Analysis of Leader Selection Strategies in a Multi-Objective Particle Swarm Optimizer

Antonio J. Nebro
Dpto. Lenguajes y Ciencias de la Computación
University of Málaga
Spain
Email: antonio@lcc.uma.es

Juan J. Durillo
Institut fur Informatik
University of Innsbruck
Austria
Email: juan@dps.uibk.ac.at

Carlos A. Coello Coello
Depto. de Computación
CINVESTAV-IPN
México
Email: ccoello@cs.cinvestav.mx

***Abstract*—Algorithms based on the Particle Swarm Optimization (PSO) scheme have become popular to solve both single- and multi-objective optimization problems. In this paper, we focus on SMPSO, a PSO designed to cope with this second group of problems. Taking it as our starting point, we analyze different leader selection schemes, which give rise to four new variants of SMPSO. These new versions, along with the original algorithm, are compared using a benchmark composed of 21 problems. Our study reveals that SMPSOhv, a variant that uses the hypervolume indicator to guide leader selection, is the best performing algorithm in our comparison, outperforming also the original version of SMPSO. To further assess the performance of SMPSOhv, we compare it against NSGA-II and SMS-EMOA, achieving again the best overall results in this new comparative study. Based on these observations, we conclude that the use of the hypervolume for leader selection is a promising approach for multi-objective PSO algorithms.**

I. Introduction

Particle Swarm Optimizers (PSO) [1] have been widely used in the last decade because of both their good performance and simplicity. In the field of multi-objective optimization, numerous variants of multi-objective PSO algorithms have been proposed [2]. In our research work, we proposed the so-called Speed-constrained Multi-objective PSO algorithm, or SMPSO [3], which has shown a remarkable performance in terms of different assessment criteria: quality of results [4], convergence towards the optimum solutions [5], and scalability with the problem size [6].

From a high level of abstraction, in a PSO algorithm a set (swarm) of candidate solutions (particles) to the problem navigate through the search space of an optimization problem. This navigation takes place attending to a velocity equation, which rules the way in which particles change their position. Among the factors that govern that velocity equation, two of them can be highlighted: the current position of the particle and the best positions visited so far, also referred to as leaders. Usually, the best position visited by a particle (local leader) and the best particle visited by any particle in the swarm (global leader) are considered. Nevertheless, many other different alternatives have been studied, and the use of different leader selection strategies have been analyzed in a number of works [7][8][9].

While in a single-objective PSO, the best position visited by a particle corresponds to the one having the lowest/highest value of the problem to be minimized/maximized, the selection of these best positions presents additional challenges in multi-objective optimization, where two or more conflicting functions must be optimized simultaneously. The reason is related to the non-dominance concept, that applies to two solutions when none of them improves the other in all the objectives; if so, then one dominates the other. By adopting non-dominance the result is a partial-order relationship, so defining the concept of "best solution" is not so clear as in single-objective optimization. Thus, additional mechanisms have to be considered to properly select the leaders.

Our aim in this work is to evaluate different alternatives for the leader selection scheme of our SMPSO algorithm. Besides the original mechanism included in SMPSO, we analyze three different alternatives: random selection of leaders, selection of leaders based on a neighborhood structure, and selection based on a quality indicator, such as the hypervolume [10]. The use of this last strategy follows the current trend of designing indicator-based techniques [11][12], and it was applied to a PSO algorithm in [13]; our purpose is to adopt this concept in SMPSO. To put the obtained results in a proper context, we have also compared the best performing SMPSO variant with respect to NSGA-II [14] (which is representative of the Pareto-based algorithms) and with respect to SMS-EMOA [12] (which is an indicator-based algorithm which has gained increasing popularity in recent years).

The rest of this work is structured as follows. Section II reviews some previous related work. The proposed versions of the SMPSO algorithm are described in Section III. The methodology applied to carry out our experimental study and the obtained results are detailed in Sections IV and V, respectively. Finally, Section section:conclusions summarizes the paper and provides some possible lines for future work.

II. Previous Related Work

This section is aimed at reviewing the most relevant previous related work. We focus mainly on approaches that provide a particular global leader selection mechanism in multi-objective PSO algorithms.

The first implementation of a multi-objective PSO algorithm was done by Moore *et al.* [15]. That seminal paper did not include, however, any information on how the local or global leader were selected. Since then, many strategies have been proposed. Most of these approaches share the use of an external archive where the non-dominated solutions found during the search are stored. Among them, the differences lie on the number of selected global leaders and in the way in which they are selected.

The most commonly employed strategy has consisted in using only one global leader. In [16] that leader was randomly selected from the archive. The main target of that work was the use on an elitist-mutation-mechanism in combination with PSO and it was not clear whether or not the random selection of leaders had any impact on the final solution. Others works considered the use of a density estimator for selecting leaders from the archive, trying to favor the spread of the computed approximation to the Pareto front. Examples of these indicators are, to mention a few, the crowding distance, used in [17] or in the original SMPSO itself, and a grid structure such as the one adopted in [18]. In [19], Hernández *et al.* criticized that the use of density estimators like the crowding distance tends to benefit solutions placed on the edges of the Pareto front. To mitigate that effect, the authors proposed a mechanism based on the inverse distance to the centroid of the archive content, trying to increase the diversity. In this same line, the so-called sigma-method, proposed by Mostaghim *et al.* [20], considered, for each particle, the closest leader in objective function space. The idea was to direct each particle towards a different area of the Pareto front without any interference of any other leader.

Two interesting leader selection mechanisms are described in [21] and [22]. In the first paper, the leader was computed by applying differential evolution [23] to the contents of the archive. In the second one, besides other evaluated approaches, a method based on a reward mechanism for the points in the archive was proposed. This mechanism consisted on giving a higher priority to those particles which were used as successful leaders in the past. A similar idea was applied in [24], where the particles memorized successful leaders for some iterations; in that proposal, whenever a new particle entered the archive, it was considered as a global leader with the aim of exploring new areas of the search space.

A few approaches considered the use of more than one particle from the archive as global leaders at the same time. In [25], the velocity of a particle was computed by considering all the solutions in the archive that dominate the particle's current local leader. In [26] no local leader was used, and instead, the information of several global leaders was selected.

A number of guide selection schemes and their application to a multi-objetive PSO were analyzed in [13]. One of such schemes is based on the hypervolume quality indicator; our proposal differs in that we use the hypervolume for both leader selection and for density estimator in the archive.

III. Analyzed SMPSO Variants

In this section, we include first some background on PSO. After that, the original SMPSO variant is described. Finally, four versions of SMPSO with different schemes for leader selection are presented.

