_H	10000-samples I_H	100000-samples I_H
DTLZ2 (3)	0.7456 (0.002)	0.7549 (0.001)	0.7577 (0.000)	**0.7581 (0.000)**
DTLZ2 (4)	0.9670 (0.007)	1.0207 (0.003)	1.0405 (0.001)	**1.0451 (0.000)**
DTLZ2 (5)	**1.0448 (0.000)**	**1.0448 (0.000)**	**1.0448 (0.000)**	**1.0448 (0.000)**
DTLZ2 (6)	1.1124 (0.041)	1.4090 (0.019)	1.5153 (0.009)	**1.5588 (0.002)**
DTLZ2 (7)	1.1884 (0.074)	1.6036 (0.024)	1.7477 (0.007)	**1.8043 (0.003)**
DTLZ2 (8)	1.5072 (0.125)	1.8152 (0.043)	1.9893 (0.010)	**2.0498 (0.004)**
DTLZ2 (9)	2.0565 (0.063)	2.0905 (0.046)	2.2676 (0.009)	**2.3135 (0.003)**
DTLZ2 (10)	2.3874 (0.071)	2.3114 (0.074)	2.5323 (0.008)	**2.5683 (0.003)**

Table 13 Results obtained in the DTLZ2 test problem with 3, 4, 5, 6, 7, 8, 9 and 10 objective functions by aviSMS-EMOA using 10^2, 10^3, 10^4 and 10^5 samples to approximate the contribution to I_H. We show average values over 30 independent runs using the hypervolume indicator I_H. The values in parentheses correspond to the standard deviations.

f	100-samples time	1000-samples time	10000-samples time	100000-samples time
DTLZ2 (3)	**82.0507 (5.654)**	178.4597 (3.499)	969.8713 (15.051)	8904.6407 (104.433)
DTLZ2 (4)	**84.6053 (3.655)**	160.5430 (3.103)	792.2043 (10.598)	7464.5580 (127.088)
DTLZ2 (5)	**86.7600 (3.023)**	135.5130 (2.812)	528.9030 (8.302)	5135.6250 (88.684)
DTLZ2 (6)	**88.7860 (2.877)**	126.9493 (2.811)	402.0157 (4.699)	3546.9327 (83.094)
DTLZ2 (7)	**93.7263 (3.195)**	125.3957 (2.989)	367.8517 (3.811)	2914.5407 (66.569)
DTLZ2 (8)	**95.9470 (1.954)**	110.1653 (3.706)	374.0367 (4.304)	2812.7597 (43.800)
DTLZ2 (9)	**98.7967 (3.043)**	112.6773 (2.650)	391.5673 (3.303)	3146.0043 (42.011)
DTLZ2 (10)	**100.9503 (3.005)**	115.7223 (3.212)	414.8363 (4.425)	3311.5910 (37.305)

Table 14 Time required by aviSMS-EMOA for the DTLZ2 test problem with 3, 4, 5, 6, 7, 8, 9 and 10 objective functions, using 10^2, 10^3, 10^4 and 10^5 samples to approximate the contribution to I_H. s = seconds. aviSMS-EMOA was compiled using the GNU C compiler and it was executed on a computer with a processor running at 2.66GHz and with 4GB in RAM.