Extending the Speed-Constrained Multi-objective PSO (SMPSO) with Reference Point Based Preference Articulation

Antonio J. Nebro[1(✉)], Juan J. Durillo[2], José García-Nieto[1], Cristóbal Barba-González[1], Javier Del Ser[3,4,5], Carlos A. Coello Coello[6], Antonio Benítez-Hidalgo[1], and José F. Aldana-Montes[1]

[1] Dept. de Lenguajes y Ciencias de la Computación, Universidad de Málaga, Málaga, Spain
antonio@lcc.uma.es
[2] Leibniz Supercomputing Centre, Munich, Germany
[3] TECNALIA Research and Innovation, Derio, Spain
[4] University of the Basque Country (UPV/EHU), Bilbao, Spain
[5] Basque Center for Applied Mathematics (BCAM), Bilbao, Spain
[6] Computer Science Department, CINVESTAV-IPN (Evolutionary Computation Group), Ciudad de México, Mexico

Abstract. The Speed-constrained Multi-objective PSO (SMPSO) is an approach featuring an external bounded archive to store non-dominated solutions found during the search and out of which leaders that guide the particles are chosen. Here, we introduce SMPSO/RP, an extension of SMPSO based on the idea of reference point archives. These are external archives with an associated reference point so that only solutions that are dominated by the reference point or that dominate it are considered for their possible addition. SMPSO/RP can manage several reference point archives, so it can effectively be used to focus the search on one or more regions of interest. Furthermore, the algorithm allows interactively changing the reference points during its execution. Additionally, the particles of the swarm can be evaluated in parallel. We compare SMPSO/RP with respect to three other reference point based algorithms. Our results indicate that our proposed approach outperforms the other techniques with respect to which it was compared when solving a variety of problems by selecting both achievable and unachievable reference points. A real-world application related to civil engineering is also included to show up the real applicability of SMPSO/RP.

Keywords: Multi-objective optimization · SMPSO
Decision making · Reference point

1 Introduction

Dealing with a multi-objective optimization problem involves finding its Pareto front or a reasonably good approximation to it in case of using non-exact

A. Auger et al. (Eds.): PPSN 2018, LNCS 11101, pp. 298–310, 2018.
https://doi.org/10.1007/978-3-319-99253-2_24

optimization techniques such as metaheuristics [1]. This accuracy is expressed, in general, in terms of convergence and diversity, with the aim of offering the decision maker (DM) a set of optimal or quasi-optimal solutions evenly spread along the Pareto front. In practice, the DM is usually only interested in a portion of the Pareto front, which can be provided by integrating user's preferences within multi-objective metaheuristics [2]. The preference information can be given to the algorithm *a prori*, before starting the search process, and/or in an interactive way, during the search.

In this paper, we propose an extension of the SMPSO multi-objective particle swarm algorithm [3] to allow DMs to guide the search towards one or more regions of interest by indicating preferences a priori and interactively. SMPSO features a bounded-size external archive where a diverse subset of the non-dominated solutions found during the search is kept and from which global leaders are chosen to compute the speed of the particles. When the archive becomes full, a density estimator (e.g., the crowding distance [4]) is applied in order to remove the solution which least contributes in terms of diversity.

Our extension makes use of reference points as a mean for articulating DM's preferences. We associate an external archive to each reference point. Newly solutions (i.e., every time a particle changes its position) are checked to be added within each of these archives as follows: if the newly generated solution and the archive reference point are non-dominated with respect to each other, nothing is done; otherwise, the former is added to the archive using the same strategy as in SMPSO. This way, reference point archives only keep the non-dominated solutions of the selected preference region. Our proposal, called SMPSO/RP, also modifies the leader selection strategy to select an external archive randomly and then take the leader from it; this mechanism promotes diversity of the swarm and avoids concentrating the search in a single region of interest.

As solving real-world problems might be highly time-consuming, adding the capability of changing the reference points interactively is a basic feature that allows the DM to adjust and focus the search towards the regions of interest. On the contrary, approaches based on static reference points would require re-starting the search from scratch every time the reference point is changed. In SMPSO/RP, the strategy followed when a reference point is changed is to remove all the solutions of the corresponding archive that are non-dominated with respect the new reference point.

The main contributions of this paper are summarized as follows:

1. A new algorithm, SMPSO/RP, that extends SMPSO by incorporating interactive reference point preference articulation. SMPSO/RP has the following features:
   - Ability to deal with one or more DM preferences or regions of interest.
   - Ability to interactively change DM preferences by means of changing the desired reference points.
   - Ability of parallel evaluations of particles.
   - GUI for visualizing the computed front evolution for problems with two and three objectives.

2. Comparison against three reference point based multi-objective evolutionary algorithms.
3. Application of SMPSO/SP to a real-world problem of the domain of civil engineering.
4. Freely available implementation of SMPSO/RP within the jMetal [5] framework[1].