A. Background on Particle Swarm Optimisation

PSO is a population-based optimization technique which has its inspiration on bird flocking [1]. In its more basic form, each particle in PSO updates its position, $\vec{x}_i$, at the generation t with the formula:

$$\vec{x}_i(t) = \vec{x}_i(t-1) + \vec{v}_i(t) \quad (1)$$

where the factor $\vec{v}_i(t)$ is known as the velocity and is given by

$$\vec{v}_i(t) = w \cdot \vec{v}_i(t-1) + C_1 \cdot r_1 \cdot (\vec{x}_{p_i} - \vec{x}_i) + C_2 \cdot r_2 \cdot (\vec{x}_{g_i} - \vec{x}_i) \quad (2)$$

In this formula, $\vec{x}_{p_i}$ is the best solution that $\vec{x}_i$ has viewed, $\vec{x}_{g_i}$ is the best particle that the entire swarm has viewed, w is the inertia weight of the particle and controls the trade-off between global and local experience, r_1 and r_2 are two uniformly distributed random numbers in the range $[0,1]$, and C_1 and C_2 are specific parameters which control the effect of the personal and global best particles.

In order to control the particle's velocity, instead of using upper and lower parameter values which limit the step size of the velocity, in SMPSO we adopted a *constriction coefficient* (Eq. 3) obtained from the constriction factor χ originally developed by Clerc and Kennedy (Eq. 2) in [27].

$$\chi = \frac{2}{2 - \varphi - \sqrt{\varphi^2 - 4\varphi}} \quad (3)$$

where

$$\varphi = \begin{cases} C_1 + C_2 & \text{if } C_1 + C_2 > 4 \\ 1 & \text{if } C_1 + C_2 \leq 4 \end{cases} \quad (4)$$

We also introduced a mechanism for bounding the accumulated velocity of each variable j (in each particle) by means of the following *velocity constriction* equation:

$$v_{i,j}(t) = \begin{cases} delta_j & \text{if } v_{i,j}(t) > delta_j \\ -delta_j & \text{if } v_{i,j}(t) \leq -delta_j \\ v_{i,j}(t) & \text{otherwise} \end{cases} \quad (5)$$

where

$$delta_j = \frac{(upper\_limit_j - lower\_limit_j)}{2} \quad (6)$$

Summarizing, in SMPSO the velocity of the particles is calculated according to Eq. 2; the resulting velocity is then multiplied by the constriction factor (Eq. 3) and the resulting value is constrained by using Eq. 5.

B. Pseudocode of the SMPSO Algorithm

Algorithm 1 depicts the pseudocode of SMPSO. It starts by initializing the swarm (Line 1). This phase includes the position, velocity, and p (individual best) of the particles. The leaders archive is initialized with the non-dominated solutions in the swarm (Line 2). Then, the main loop of the algorithm is executed for a maximum number of iterations. The velocities and positions of the particles are calculated first (Lines 5 and 6) and a mutation operator is applied with a given probability (Line 7). The resulting particles are evaluated (Line 8) and

Algorithm 1 Pseudocode of SMPSO

```
1: initializeSwarm()
2: initializeLeadersArchive()
3: generation = 0
4: while generation < maxGenerations do
5:     computeSpeed() // Eqs. 2 - 6
6:     updatePosition() // Eq. 1
7:     mutation() // Turbulence
8:     evaluation()
9:     updateLeadersArchive()
10:    updateParticlesMemory()
11:    generation ++
12: end while
13: returnLeadersArchive()
```

both the particle's memory and the leaders archive are updated (Lines 9 and 10). The algorithm returns the leaders archive as the approximation set found (Line 13).

Given that the leaders archive can become full, the crowding distance of NSGA-II is used to decide which particles must remain in it. The turbulence operator is the polynomial mutation operator [28]. To choose the *pbest* particle to apply Eq. 2, two solutions are randomly taken from the leaders archive and the one having the largest crowding distance is selected. This mechanism is intended not only to guide the particles in the swarm towards the area where non-dominated solutions are located, but also to achieve a uniform diversity.

C. SMPSO Variants

Here, we present the different SMPSO variants analyzed in this work. In particular, we focus on three methods for leader selection: random selection of leaders, use of a neighborhood structure, and use of a quality indicator. Using these schemes, we propose four new variants of SMPSO: SMPSO_r, cellSMPSO, SMPSOhv, and SMPSOhv_r. It is worth mentioning that all the evaluated proposals differ only in the way that the global leader is selected, being the local leader selection the same as in the original SMPSO. The proposed variants are explained in the following.

1) SMPSO_r: This is the simplest of the analyzed versions. It behaves exactly as SMPSO but, when the velocity of a particle has to be updated, it considers as leader a solution randomly selected from the leaders archive, as in [16]. We have included this version as a sanity check, to see if any possible improvement in the SMPSO variants are due to the leader selection scheme.

2) cellSMPSO: This variant analises the effect of structuring the swarm using a neighborhood for leader selection. For this, we have used a cellular topology as the one adopted in [29], that allows to define a set of neighbors for each particle based on the position where the particle is located. In particular, we have considered eight neighbors for each solution. The idea is that the global leader is selected among the local leaders of the neighbor particles. Preliminary experiments showed, however, that this approach experimented some troubles for converging towards the Pareto optimal front. One of the hypothesis for this behavior is that selecting leaders in this way may increases too much the diversity of the algorithm, penalizing convergence. To diminish this effect, we have considered that in 50% of the cases, the leaders are selected among the neighboring solutions and in the other 50% the leaders are selected as in the original SMPSO.

3) SMPSOhv: This variant uses the hypervolume quality indicator (described later in Section IV), following the idea of selecting as a leader one of the particles contributing the most to the hypervolume of the Pareto front approximation computed so far by the algorithm. To apply this scheme, we changed the archive of leaders for an archive managed by the contribution of each solution to the value of this indicator. This archive works as the one described for SMPSO but, when the archive becomes full, instead of discarding the solution with the smallest crowding distance, we choose the solution contributing the least to the hypervolume. In this variant, when the velocity of a particle has to be updated, two solutions are randomly selected from the archive, and the one contributing the most to the archive's hypervolume is selected as the leader.

4) SMPSOhv_r: This variant uses the same archive described for the previous variant, but the leader is randomly selected from the archive, without considering its contribution to the hypervolume.

IV. Experimental Methodology

This section describes the benchmark problems adopted for our tests, the parameter settings and the methodology followed in our experiments.

A. Benchmark Problems

We have considered three different benchmark families, accounting for a total of 21 test problems. More specifically, we have considered the following benchmarks: ZDT [30] (problems ZDT1-4 and ZDT6), DTLZ [31] (problems DTLZ1-7), and WFG [32] (problems WFG1-9).

Overall, our benchmarking set contains problems with different properties: convex, non-convex, disconnected, multi-frontal, many-to-one problems. The WFG and DTLZ problems are configured with two and three objectives, respectively.

B. Parameters Settings

For all the variants we have used the same parameters settings to guarantee a fair comparison among the techniques. We have used 100 particles in all cases and an archive consisting of a maximum of 100 leaders. As SMPSO uses a polynomial-based mutation, we have considered that this perturbation is applied with a probability $1/L$ to each variable, being L the number of variables of the problem.

As indicated in the introduction, the best performing SMPSO variant will be compared with two algorithms, NSGA-II and SMS-EMOA, to put the results in an appropriate context. Both algorithms use populations of size equal to 100 and SBX [28] and polynomial-based mutation are adopted for crossover and mutation, respectively. The crossover probability is $p_c = 0.9$ and the mutation probability is $p_m = 1/L$, as in SMPSO. The stopping condition is to perform $25,000$ function evaluations in each of the studied problems.