The rest of the paper is organized as follows. Section 2 contains background concepts and our proposal is described in Sect. 3. We devote Sects. 4 and 5 for assessing the performance of SMPSO/RP. A real-world application of our proposal is included in Sect. 6. The conclusions and some possible paths of future work are indicated in Sect. 7.

2 Background

Preference-based multi-objective metaheuristics aim at finding the most interesting parts according the criteria of a DM instead of the full Pareto front. This has been a relatively active research area in the last two decades [6–8].

In this work we are interested in the reference point method [9]. This method constitutes a simple way to delimit an interest region of the objective space by the definition of a user-defined point by the DM, as it requires no parameter defining the width of the region of interest. Given a reference point P, the region of interest is the subset of the Pareto front dominated by P if this is not achievable, or the subset of the Pareto front dominating P if this is achievable. This approach is very similar to the g-dominance concept [10]. Figure 1 illustrates an example of the regions of interest delimited by an achievable and unachievable reference point. Our purpose is to extend SMPSO to allow guiding the search according to this kind of preference articulation mechanism.

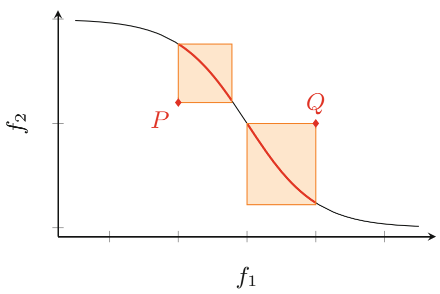

Fig. 1. Examples of the regions of interest delimited by points P (unachievable) and Q (achievable).

[1] https://github.com/jMetal/jMetal.

SMPSO [3] is an algorithm following the classic particle swarm algorithm metaheuristic [11], so it manages a set of solutions or *particles* which are referred to as the *swarm*. The position of particle $\boldsymbol{x}_i$ at the generation t is updated with Eq. (1):

$$\boldsymbol{x}_i(t) = \boldsymbol{x}_i(t-1) + \boldsymbol{v}_i(t) \tag{1}$$

where the factor $\boldsymbol{v}_i(t)$ is known as velocity, and it is defined as:

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

In Eq. (1), $\boldsymbol{x}_{p_i}$ is the best solution that $\boldsymbol{x}_i$ has viewed, $\boldsymbol{x}_{g_i}$ is the best particle (known as the *leader*) that the entire swarm has viewed, w is the inertia weight of the particle and controls the trade-off between global and local influence, 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.

The motivation to develop SMPSO was originated after stating that the MOPSO algorithm [12], a previously proposed multi-objective PSO based on Eqs. 1 and 2, was able of efficiently solve parameter scalable problems [13] but it had difficulties when dealing with the (multi-frontal) ZDT4 problem. We discovered that by applying the *constriction coefficient* (Eq. (3)) obtained from the constriction factor χ originally developed by Clerc and Kennedy (Eq. (2)) in [14], SMPSO could successfully solve that problem with up to 2048 variables. The constriction coefficient is defined as:

$$\chi = \frac{2}{2 - \varphi - \sqrt{\varphi^2 - 4\varphi}} \tag{3}$$

where

$$\varphi = \begin{cases} C_1 + C_2 & \text{if } C_1 + C_2 > 4 \\ 0 & \text{if } C_1 + C_2 \leq 4 \end{cases} \tag{4}$$

Additionally, SMPSO further bounds the accumulated velocity of each variable j (in each particle) by means of the following *velocity constriction* equation:

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

where

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

As commented beforehand, SMPSO adopts the use of an external archive to store the non-dominated solutions and out of which leaders are chosen.

3 Algorithm Proposal

The basic component of SMPSO/RP is the concept of reference point archive (i.e., an external archive with an associated reference point). The basic idea is to modify the strategy for adding new solutions to the external archive, in such a way that only solutions within the area of interest defined by a reference point P are kept. The basic approach is as follows: given a solution a to be inserted, it is first compared with P. If either a dominates P or vice-versa, then a is checked for insertion in the archive as done in the original SMPSO. However, if none of them dominates the other, a is discarded.

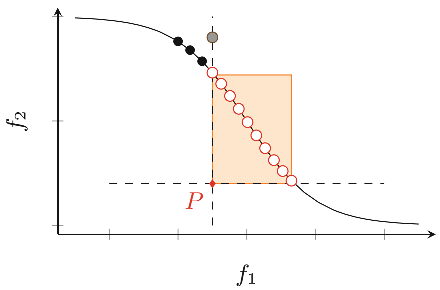

Fig. 2. Example illustrating how a point in the boundary of the region of interest can remain in the reference point archive.

This strategy does not work properly in two scenarios. First, when the archive is empty and only non-dominated solutions regarding P are generated by the search. This scenario results in an empty archive which renders the working behavior of SMPSO impossible, as it may need to select a global leaders from this archive. Our solution is to incorporate the non-dominated solution if the archive is empty. This solution is expected to be removed later by any other solution dominating it.

The second situation has to do with a poor convergence of solutions on any of the ends of the region of interest. The Fig. 2 illustrates this issue. The white points are inside the region of interest defined by P, and the point with a gray background is exactly in the boundary of this region. The gray point is non-dominated regarding the white points and therefore always kept in the archive as it is assigned an infinite crowding distance by the density estimator. However, it is not close to the Pareto front, so convergence is negatively affected. This would not happen if some of the black points on the left would belong to the region of interest, because they dominate the gray point, which would have been either removed or never inserted. Our approach, then, is to insert non-dominated points which are outside the region of interest with a certain probability for the sake of filtering these poorly converged points in the ends of the region of interest (after some pilot tests, we have set this probability to 0.05). These points outside the area of interest are removed later from the archive.

SMPSO/RP can have more than one reference point archive, so the DM can indicate several regions of interest. The working procedure of SMPSO/RP resembles that of SMPSO, except for subtle yet very relevant differences: the leader selection, which take a leader from a randomly selected reference point archive, and all the archives are updated when any particle moves. SMPSO/RP has been implemented in jMetal 5 [15], which provides parallelism support to evaluate all the solutions in a population or swarm in parallel in multi-core systems. As only the evaluations are computed in parallel, linear speed-ups cannot be expected given that the rest of the algorithm is sequential code. However, these scheme has the advantage that no changes in the algorithm are needed.

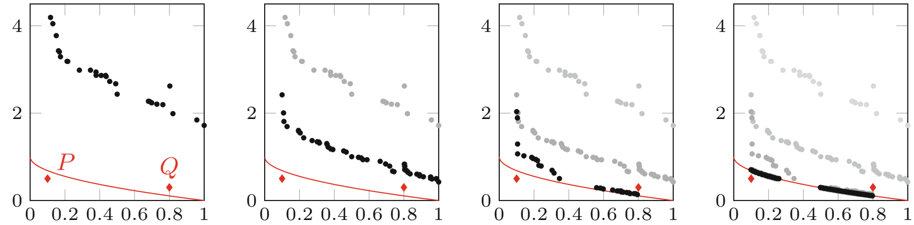

Fig. 3. Example of the front evolution when solving the ZDT1 problem indicating the unachievable reference point P (0.1,0.5) and the achievable one Q (0.8,0.3). The plots depicts the fronts at generations 10, 50, 80, and 120. The population and archive sizes are set to 100.

To illustrate how SMPSO/RP works, Fig. 3 depicts an example of how the computed front evolves over the generations when two reference points, one of each type, have been indicated by the DM.

4 Experimental Setup

In this section, we detail the experimentation we have carried out to assess the performance of SMPSO/RP. We describe first the selected algorithms to be compared with our proposal and their parameter settings. Then, we present the chosen benchmark problems and the reference points that have been specified. Finally, we describe the experimentation methodology.

The regions of interest computed by SMPSO/RP are delimited by the dominance relationship in relation to the reference point. Hence, we have considered three algorithms following the same principle. These algorithms are WASF-GA [6], gSMS-EMOA, and gNSGA-II.

WASF-GA or Weighting Achievement Scalarizing Function Genetic Algorithm uses an *scalarization* approach with weight vectors. In each generation WASF-GA classifies individuals into fronts by taking into account the achievement scalarizing function and the reference point. It also requires to know the

Table 1. Achievable and unachievable points selected for each of the ZDT, DTLZ, and WFG problems.

Problem	Achievable	Unachievable	Problem	Achievable	Unachievable
ZDT1	$(0.80, 0.60)$	$(0.20, 0.40)$	WFG1	$(1.31, 1.61)$	$(0.49, 0.88)$
ZDT2	$(0.80, 0.80)$	$(0.50, 0.30)$	WFG2	$(1.80, 2.91)$	$(0.23, 0.20)$
ZDT3	$(0.30, 0.80)$	$(0.20, 0.00)$	WFG3	$(1.75, 2.55)$	$(0.56, 1.61)$
ZDT4	$(0.99, 0.95)$	$(0.08, 0.25)$	WFG4	$(1.88, 3.71)$	$(0.29, 2.93)$
ZDT6	$(0.78, 0.61)$	$(0.39, 0.21)$	WFG5	$(1.88, 2.46)$	$(0.47, 1.98)$
DTLZ1	$(0.41, 0.36)$	$(0.00, 0.02)$	WFG6	$(1.46, 3.44)$	$(0.28, 0.10)$
DTLZ2	$(0.83, 0.92)$	$(0.07, 0.51)$	WFG7	$(1.17, 3.74)$	$(0.11, 3.03)$
DTLZ3	$(0.87, 1.00)$	$(0.15, 0.42)$	WFG8	$(1.92, 3.60)$	$(0.29, 3.56)$
DTLZ4	$(0.97, 0.59)$	$(0.41, 0.51)$	WFG9	$(1.83, 3.92)$	$(0.81, 2.15)$
DTLZ5	$(0.97, 0.59)$	$(0.03, 0.27)$			
DTLZ6	$(0.76, 0.84)$	$(0.08, 0.48)$			
DTLZ7	$(0.85, 3.88)$	$(0.62, 1.27)$			

ranges of the objective solutions in the Pareto front from the ideal and nadir points, which need to be estimated.