C. Quality Assessment

To assess the performance of a multi-objective evolutionary algorithm, a number of quality indicators have been proposed, aiming to quantify to some extent convergence (how close it

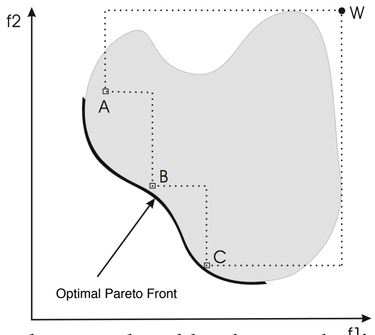


Fig. 1: Hypervolume enclosed by the non-dominated solutions A, B, and C.

is the computed approximation to the Pareto optimal front) and diversity (how well distributed are the solutions in that approximation). Here, we adopt two widely used indicators: additive epsilon [33]) and hypervolume [10]. These quality indicators are defined as follows:

- **Epsilon** (I_ϵ^+). Given a computed approximation front for a problem, A, this indicator is a measure of the smallest distance that one would need to translate every solution in A, so that it dominates the Pareto optimal front of this problem. More formally, given $\vec{z^1} = (z_1^1, \ldots, z_n^1)$ and $\vec{z^2} = (z_1^2, \ldots, z_n^2)$, where n is the number of objectives:

$$I_{\epsilon+}^1(A) = inf \left\{ \epsilon \in \mathbb{R} | \forall \vec{z^2} \in \mathcal{PF}^* \; \exists \vec{z^1} \in A : \vec{z^1} \prec_\epsilon \vec{z^2} \right\} \tag{7}$$

  where, $\vec{z^1} \prec_\epsilon \vec{z^2}$ if and only if $\forall 1 \leq i \leq n : z_i^1 < \epsilon + z_i^2$.
  Fronts with small values of I_ϵ^+ are desirable.

- **Hypervolume** (I_{HV}). This indicator calculates the volume, in objective function space, covered by members of a non-dominated set of solutions $Q = A, B, C$ and a reference point W, e.g., the region enclosed into the discontinuous line in Fig. 1, for problems where all the objectives are to be minimized [10]. Mathematically, for each solution $i \in Q$, a hypercube v_i is constructed with a W and the solution i as the diagonal corners of the hypercube. The reference point can be simply found by constructing a vector with the worst objective function values. Thereafter, the union of all hypercubes is computed and its hypervolume (I_{HV}) is calculated as:

$$I_{HV} = \text{volume} \left(\bigcup_{i=1}^{|Q|} v_i \right). \tag{8}$$

  The higher the value of I_{HV}, the better the approximated Pareto front is.

Since these indicators are not free from arbitrary scaling of the objectives, we apply them after normalizing the values of the objective functions.

D. Analysis of Results

For each combination of algorithm and problem we have made 30 independent runs, and we report the median, $\tilde{x}$, and the interquartile range, IQR, as measures of location (or central tendency) and statistical dispersion, respectively, for each considered indicator. When presenting the obtained values in tables, we emphasize with a dark gray background the best result for each problem, and a clear grey background is used to indicate the second best result; this way, we can see at a glance the most salient algorithms.

When comparing the values yielded by two algorithms on a given problem, we check if differences in the results are statistically significant. To cope with this issue, we have applied the unpaired Wilcoxon rank-sum test, a non-parametric statistical hypothesis test, which allows us to make pairwise comparisons between algorithms to analyze the significance of the obtained data [34]. A confidence level of 95% (i.e., significance level of 5% or *p*-value under 0.05) has been used in all cases, meaning that the differences were unlikely to occur by chance with a probability of 95%.

TABLE II: Comparison among the SMPSO variants: Problems in which no statistical confidence has been found for the I_ϵ^+

	SMPSO_r	cellSMPSO	SMPSOhv	SMPSOhv_r
SMPSO	DTLZ2		WFG8	
SMPSO_r		ZDT1, ZDT2, ZDT3 ZDT4, ZDT6, DTLZ1 DTLZ2, DTLZ3, DTLZ4 DTLZ5, DTLZ6, DTLZ7 WFG1, WFG2, WFG3 WFG4, WFG5, WFG6 WFG7, WFG8, WFG9		
cellSMPSO				WFG8
SMPSOhv				ZDT1, ZDT4, ZDT6 DTLZ1, DTLZ2, DTLZ3 DTLZ5, DTLZ6, WFG1 WFG5, WFG9

V. Results

This section compares the different proposed SMPSO variants in the aforementioned benchmark problems. The best performing variant will be later compared with SMS-EMOA and NSGA-II.

A. SMPSO Variants Comparison

We first analyze the values obtained in the I_ϵ^+ indicator, which are included in Table I. The SMPSO variant which selects the leaders based on their contribution to the hypervolume indicator, SMPSOhv, has been the most salient alternative, obtaining the best results in all but four cases, in which it was the second best alternative. SMPSOhv_r, which selects the leader randomly from the archive managed by the hypervolume for discarding solutions, has obtained the best value in two out of the 21 evaluated problems and the second best results in another 15 cases. The original SMPSO algorithm has obtained the best results in only two problems and the second best results in another two. The other two analyzed versions have obtained the poorest values in the comparison.

Table II summarizes the results of our statistical analysis. In particular, for each pair of algorithms, the table includes the problems in which no statistical confidence has been found in our experiments. We can observe that, from the point of view of the convergence indicator, there are no statistical differences between the two worst techniques in the comparison, SMPSO_r and cellSMPSO. The number of problems in which no statistical significance has been found between SMPSOhv

TABLE I: Median IQR of the comparison among the different SMPSO variants with respect to I_{ϵ}^{+}