The other chosen algorithms, gNSGA-II and gSMS-EMOA, are variants of the original NSGA-II and SMS-EMOA algorithms modified to incorporate the concept of g-dominance [10]. NSGA-II [4] is by far the most well-known and used multi-objective evolutionary algorithm, and it is characterized by following a generational scheme which applies a non-dominated sorting algorithm and the crowding distance density estimator to promote, respectively, convergence and diversity. SMS-EMOA [16] is a typical representative of indicator-based multi-objective evolutionary metaheuristics; it is based on a steady-state version of NSGA-II but replacing the crowding distance by the hypervolume contribution. None of the algorithms evaluated in this paper requires additional parameter to determine the extent of the region of interest. Algorithms requiring so, like R-NSGA-II [17] or RPSO-SS [18], are out of the scope of this initial analysis.

All the solvers share common parameter settings. The population/swarm size is set to 100. The stopping condition is to compute 25,000 function evaluations. The mutation operator (turbulence in SMPSO/RP) is the polynomial mutation, applied with probability of $1/L$ (being L the number of decision variables of the problem) and a distribution index of 0.20. gNSGA-II, gSMS-EMOA, and WASF-GA apply SBX crossover with a probability of 0.9 and distribution index of 20.0. As these three algorithms only allow indicating a reference point, SMPSO/RP is configured with an external archive with capacity for 100 solutions. WASF-GA generates 100 weight vectors with $\epsilon = 0.01$.

As benchmark problems, we have selected the ZDT [19], DTLZ [20], and WFG [21] families and we have solved them by indicating both an achievable and an unachievable reference point. In this study, we have considered the

Table 2. Median and interquartile range of the hypervolume quality indicator when solving the problems indicating achievable reference points.

	SMPSO/RP	gSMS-EMOA	gNSGAII	WASF-GA
ZDT1	$5.58e-01_{7.6e-05}$	$5.58e-01_{6.6e-05}$	$5.55e-01_{8.9e-04}$	$5.57e-01_{4.7e-04}$
ZDT2	$4.48e-01_{6.9e-05}$	$4.48e-01_{9.4e-05}$	$4.43e-01_{1.2e-03}$	$4.46e-01_{5.7e-04}$
ZDT3	$3.60e-01_{3.9e-05}$	$3.59e-01_{9.7e-05}$	$3.57e-01_{5.4e-04}$	$3.57e-01_{3.2e-04}$
ZDT4	$6.43e-01_{2.3e-04}$	$6.40e-01_{4.3e-03}$	$6.35e-01_{4.9e-03}$	$6.38e-01_{4.2e-03}$
ZDT6	$4.16e-01_{6.3e-05}$	$4.12e-01_{1.4e-03}$	$3.95e-01_{7.3e-03}$	$4.03e-01_{2.4e-03}$
DTLZ1	$4.94e-01_{7.7e-05}$	$0.00e+00_{0.0e+00}$	$0.00e+00_{0.0e+00}$	$4.88e-01_{7.3e-03}$
DTLZ2	$3.96e-01_{1.2e-04}$	$3.96e-01_{1.5e-05}$	$3.94e-01_{3.9e-04}$	$3.96e-01_{2.2e-05}$
DTLZ3	$2.85e-01_{8.7e-05}$	$0.00e+00_{0.0e+00}$	$0.00e+00_{0.0e+00}$	$1.41e-01_{2.0e-01}$
DTLZ4	$4.11e-01_{9.1e-05}$	$4.11e-01_{2.8e-05}$	$4.09e-01_{7.2e-04}$	$4.10e-01_{4.1e-01}$
DTLZ5	$4.12e-01_{9.0e-05}$	$4.13e-01_{1.4e-05}$	$4.11e-01_{5.1e-04}$	$4.12e-01_{4.5e-05}$
DTLZ6	$4.48e-01_{8.3e-05}$	$0.00e+00_{0.0e+00}$	$0.00e+00_{0.0e+00}$	$5.25e-02_{9.0e-02}$
DTLZ7	$3.05e-01_{2.7e-05}$	$3.04e-01_{1.1e-01}$	$3.03e-01_{1.1e-01}$	$3.03e-01_{8.1e-05}$
WFG1	$0.00e+00_{0.0e+00}$	$0.00e+00_{0.0e+00}$	$0.00e+00_{1.3e-02}$	$0.00e+00_{3.5e-03}$
WFG2	$4.74e-01_{3.4e-04}$	$4.74e-01_{1.2e-03}$	$4.73e-01_{1.2e-03}$	$4.72e-01_{1.1e-03}$
WFG3	$4.94e-01_{2.3e-04}$	$4.94e-01_{1.1e-03}$	$4.91e-01_{1.2e-03}$	$4.93e-01_{7.4e-04}$
WFG4	$3.51e-01_{8.8e-03}$	$3.53e-01_{3.1e-05}$	$3.51e-01_{7.0e-04}$	$3.52e-01_{3.2e-04}$
WFG5	$2.52e-01_{2.9e-05}$	$2.52e-01_{2.5e-05}$	$2.51e-01_{2.0e-04}$	$2.51e-01_{2.6e-05}$
WFG6	$4.48e-01_{1.1e-04}$	$3.02e-01_{2.4e-01}$	$3.10e-01_{1.8e-01}$	$3.68e-01_{9.6e-02}$
WFG7	$4.42e-01_{1.8e-04}$	$4.43e-01_{2.6e-04}$	$4.40e-01_{7.8e-04}$	$4.42e-01_{4.2e-04}$
WFG8	$2.56e-01_{7.5e-02}$	$2.26e-01_{1.3e-03}$	$2.26e-01_{1.2e-03}$	$2.26e-01_{5.8e-04}$
WFG9	$3.27e-01_{2.3e-04}$	$3.26e-01_{3.7e-03}$	$3.23e-01_{3.1e-03}$	$3.25e-01_{3.4e-03}$