	SMPSO	SMPSO_r	cellSMPSO	SMPSOhv	SMPSOhv_r
ZDT1	$5.56e-03_{1.9e-04}$	$5.75e-03_{2.7e-04}$	$9.71e-03_{2.0e-03}$	$5.22e-03_{2.3e-04}$	$5.31e-03_{2.2e-04}$
ZDT2	$5.51e-03_{1.1e-04}$	$5.45e-03_{1.7e-04}$	$7.91e-03_{1.9e-03}$	$5.06e-03_{1.5e-04}$	$5.26e-03_{2.1e-04}$
ZDT3	$5.68e-03_{1.0e-03}$	$6.29e-03_{1.8e-03}$	$2.63e-02_{2.3e-02}$	$3.84e-03_{5.4e-04}$	$4.13e-03_{1.0e-03}$
ZDT4	$6.19e-03_{3.5e-04}$	$6.36e-03_{6.5e-04}$	$5.70e-02_{5.4e-02}$	$5.53e-03_{2.7e-04}$	$5.60e-03_{3.4e-04}$
ZDT6	$4.59e-03_{4.1e-04}$	$4.64e-03_{2.1e-04}$	$6.13e-03_{2.4e-03}$	$4.43e-03_{2.8e-04}$	$4.44e-03_{3.8e-04}$
DTLZ1	$5.69e-02_{9.5e-03}$	$6.11e-02_{1.1e-02}$	$8.63e-02_{2.5e-02}$	$2.79e-02_{1.2e-03}$	$2.84e-02_{1.0e-03}$
DTLZ2	$1.32e-01_{1.9e-02}$	$1.49e-01_{2.4e-02}$	$1.45e-01_{2.0e-02}$	$5.35e-02_{2.3e-03}$	$5.35e-02_{2.7e-03}$
DTLZ3	$1.49e-01_{5.0e-01}$	$1.49e-01_{3.0e-02}$	$4.79e-01_{2.5e-01}$	$5.29e-02_{4.6e-03}$	$5.38e-02_{2.8e-03}$
DTLZ4	$1.24e-01_{3.7e-02}$	$1.24e-01_{4.8e-02}$	$2.10e-01_{4.3e-02}$	$5.70e-02_{1.7e-02}$	$8.14e-02_{7.4e-02}$
DTLZ5	$4.61e-03_{3.0e-04}$	$4.60e-03_{3.3e-04}$	$6.72e-03_{1.2e-03}$	$3.83e-03_{2.7e-04}$	$3.80e-03_{2.2e-04}$
DTLZ6	$4.38e-03_{3.2e-04}$	$4.32e-03_{6.1e-04}$	$4.65e-03_{5.0e-04}$	$3.73e-03_{1.3e-04}$	$3.77e-03_{3.0e-04}$
DTLZ7	$1.78e-01_{7.1e-02}$	$1.67e-01_{7.3e-02}$	$1.99e-01_{8.6e-02}$	$1.44e-01_{7.2e-03}$	$1.45e-01_{8.4e-03}$
WFG1	$1.14e+00_{4.4e-02}$	$1.19e+00_{7.3e-02}$	$1.35e+00_{5.1e-02}$	$1.16e+00_{5.9e-02}$	$1.20e+00_{6.5e-02}$
WFG2	$1.48e-02_{2.1e-03}$	$2.02e-02_{7.7e-03}$	$7.49e-02_{1.1e-02}$	$1.04e-02_{4.6e-03}$	$1.97e-02_{6.4e-03}$
WFG3	$2.00e+00_{9.6e-04}$	$2.00e+00_{1.4e-03}$	$2.03e+00_{1.4e-02}$	$2.00e+00_{1.2e-04}$	$2.00e+00_{2.7e-04}$
WFG4	$5.22e-02_{6.6e-03}$	$5.49e-02_{4.8e-03}$	$7.13e-02_{8.8e-03}$	$5.13e-02_{5.9e-03}$	$5.48e-02_{1.2e-02}$
WFG5	$6.36e-02_{7.2e-04}$	$6.38e-02_{3.6e-03}$	$6.60e-02_{3.5e-02}$	$6.25e-02_{1.0e-03}$	$6.26e-02_{1.4e-03}$
WFG6	$1.66e-02_{1.3e-03}$	$1.74e-02_{2.3e-03}$	$5.97e-02_{1.1e-02}$	$1.33e-02_{1.1e-03}$	$1.39e-02_{1.3e-03}$
WFG7	$1.86e-02_{1.4e-03}$	$1.84e-02_{1.5e-03}$	$7.05e-02_{1.5e-02}$	$1.42e-02_{7.3e-04}$	$1.48e-02_{8.5e-04}$
WFG8	$3.84e-01_{5.5e-02}$	$4.16e-01_{4.1e-02}$	$4.28e-01_{5.0e-02}$	$3.92e-01_{5.7e-02}$	$4.24e-01_{3.8e-02}$
WFG9	$2.84e-02_{2.3e-03}$	$2.74e-02_{3.1e-03}$	$3.79e-02_{5.4e-03}$	$2.32e-02_{1.7e-03}$	$2.38e-02_{2.0e-03}$

and SMPSOhv_r is also high. However, there were still many cases where the former algorithm outperformed the latter with statistical significance.

The main conclusion extracted from these results is that the algorithms which incorporate an archive based on hypervolume for discarding solutions have computed fronts with better convergence than the others. This result could be somehow expected, since the crowding distance (i.e., the density estimator used in the other alternatives) does not take convergence into account. It is remarkable that the original SMPSO, which proved to be very effective when solving the ZDT problems, is clearly outperformed on this suite by the versions using the hypervolume.

The results yielded by the analyzed SMPSO variants for I_{HV} are included in Table III. SMPSOhv has obtained the best results in all the evaluated problems but in three instances (DTLZ6, DTLZ7, and WFG1), and SMPSOhv_r has been the second best alternative in a high number of cases (16 out of the 21 problems adopted). SMPSO obtained the best values of the indicator in two cases, and the second best value in another two. cellSMPSO and SMPSO_r have also obtained good figures in a couple of problems, which is something that didn't happen in the case of the convergence indicator.

The results of our statistical analysis for the I_{HV} indicator are summarized in Table IV. In this case, the number of problems in which statistical differences can be assured between SMPSOhv and SMPSOhv_r is higher than when using I_{ϵ}^{+}. This means that leader selection based on I_{HV} helps to obtain fronts with a better spread in a high number of problems. This fact is, however, not as noticeable in the comparison between SMPSO and SMPSO_r, where no statistical confidence has been found in many problems.

B. Comparison with State-of-the-art Algorithms

In this section, we compare the SMPSO variant performing the best, SMPSOhv, with respect to NSGA-II and SMS-EMOA. The idea is to put the obtained results in context with those produced by two state-of-the-art multi-objective optimizers.

Proceeding as before, we analyze first the obtained values with respect to the I_{ϵ}^{+} indicator (Table V). We can observe

TABLE IV: Comparison among the SMPSO variants: Problems in which no statistical confidence has been found for I_{HV}

	SMPSO_r	cellSMPSO	SMPSOhv	SMPSOhv_r
SMPSO	ZDT1, DTLZ3, DTZL6 DTLZ7, WFG5, WFG6 WFG8, WFG9	WFG5	DTLZ6 WFG1 WFG3	DTLZ6, WFG1 WFG2, WFG7 WFG8
SMPSO_r		DTLZ2 WFG5	DTLZ6 WFG1	DTLZ6, WFG1 WFG3, WFG7 WFG8
cellSMPSO				
SMPSOhv				DTLZ1, DTLZ5 DTLZ6, WFG1 WFG5

TABLE VI: Comparison SMPSOhv vs SMS-EMOA vs NSGA-II: Problems in which no statistical confidence has been found with respect to the I_{ϵ}^{+}

	SMSEMOA	NSGA-II
SMPSOhv	DTLZ4, DTLZ5 DTLZ7, WFG2	WFG8
SMSEMOA		ZDT4, DTLZ4 WFG3

that SMPSOhv has obtained the best values in 13 out of the 21 evaluated problems, and the second best values in other six problems. SMS-EMOA and NSGA-II have yielded the best indicator values in 6 and 2 problems, respectively.

Table VI includes the problems in which no statistical confidence has been found among the evaluated algorithms in terms of I_{ϵ}^{+}. In this case, we see that the results have been significant in most cases. Only in the comparison between SMPSOhv and SMS-EMOA, statistical confidence was not found in 4 out of the 21 evaluated problems.