two-objective formulations of the DTLZ and WFG problems. As reference points, we have chosen the ones defined in [6], summarized in Table 1.

To compare the four metaheuristics, we have executed 30 independent runs per configuration and computed the hypervolume [22] as a quality indicator to measure both the convergence and diversity of the obtained Pareto front approximations. As this indicator needs a reference point to be calculated and the Pareto fronts of all the problems are known, we have removed from the reference fronts all the solutions that fall out of the region delimited by the reference points.

We report in the tables summarizing the results the median and interquartile range (IQR) as measures of central tendency and dispersion, respectively. With the aim of providing these results with statistical confidence (in this study, p-value $= 0.05$), we have applied Friedman's ranking and Holm's post-hoc multicompare tests [23] to know which algorithms are statistically worse than the control one (i.e., the one with the best ranking).

5 Results and Discussion

Table 2 summarizes the obtained results when the indicated reference point is achievable. The cells with dark and light gray background indicate the best and second best hypervolume values, respectively. We observe that SMPSO/RP outperformed the other techniques in 14 out of the 21 evaluated problems, followed by gSMS-EMOA which obtained the best indicator values in 6 problems.

The results yielded when indicating unachievable reference points are included in Table 3. SMPSO/RP is again the best performing algorithm since it

Table 3. Median and interquartile range of the hypervolume quality indicator when solving the problems indicating unachievable reference points.

	SMPSO/RP	gSMS-EMOA	gNSGAII	WASF-GA
ZDT1	$5.19e-01_{7.5e-05}$	$5.19e-01_{2.5e-04}$	$5.14e-01_{1.1e-03}$	$5.17e-01_{6.8e-04}$
ZDT2	$4.53e-01_{3.7e-05}$	$4.53e-01_{1.2e-04}$	$4.48e-01_{9.5e-04}$	$4.51e-01_{4.6e-04}$
ZDT3	$4.91e-01_{2.9e-05}$	$4.90e-01_{6.2e-04}$	$4.87e-01_{2.7e-03}$	$4.88e-01_{4.3e-04}$
ZDT4	$5.69e-01_{2.6e-04}$	$5.62e-01_{5.9e-03}$	$5.58e-01_{7.4e-03}$	$5.62e-01_{5.5e-03}$
ZDT6	$4.30e-01_{4.5e-05}$	$4.25e-01_{7.9e-04}$	$4.10e-01_{3.5e-03}$	$4.16e-01_{2.4e-03}$
DTLZ1	$4.95e-01_{4.9e-05}$	$4.87e-01_{2.3e-02}$	$4.80e-01_{7.9e-02}$	$4.89e-01_{5.8e-03}$
DTLZ2	$3.10e-01_{6.9e-05}$	$3.10e-01_{4.0e-05}$	$3.07e-01_{4.5e-04}$	$3.09e-01_{2.1e-05}$
DTLZ3	$3.19e-01_{2.0e-04}$	$0.00e+00_{0.0e+00}$	$0.00e+00_{0.0e+00}$	$1.67e-01_{2.4e-01}$
DTLZ4	$3.91e-01_{2.4e-04}$	$3.91e-01_{5.6e-05}$	$3.89e-01_{4.4e-04}$	$3.91e-01_{3.3e-05}$
DTLZ5	$2.66e-01_{1.9e-04}$	$2.66e-01_{3.6e-05}$	$2.64e-01_{3.2e-04}$	$2.66e-01_{1.6e-05}$
DTLZ6	$3.11e-01_{1.6e-05}$	$0.00e+00_{0.0e+00}$	$0.00e+00_{0.0e+00}$	$1.55e-01_{5.6e-02}$
DTLZ7	$5.85e-01_{2.1e-05}$	$5.85e-01_{4.3e-05}$	$5.83e-01_{5.2e-04}$	$5.59e-01_{4.8e-05}$
WFG1	$0.00e+00_{6.8e-04}$	$3.28e-02_{1.1e-01}$	$1.66e-01_{1.3e-01}$	$4.77e-01_{3.2e-01}$
WFG2	$5.56e-01_{4.5e-04}$	$5.54e-01_{2.4e-03}$	$5.54e-01_{2.3e-03}$	$5.53e-01_{3.2e-03}$
WFG3	$4.95e-01_{3.4e-04}$	$4.93e-01_{1.1e-03}$	$4.90e-01_{2.2e-03}$	$4.91e-01_{1.9e-03}$
WFG4	$3.59e-01_{4.7e-03}$	$3.66e-01_{5.9e-05}$	$3.62e-01_{6.8e-04}$	$3.65e-01_{4.6e-04}$
WFG5	$2.20e-01_{1.1e-05}$	$2.20e-01_{3.7e-05}$	$2.18e-01_{3.8e-04}$	$2.18e-01_{6.8e-06}$
WFG6	$0.00e+00_{0.0e+00}$	$0.00e+00_{0.0e+00}$	$0.00e+00_{0.0e+00}$	$0.00e+00_{0.0e+00}$
WFG7	$3.70e-01_{6.0e-04}$	$3.70e-01_{1.1e-04}$	$3.67e-01_{9.6e-04}$	$3.68e-01_{5.5e-04}$
WFG8	$3.04e-01_{1.4e-02}$	$2.87e-01_{3.3e-03}$	$2.87e-01_{4.7e-03}$	$2.87e-01_{3.2e-03}$
WFG9	$2.85e-01_{5.1e-04}$	$2.84e-01_{3.3e-03}$	$2.81e-01_{3.3e-03}$	$2.81e-01_{2.3e-03}$

obtained the best hypervolume values in 12 out of 21 the problems. Meanwhile, the second best, gSMS-EMOA, only achieved the best results in 7 problems.

Table 4. Average Friedman's rankings with Holm's Adjusted p-values (0.05) of compared algorithms when solving the problems indicating achievable (left) and unachievable (right) reference points.

Achievable (I_{HV})			Unachievable (I_{HV})		
Algorithm	Fri_{Rank}	$Holm_{Ap}$	Algorithm	Fri_{Rank}	$Holm_{Ap}$
***SMPSO/RP**	**1.52**	-	***SMPSO/RP**	**1.59**	-
gSMS-EMOA	2.09	1.51e−01	gSMS-EMOA	2.16	1.51e−01
WASF-GA	2.76	3.77e−03	WASF-GA	2.71	9.94e−03
gNSGAII	3.62	4.34e−07	gNSGAII	3.52	3.88e−06

As shown in Table 4, SMPSO/RP is the best ranked algorithm according to Friedman's test for achievable, as well as for unachievable reference points. SMPSO/RP is then established as the control algorithm in the post-hoc Holm tests. The adjusted p-values ($Holm_{Ap}$ in Table 4) resulting from these comparisons are lower than the confidence level (0.05) for WASF-GA and gNSGAII, which means that differences between SMPSO/RP and these two algorithms are statistically significant.

To have an insight of the time reductions when running SMPSO/RP in a multi-core system, we have executed it on a machine featuring a quad-core Intel i7 at 2.2. GHz and 16 GB of 1600 MHz DDR3 RAM with hyper-threading enabled. In particular we have performed these execution using 1, 2, 4 and 8 threads when solving the ZDT4 problem and reference point $(0.5, 0.5)$. We have added an idle loop inside the objective functions to increase their computing. The times obtained are 61.5, 45.5, 30.85 and 19.45 s, which mean speed-ups of 1.3, 1.99 and 3.16 with 2, 4, and 8 threads respectively. These speed-ups are expected because, as commented in Sect. 3, only the function evaluations are performed in parallel. Nevertheless, the time reductions are significant and have been achieved with neither major changes in the code nor extra configuration.

6 Use Case

This section describes the application of SMPSO/RP to a real-world problem in the field of structural design. The selected problem aims to optimize the design of a cable stayed-bridge having two objectives (total weight and deformation), encompassing 26 decision variables and 68 constraints [24].

We assume here that a civil engineer is interested in finding the region of the front including solutions with the lowest weights. Without any initial knowledge regarding the weight of different solutions, the starting reference point for the civil engineer is set to $(0.0, 0.0)$ and he/she interactively changes it during the search as information about different computed structures is obtained. A possible execution is shown in Fig. 4 and is described next:

1. Generation 14: Reference point: $(0.0, 0.0)$. The algorithm is looking for a first feasible solution.
2. Generation 64: SMPSO/RP has found a feasible region and a set of non-dominated solutions.