In the case of the I_{HV} (see Table VII), the analysis of the algorithms has the same sign: SMPSOhv has been the most salient algorithm, followed by SMS-EMOA and, finally, NSGA-II. In fact, the results are practically the same, in terms of the problems in which each algorithm has yielded the best values, that those obtained with the convergence indicator. This fact is also reflected in Table VIII, where the problems in which no statistical confidence has been found between SMPSOhv and SMS-EMOA, for this indicator, are almost the same as for the other performance indicator.

To illustrate the fronts that can be produced by the three

TABLE III: Median IQR of the comparison among the different SMPSO variants with respect to I_{HV}

	SMPSO	SMPSO_r	cellSMPSO	SMPSOhv	SMPSOhv_r
ZDT1	$6.62e-01_{1.1e-04}$	$6.62e-01_{1.6e-04}$	$6.59e-01_{1.4e-03}$	$6.62e-01_{9.3e-06}$	$6.62e-01_{1.6e-05}$
ZDT2	$3.29e-01_{1.4e-04}$	$3.29e-01_{1.2e-04}$	$3.27e-01_{1.1e-03}$	$3.29e-01_{6.3e-06}$	$3.29e-01_{1.5e-05}$
ZDT3	$5.15e-01_{5.1e-04}$	$5.15e-01_{9.6e-04}$	$5.08e-01_{4.7e-03}$	$5.16e-01_{9.3e-05}$	$5.16e-01_{4.9e-04}$
ZDT4	$6.61e-01_{2.7e-04}$	$6.61e-01_{2.4e-04}$	$6.52e-01_{1.3e-02}$	$6.62e-01_{2.0e-05}$	$6.62e-01_{2.6e-05}$
ZDT6	$4.01e-01_{1.1e-04}$	$4.01e-01_{1.2e-04}$	$4.01e-01_{2.4e-04}$	$4.01e-01_{2.3e-05}$	$4.01e-01_{3.1e-05}$
DTLZ1	$7.36e-01_{1.4e-02}$	$7.30e-01_{1.1e-02}$	$7.10e-01_{3.4e-02}$	$7.89e-01_{5.4e-04}$	$7.89e-01_{5.7e-04}$
DTLZ2	$3.52e-01_{8.8e-03}$	$3.43e-01_{7.8e-03}$	$3.43e-01_{9.8e-03}$	$4.25e-01_{3.6e-04}$	$4.25e-01_{6.3e-04}$
DTLZ3	$3.48e-01_{1.1e-01}$	$3.46e-01_{1.6e-02}$	$3.30e-01_{3.2e-02}$	$4.25e-01_{1.6e-03}$	$4.24e-01_{2.8e-03}$
DTLZ4	$3.59e-01_{1.7e-02}$	$3.60e-01_{1.2e-02}$	$2.91e-01_{1.9e-02}$	$4.16e-01_{8.0e-03}$	$4.05e-01_{4.6e-02}$
DTLZ5	$9.37e-02_{1.0e-04}$	$9.36e-02_{1.6e-04}$	$9.33e-02_{1.7e-04}$	$9.40e-02_{6.3e-05}$	$9.40e-02_{7.2e-05}$
DTLZ6	$9.49e-02_{4.5e-05}$	$9.49e-02_{6.2e-05}$	$9.50e-02_{5.3e-05}$	$9.49e-02_{6.6e-05}$	$9.49e-02_{6.4e-05}$
DTLZ7	$2.74e-01_{8.4e-03}$	$2.72e-01_{5.9e-03}$	$2.64e-01_{8.5e-03}$	$2.56e-01_{1.5e-03}$	$2.57e-01_{7.0e-03}$
WFG1	$1.18e-01_{4.7e-03}$	$1.15e-01_{5.3e-03}$	$9.42e-02_{3.7e-03}$	$1.17e-01_{5.8e-03}$	$1.15e-01_{8.7e-03}$
WFG2	$5.61e-01_{7.2e-04}$	$5.60e-01_{1.6e-03}$	$5.37e-01_{3.3e-03}$	$5.63e-01_{7.9e-04}$	$5.61e-01_{9.7e-04}$
WFG3	$4.41e-01_{1.5e-04}$	$4.41e-01_{2.6e-04}$	$4.29e-01_{1.9e-03}$	$4.42e-01_{5.4e-05}$	$4.42e-01_{6.3e-05}$
WFG4	$2.03e-01_{2.4e-03}$	$2.01e-01_{1.4e-03}$	$1.92e-01_{1.5e-03}$	$2.04e-01_{2.5e-03}$	$2.01e-01_{2.4e-03}$
WFG5	$1.96e-01_{8.6e-05}$	$1.96e-01_{6.5e-05}$	$1.96e-01_{9.1e-05}$	$1.97e-01_{4.6e-05}$	$1.97e-01_{3.4e-05}$
WFG6	$2.09e-01_{3.3e-04}$	$2.09e-01_{3.5e-04}$	$1.95e-01_{2.2e-03}$	$2.11e-01_{1.2e-04}$	$2.11e-01_{1.5e-04}$
WFG7	$2.09e-01_{2.1e-04}$	$2.09e-01_{3.4e-04}$	$1.91e-01_{2.2e-03}$	$2.11e-01_{7.4e-05}$	$2.11e-01_{1.2e-04}$
WFG8	$1.48e-01_{1.5e-03}$	$1.48e-01_{6.3e-04}$	$1.35e-01_{2.2e-03}$	$1.49e-01_{7.9e-04}$	$1.48e-01_{9.5e-04}$
WFG9	$2.35e-01_{7.2e-04}$	$2.35e-01_{5.7e-04}$	$2.30e-01_{7.1e-04}$	$2.38e-01_{7.4e-04}$	$2.37e-01_{4.7e-04}$

TABLE V: Median IQR of the comparison SMPSOhv vs SMS-EMOA vs NSGA-II with respect to I_{ϵ}^{+}