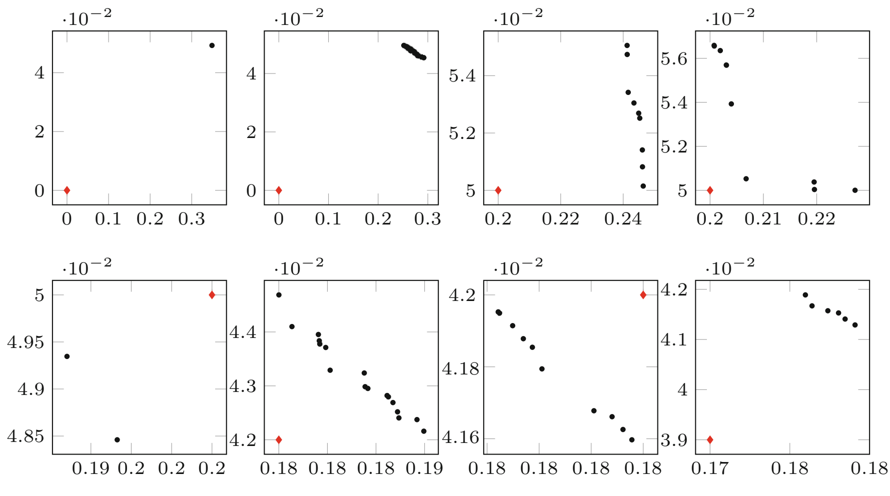

Fig. 4. Example of guiding the search in the structural design problem. Each plot depicts the front at generation 14, 64, 104, 208, 278, 465, 534, and 599, respectively. The reference point changes from $(0.0, 0.0)$ to $(0.2, 0.05)$, $(0.18, 0.042)$, and $(0.17, 0.039)$. The x-axis represents the weight and the y-axis the deformation.

3. Generation 104. The reference point is changed to $(0.2, 0.05)$, which currently is unfeasible.
4. Generation 208. The front is evolving towards the current reference point.
5. Generation 278. The current reference point is feasible and the computed front of solutions is spreading.
6. Generation 465. The reference point is changed to $(0.18, 0.042)$, which is currently unfeasible.
7. Generation 534. The current reference point is feasible and the computed front of solutions is spreading.
8. Generation 599. The reference point is changed to $(0.17, 0.039)$, which is currently unfeasible. At this stage, the engineer is satisfied with the solutions obtained and the optimization process is stopped.

7 Conclusions and Future Research Lines

We introduced SMPSO/RP, an extension of the SMPSO incorporating a preference articulation mechanism based on indicating reference points. Our approach allows changing the reference points interactively and evaluating particles of the swarm in parallel. SMPSO/RP is implemented within the jMetal framework and its source code is freely available.

We have compared our proposal against three other related algorithms on a benchmark composed of 21 problems. Our results indicate that SMPSO/RP achieved the best overall performance when indicating both achievable and unachievable reference points. We have also measured the time reductions that have been achieved when running the algorithm in a multi-core processor platform.

As a line of future work, we are working on adapting SMPSO/RP to efficiently deal with many-objective problems. This implies to rethink the archiving policy and derive novel Pareto density metrics suitable for such problem formulations.

Acknowledgement. This work has been partially funded by Grants TIN2017-86049-R (Spanish Ministry of Education and Science) and P12-TIC-1519 (Plan Andaluz de Investigación, Desarrollo e Innovación). Cristóbal Barba-González is supported by Grant BES-2015-072209 (Spanish Ministry of Economy and Competitiveness). José García-Nieto is the recipient of a Post-Doctoral fellowship of "Captación de Talento para la Investigación" Plan Propio at Universidad de Málaga. Javier Del Ser thanks the Basque Government for its funding support through the EMAITEK program. Carlos A. Coello Coello is supported by CONACyT project no. 221551.

References

1. Coello Coello, C., Lamont, G., van Veldhuizen, D.: Multi-Objective Optimization Using Evolutionary Algorithms, 2nd edn. Wiley, Hoboken (2007)
2. Coello Coello, C.: Handling preferences in evolutionary multiobjective optimization: a survey. In: Proceedings of the IEEE Conference on Evolutionary Computation, ICEC, vol. 1, pp. 30–37 (2000)