	SMPSOhv	SMSEMOA	NSGA-II
ZDT1	$5.22e-03_{2.3e-04}$	$5.69e-03_{2.6e-04}$	$1.26e-02_{1.9e-03}$
ZDT2	$5.06e-03_{1.5e-04}$	$5.76e-03_{3.1e-04}$	$1.33e-02_{2.0e-03}$
ZDT3	$3.84e-03_{5.4e-04}$	$4.04e-03_{6.6e-04}$	$8.80e-03_{1.8e-03}$
ZDT4	$5.53e-03_{2.7e-04}$	$1.41e-02_{5.8e-02}$	$1.45e-02_{4.1e-03}$
ZDT6	$4.43e-03_{2.8e-04}$	$7.46e-03_{4.5e-04}$	$1.50e-02_{3.3e-03}$
DTLZ1	$2.79e-02_{1.2e-03}$	$2.84e-02_{1.8e-03}$	$7.03e-02_{2.1e-02}$
DTLZ2	$5.35e-02_{2.3e-03}$	$5.10e-02_{1.8e-03}$	$1.29e-01_{2.6e-02}$
DTLZ3	$5.29e-02_{4.6e-03}$	$3.68e+00_{2.1e+00}$	$5.26e+00_{2.7e+00}$
DTLZ4	$5.70e-02_{1.7e-02}$	$3.40e-01_{5.9e-01}$	$1.13e-01_{2.2e-02}$
DTLZ5	$3.83e-03_{2.7e-04}$	$3.77e-03_{1.4e-04}$	$1.03e-02_{2.8e-03}$
DTLZ6	$3.73e-03_{1.3e-04}$	$3.30e-01_{4.1e-02}$	$8.51e-01_{8.5e-02}$
DTLZ7	$1.44e-01_{7.2e-03}$	$1.44e-01_{1.7e-02}$	$1.23e-01_{4.5e-02}$
WFG1	$1.16e+00_{5.9e-02}$	$1.00e+00_{3.9e-01}$	$2.67e-01_{4.3e-01}$
WFG2	$1.04e-02_{4.6e-03}$	$3.59e-01_{7.0e-01}$	$3.65e-01_{7.0e-01}$
WFG3	$2.00e+00_{1.2e-04}$	$2.00e+00_{1.4e-03}$	$2.00e+00_{1.6e-03}$
WFG4	$5.13e-02_{5.9e-03}$	$1.29e-02_{8.6e-04}$	$3.36e-02_{8.5e-03}$
WFG5	$6.25e-02_{1.0e-03}$	$6.21e-02_{5.8e-04}$	$8.35e-02_{9.2e-03}$
WFG6	$1.33e-02_{1.1e-03}$	$2.38e-02_{2.6e-02}$	$4.48e-02_{1.9e-02}$
WFG7	$1.42e-02_{7.3e-04}$	$1.31e-02_{7.2e-04}$	$3.49e-02_{1.0e-02}$
WFG8	$3.92e-01_{5.7e-02}$	$5.21e-01_{9.7e-02}$	$5.11e-01_{2.3e-01}$
WFG9	$2.32e-02_{1.7e-03}$	$1.53e-02_{1.5e-03}$	$3.63e-02_{7.4e-03}$

TABLE VIII: Comparison among SMPSOhv vs SMS-EMOA vs NSGA-II: Problems in which no statistical confidence has been found with respect to I_{HV}

	SMSEMOA	NSGA-II
SMPSOhv	DTLZ4, DTLZ5 WFG2, WFG5	WFG2
SMSEMOA		ZDT4, DTLZ4 WFG6, WFG8

compared metaheuristics, Fig. 2 shows some examples of the best approximation sets computed by these algorithms according to the hypervolume. The chosen problems are ZDT4, WFG6, and DTLZ6.

To finish this section, is worth mentioning the issue of running times, given that it is well-known that computing the hypervolume is expensive. All the algorithms in this paper have been implemented with the jMetal framework [35]. In both, SMS-EMOA and SMPSOhv, we have not included any technique to improve the calculation of the contribution of the hypervolume [36]. Thus, while the original SMPSO and NSGA-II required less than a second to complete a typical run in our testing computer when solving the bi-objective problems, the hypervolume techniques needed about one minute. Improving the implementation of these algorithms is in progress but, in any case, this does not affect the numerical performance of the techniques.

VI. Conclusions and Future Work

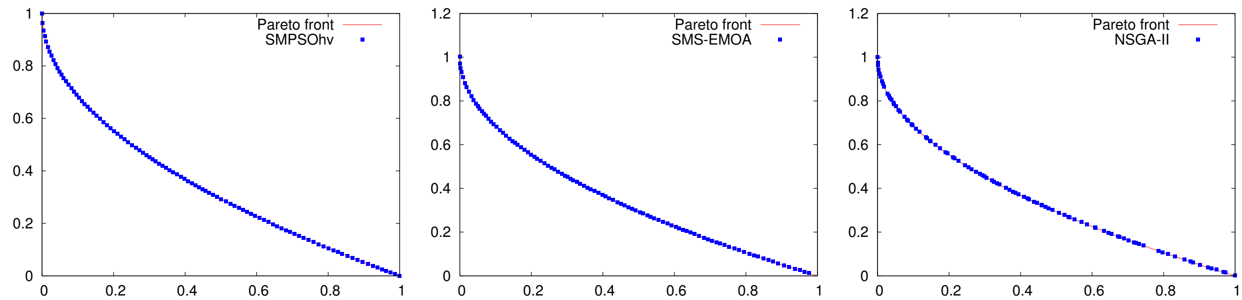
In this paper, we have analyzed a number of leader selection mechanisms in SMPSO, a competitive multi-objective particle swarm optimizer. In particular, four new different versions of SMPSO have been tested. These versions involve the selection of leaders randomly from an external archive (SMPSO_r), from the swarm using a neighborhood structure (cellSMPSO), by considering the contribution to the hypervolume of the particles in the external archive and taking the best of two randomly chosen particles (SMPSOhv), and the same as SMPSOhv but simply randomly selecting the leaders from it (SMPSOhv_r).

These variants have been tested using a benchmark composed of problems with two and three objectives. The obtained results have been analyzed by applying two quality indicators, measuring convergence and diversity. Our results showed that SMPSOhv has been the overall best technique in our comparison. SMPSOhv has been also compared with respect to SMS-EMOA and NSGA-II, two state-of-the-art algorithms, obtaining better values than them in a high number of cases, thus pointing out the advantages of using the hypervolume quality indicator for leader selection in a multi-objective PSO algorithm.

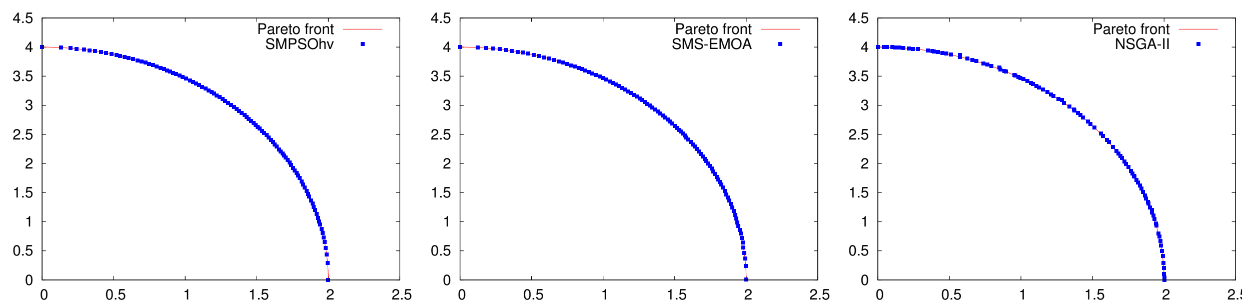
Further work will focus on analyzing the inclusion of other quality indicators in SMPSO and applying SMPSOhv to real-world optimization problems.