3. Nebro, A., Durillo, J., García-Nieto, J., Coello Coello, C., Luna, F., Alba, E.: SMPSO: a new PSO-based metaheuristic for multi-objective optimization. In: IEEE Symposium on Computational Intelligence in Multi-Criteria Decision-Making, MCDM 2009, pp. 66–73. IEEE Press (2009)
4. Deb, K., Pratap, A., Agarwal, S., Meyarivan, T.: A fast and elitist multiobjective genetic algorithm: NSGA-II. IEEE Trans. Evol. Comput. **6**(2), 182–197 (2002)
5. Durillo, J.J., Nebro, A.J.: jMetal: a Java framework for multi-objective optimization. Adv. Eng. Softw. **42**(10), 760–771 (2011)
6. Ruiz, A., Saborido, R., Luque, M.: A preference-based evolutionary algorithm for multiobjective optimization: the weighting achievement scalarizing function genetic algorithm. J. Glob. Optim. **62**(1), 101–129 (2015)
7. Branke, J.: MCDA and multiobjective evolutionary algorithms. In: Greco, S., Ehrgott, M., Figueira, J. (eds.) Multiple Criteria Decision Analysis. ISOR, vol. 233, pp. 977–1008. Springer, New York (2016). https://doi.org/10.1007/978-1-4939-3094-4_23
8. Li, L., Wang, Y., Trautmann, H., Jing, N., Emmerich, M.: Multiobjective evolutionary algorithms based on target region preferences. Swarm Evol. Comput. **40**, 196–215 (2018)
9. Wierzbicki, A.P.: Reference point approaches. In: Gal, T., Stewart, T.J., Hanne, T. (eds.) Multicriteria Decision Making. ISOR, vol. 21, pp. 237–275. Springer, Boston (1999). https://doi.org/10.1007/978-1-4615-5025-9_9
10. Molina, J., Santana, L., Hernández-Díaz, A., Coello Coello, C., Caballero, R.: g-dominance: Reference point based dominance for multiobjective metaheuristics. Eur. J. Oper. Res. **197**(2), 685–692 (2009)
11. Kennedy, J., Eberhart, R.: Particle swarm optimization. In: Proceedings of IEEE International Conference on Neural Networks, pp. 1942–1948 (1995)
12. Sierra, M.R., Coello Coello, C.A.: Improving PSO-based multi-objective optimization using crowding, mutation and ϵ-dominance. In: Coello Coello, C.A., Hernández Aguirre, A., Zitzler, E. (eds.) EMO 2005. LNCS, vol. 3410, pp. 505–519. Springer, Heidelberg (2005). https://doi.org/10.1007/978-3-540-31880-4_35
13. Durillo, J., Nebro, A., Coello Coello, C., Garcia-Nieto, J., Luna, F., Alba, E.: A study of multiobjective metaheuristics when solving parameter scalable problems. IEEE Trans. Evol. Comput. **14**(4), 618–635 (2010)
14. Clerc, M., Kennedy, J.: The particle swarm - explosion, stability, and convergence in a multidimensional complex space. IEEE Trans. Evol. Comput. **6**(1), 58–73 (2002)
15. Nebro, A.J., Durillo, J.J., Vergne, M.: Redesigning the jMetal multi-objective optimization framework. In: Proceedings of the Companion of the Conference on Genetic and Evolutionary Computation (GECCO), pp. 1093–1100 (2015)
16. Beume, N., Naujoks, B., Emmerich, M.: SMS-EMOA: multiobjective selection based on dominated hypervolume. Eur. J. Oper. Res. **181**(3), 1653–1669 (2007)
17. Deb, K., Sundar, J., Ubay, B., Chaudhuri, S.: Reference point based multi-objective optimization using evolutionary algorithm. Int. J. Comput. Intell. Res. **2**(6), 273–286 (2006)
18. Allmendinger, R., Li, X., Branke, J.: Reference point-based particle swarm optimization using a steady-state approach. In: Li, X., et al. (eds.) SEAL 2008. LNCS, vol. 5361, pp. 200–209. Springer, Heidelberg (2008). https://doi.org/10.1007/978-3-540-89694-4_21
19. Zitzler, E., Deb, K., Thiele, L.: Comparison of multiobjective evolutionary algorithms: empirical results. Evol. Comput. **8**(2), 173–195 (2000)

20. Deb, K., Thiele, L., Laumanns, M., Zitzler, E.: Scalable test problems for evolutionary multiobjective optimization. In: Abraham, A., Jain, L., Goldberg, R. (eds.) Evolutionary Multiobjective Optimization. AI&KP, pp. 105–145. Springer, London (2005). https://doi.org/10.1007/1-84628-137-7_6
21. Huband, S., Barone, L., While, L., Hingston, P.: A scalable multi-objective test problem toolkit. In: Coello Coello, C.A., Hernández Aguirre, A., Zitzler, E. (eds.) EMO 2005. LNCS, vol. 3410, pp. 280–295. Springer, Heidelberg (2005). https://doi.org/10.1007/978-3-540-31880-4_20
22. Zitzler, E., Thiele, L.: Multiobjective evolutionary algorithms: a comparative case study and the strength pareto approach. Trans. Evol. Comput. **3**(4), 257–271 (1999)
23. Derrac, J., García, S., Molina, D., Herrera, F.: A practical tutorial on the use of non-parametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms. Swarm Evol. Comput. **1**(1), 3–18 (2011)
24. Zavala, G., Nebro, A.J., Luna, F., Coello Coello, C.: Structural design using multiobjective metaheuristics. Comparative study and application to a real-world problem. Struct. Multidiscip. Optim. **53**(3), 545–566 (2016)