TABLE VII: Median IQR of the comparison SMPSOhv vs SMS-EMOA vs NSGA-II with respect to I_{HV}

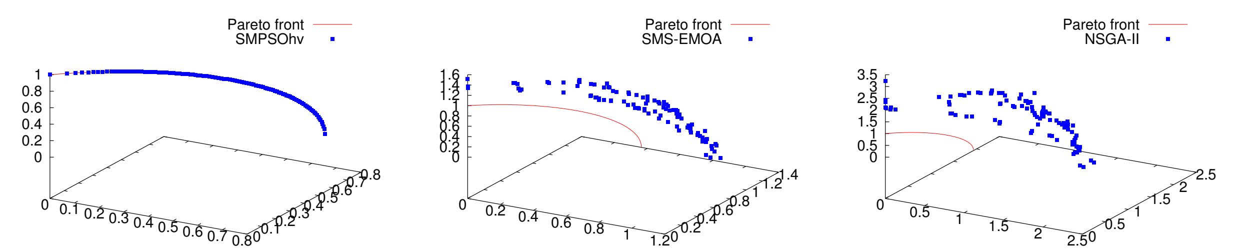
	SMPSOhv	SMSEMOA	NSGA-II
ZDT1	$6.62e-01_{9.3e-06}$	$6.62e-01_{5.5e-05}$	$6.59e-01_{4.3e-04}$
ZDT2	$3.29e-01_{6.3e-06}$	$3.29e-01_{8.5e-05}$	$3.26e-01_{6.0e-04}$
ZDT3	$5.16e-01_{9.3e-05}$	$5.16e-01_{6.2e-05}$	$5.15e-01_{2.5e-04}$
ZDT4	$6.62e-01_{2.0e-05}$	$6.57e-01_{4.3e-03}$	$6.56e-01_{3.1e-03}$
ZDT6	$4.01e-01_{2.3e-05}$	$3.97e-01_{1.2e-03}$	$3.88e-01_{2.0e-03}$
DTLZ1	$7.89e-01_{5.4e-04}$	$7.87e-01_{1.9e-03}$	$7.26e-01_{2.9e-02}$
DTLZ2	$4.25e-01_{3.6e-04}$	$4.27e-01_{3.8e-04}$	$3.74e-01_{7.2e-03}$
DTLZ3	$4.25e-01_{1.6e-03}$	$0.00e+00_{0.0e+00}$	$0.00e+00_{0.0e+00}$
DTLZ4	$4.16e-01_{8.0e-03}$	$3.14e-01_{2.1e-01}$	$3.74e-01_{5.8e-03}$
DTLZ5	$9.40e-02_{6.3e-05}$	$9.40e-02_{4.1e-05}$	$9.28e-02_{3.5e-04}$
DTLZ6	$9.49e-02_{6.6e-05}$	$0.00e+00_{0.0e+00}$	$0.00e+00_{0.0e+00}$
DTLZ7	$2.56e-01_{1.5e-03}$	$2.54e-01_{7.2e-03}$	$2.79e-01_{5.1e-03}$
WFG1	$1.17e-01_{5.8e-03}$	$3.53e-01_{1.7e-01}$	$5.95e-01_{1.1e-01}$
WFG2	$5.63e-01_{7.9e-04}$	$5.62e-01_{2.9e-03}$	$5.62e-01_{2.5e-03}$
WFG3	$4.42e-01_{5.4e-05}$	$4.42e-01_{2.9e-04}$	$4.41e-01_{5.5e-04}$
WFG4	$2.04e-01_{2.5e-03}$	$2.19e-01_{1.5e-05}$	$2.17e-01_{3.8e-04}$
WFG5	$1.97e-01_{4.6e-05}$	$1.97e-01_{3.2e-05}$	$1.95e-01_{3.5e-04}$
WFG6	$2.11e-01_{1.2e-04}$	$2.04e-01_{1.7e-02}$	$2.04e-01_{1.1e-02}$
WFG7	$2.11e-01_{7.4e-05}$	$2.11e-01_{2.6e-05}$	$2.09e-01_{4.3e-04}$
WFG8	$1.49e-01_{7.9e-04}$	$1.48e-01_{1.6e-03}$	$1.47e-01_{3.5e-03}$
WFG9	$2.38e-01_{7.4e-04}$	$2.41e-01_{1.2e-03}$	$2.37e-01_{1.6e-03}$



(a) SMPSOhv in ZDT4 (b) SMS-EMOA in ZDT4 (c) NSGA-II in ZDT4



(d) SMPSOhv in WFG6 (e) SMS-EMOA in WFG6 (f) NSGA-II in WFG6



(g) SMPSOhv in DTLZ6 (h) SMS-EMOA in DTLZ6 (i) NSGA-II in DTLZ6

Fig. 2: Example of fronts computed by SMPSOhv, SMS-EMOA, and NSGA-II for three of the evaluated benchmark problems: ZDT4 (top), WFG6 (middle), and DTLZ6 (bottom).

ACKNOWLEDGMENT

Antonio J. Nebro is supported by projects TIN2011-25840 (Spanish Ministry of Education and Science) and P11-TIC-7529 (Innovation, Science and Enterprise Ministry of the regional government of the Junta de Andalucía). Juan J. Durillo acknowledges the Austrian Research Promotion Agency under contract nr. 834307 (AutoCore). Carlos A. Coello Coello acknowledges support from CONACyT project no. 103570.

References

[1] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Fourth IEEE International Conference on Neural Networks*, 1995, pp. 1942–1948.

[2] M. Reyes-sierra and C. A. C. Coello, "Multi-objective particle swarm optimizers: A survey of the state-of-the-art," *International Journal Of Computational Intelligence Research*, vol. 2, no. 3, pp. 287–308, 2006.

[3] A. Nebro, J. Durillo, J. García-Nieto, C. Coello Coello, F. Luna, and E. Alba, "Smpso: A new pso-based metaheuristic for multi-objective optimization," in *2009 IEEE Symposium on Computational Intelligence in Multicriteria Decision-Making (MCDM 2009)*. IEEE Press, 2009, pp. 66–73.

[4] J. Durillo, J. García-Nieto, A. Nebro, C. A. Coello Coello, F. Luna, and E. Alba, "Multi-objective particle swarm optimizers: An experimental comparison," in *5th International Conference, EMO 2009*, ser. Lecture Notes in Computer Science, vol. 5467. Nantes, France: Springer Berlin / Heidelberg, April 2009, pp. 495–509.

[5] J. J. Durillo, A. J. Nebro, F. Luna, C. A. Coello Coello, and E. Alba, "Convergence speed in multi-objective metaheuristics: Efficiency criteria and empirical study," *International Journal for Numerical Methods in Engineering*, vol. 84, no. 11, pp. 1344 – 1375, December 2010.

[6] J. J. Durillo, A. J. Nebro, C. A. Coello Coello, J. García-Nieto, F. Luna, and E. Alba, "A study of multiobjective metaheuristics when solving parameter scalable problems," *IEEE Transctions on Evolutionary Computation*, vol. 14, no. 4, pp. 618 – 635, August 2010.

[7] R. Mendes, J. Kennedy, and J. Neves, "The fully informed particle swarm: Simpler, maybe better," *IEEE Transactions on Evolutionary Computation*, vol. 3, no. 8, pp. 204–210, 2004.

[8] J. García-Nieto and E. Alba, "Why six informants is optimal in pso," in *21st Genetic and Evolutionary Conference, GECCO 2012*, July 2012, pp. 25–32.

[9] M. A. M. de Oca and T. Stüzle, "Convergence behavior of the fully informed particle swarm optimization algorithm," in *17ht Genetic and Evolutionary Conference, GECCO 2008*, July 2008, pp. 71–78.

[10] E. Zitzler and L. Thiele, "Multiobjective evolutionary algorithms: a comparative case study and the strength pareto approach," *IEEE Trans. Evolutionary Computation*, vol. 3, no. 4, pp. 257–271, 1999.

[11] E. Zitzler and S. Künzli, "Indicator-based selection in multiobjective search," in *in Proc. 8th International Conference on Parallel Problem Solving from Nature, PPSN VIII*. Springer, 2004, pp. 832–842.

[12] N. Beume, B. Naujoks, and M. Emmerich, "Sms-emoa: Multiobjective selection based on dominated hypervolume," *European Journal of Operational Research*, vol. 181, no. 3, pp. 1653–1669, 2007.

[13] N. Padhye, J. Branke, and S. Mostaghim, "Empirical comparison of mopso methods - guide selection and diversity preservation -," in *Evolutionary Computation, 2009. CEC '09. IEEE Congress on*, 2009, pp. 2516–2523.

[14] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast elitist multi-objective genetic algorithm: Nsga-ii," *IEEE Transactions on Evolutionary Computation*, vol. 6, pp. 182–197, 2000.

[15] J. Moore, R. Chapman, and G. Dozier, "Multiobjective particle swarm optimization," in *Proceedings of the 38th annual on Southeast regional conference*, ser. ACM-SE 38. New York, NY, USA: ACM, 2000, pp. 56–57.

[16] J. Reddy, M., N. Kumar, and D., "An efficient multi-objective optimization algorithm based on swarm intelligence for engineering design," *Engineering Optimization*, vol. 39, no. 1, pp. 49–68, Jan. 2007.

[17] R. A. Santana, M. R. Pontes, and C. J. A. Bastos-Filho, "A multiple objective particle swarm optimization approach using crowding distance and roulette wheel," in *Proceedings of the 2009 Ninth International Conference on Intelligent Systems Design and Applications*, ser. ISDA '09. Washington, DC, USA: IEEE Computer Society, 2009, pp. 237–242.

[18] C. Coello Coello and M. Lechuga, "Mopso: a proposal for multiple objective particle swarm optimization," in *Evolutionary Computation, 2002. CEC '02. Proceedings of the 2002 Congress on*, vol. 2, 2002, pp. 1051 –1056.

[19] J. Hernández-Domínguez, G. Toscano-Pulido, and C. Coello Coello, "A multi-objective particle swarm optimizer enhanced with a differential evolution scheme," in *Artificial Evolution*, ser. Lecture Notes in Computer Science, J.-K. Hao, P. Legrand, P. Collet, N. Monmarch, E. Lutton, and M. Schoenauer, Eds. Springer Berlin Heidelberg, 2012, vol. 7401, pp. 169–180.

[20] S. Mostaghim and J. Teich, "Strategies for finding good local guides in multi-objective particle swarm optimization (mopso)," in *Swarm Intelligence Symposium, 2003. SIS '03. Proceedings of the 2003 IEEE*, april 2003, pp. 26 – 33.

[21] U. Wickramasinghe and X. Li, "Choosing leaders for multi-objective pso algorithms using differential evolution," in *Proceedings of the 7th International Conference on Simulated Evolution and Learning*, ser. SEAL '08. Berlin, Heidelberg: Springer-Verlag, 2008, pp. 249–258.

[22] T. Bartz-Beielstein, P. Limbourg, J. Mehnen, K. Schmitt, K. Parsopoulos, and M. Vrahatis, "Particle swarm optimizers for pareto optimization with enhanced archiving techniques," in *Evolutionary Computation, 2003. CEC '03. The 2003 Congress on*, vol. 3, dec. 2003, pp. 1780–1787.

[23] R. Storn and K. Price, "Differential evolution - a simple and efficient adaptive scheme for global optimization over continuous spaces," Berkeley, CA, Tech. Rep. TR-95-012, 1995. [Online]. Available: http://citeseer.ist.psu.edu/182432.html

[24] G. Dupont, S. Adam, Y. Lecourtier, and B. Grilhère, "Multi objective particle swarm optimization using enhanced dominance and guide selection," *International Journal of Computational Intelligence Research*, vol. 4, no. 2, pp. 145–158, 2008.

[25] K. S. Lim, S. Buyamin, A. Ahmad, and Z. Ibrahim, "An improved leader guidance in multi objective particle swarm optimization," *Asia International Conference on Modelling And Simulation*, vol. 0, pp. 34–39, 2012.

[26] S.-Y. Chiu, T.-Y. Sun, S.-T. Hsieh, and C.-W. Lin, "Cross-searching strategy for multi-objective particle swarm optimization," in *IEEE Congress on Evolutionary Computation*, 2007, pp. 3135–3141.

[27] M. Clerc and J. Kennedy, "The particle swarm - explosion, stability, and convergence in a multidimensional complex space," *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 1, pp. 58–73, 2002.

[28] K. Deb, *Multi-Objective Optimization Using Evolutionary Algorithms*. John Wiley & Sons, 2001.

[29] A. Nebro, J. Durillo, F. Luna, B. Dorronsoro, and E. Alba, "Mocell: A cellular genetic algorithm for multiobjective optimization," *International Journal Intelligent Systems*, vol. 24, no. 7, pp. 726–746, 2009.

[30] E. Zitzler, K. Deb, and L. Thiele, "Comparison of Multiobjective Evolutionary Algorithms: Empirical Results," *Evolutionary Computation*, vol. 8, no. 2, pp. 173–195, 2000.

[31] K. Deb, L. Thiele, M. Laumanns, and E. Zitzler, "Scalable Test Problems for Evolutionary Multiobjective Optimization," in *Evolutionary Multiobjective Optimization. Theoretical Advances and Applications*, A. Abraham, L. Jain, and R. Goldberg, Eds. USA: Springer, 2005, pp. 105–145.

[32] S. Huband, L. Barone, R. L. While, and P. Hingston, "A scalable multi-objective test problem toolkit," in *Third International Conference on Evolutionary MultiCriterion Optimization, EMO 2005*, ser. Lecture Notes in Computer Science, C. Coello, A. Hernández, and E. Zitler, Eds., vol. 3410. Springer, 2005, pp. 280–295.

[33] J. Knowles, L. Thiele, and E. Zitzler, "A Tutorial on the Performance Assessment of Stochastic Multiobjective Optimizers," Computer Engineering and Networks Laboratory (TIK), ETH Zurich, Tech. Rep. 214, 2006.

[34] J. Demšar, "Statistical Comparisons of Classifiers over Multiple Data Sets," *J. Mach. Learn. Res.*, vol. 7, pp. 1–30, 2006.

[35] J. J. Durillo and A. J. Nebro, "jmetal: A java framework for multi-objective optimization," *Advances in Engineering Software*, vol. 42, pp. 760–771, 2011.

[36] *Proceedings of the IEEE Congress on Evolutionary Computation, CEC 2012, Brisbane, Australia, June 10-15, 2012*. IEEE, 2012.