2013 IEEE Congress on Evolutionary Computation
June 20-23, Cancún, México

Standard Particle Swarm Optimisation 2011 at CEC-2013: A baseline for future PSO improvements

Mauricio Zambrano-Bigiarini
Institute for Environment and Sustainability
Joint Research Centre, European Commission
Via E. Fermi 2749, TP261, 21027, Ispra (VA), Italy
Email: Mauricio.Zambrano@jrc.ec.europa.eu

Maurice Clerc
Independent Consultant, France
Email: Maurice.Clerc@WriteMe.com

Rodrigo Rojas
Institute for Environment and Sustainability
Joint Research Centre, European Commission
Via E. Fermi 2749, TP261, 21027, Ispra (VA), Italy
Email: Rodrigo.Rojas@jrc.ec.europa.eu

***Abstract*—In this work we benchmark, for the first time, the latest Standard Particle Swarm Optimisation algorithm (SPSO-2011) against the 28 test functions designed for the Special Session on Real-Parameter Single Objective Optimisation at CEC-2013. SPSO-2011 is a major improvement over previous PSO versions, with an *adaptive random topology* and *rotational invariance* constituting the main advancements. Results showed an outstanding performance of SPSO-2011 for the family of unimodal and separable test functions, with a fast convergence to the global optimum, while good performance was observed for four rotated multimodal functions. Conversely, SPSO-2011 showed the weakest performance for all composition problems (i.e. highly complex functions specially designed for this competition) and certain multimodal test functions. In general, a fast convergence towards the region of the global optimum was achieved, requiring less than 10E+03 function evaluations. However, for most composition and multimodal functions SPSO-2011 showed a limited capability to "escape" from sub-optimal regions. Despite this limitation, a desirable feature of SPSO-2011 was its *scalable* behaviour, which observed up to 50-dimensional problems, i.e. keeping a similar performance across dimensions with no need for increasing the population size. Therefore, it seems advisable that future PSO improvements be focused on enhancing the algorithm's ability to solve non-separable and asymmetrical functions, with a large number of local minima and a second global minimum located far from the true optimum. This work is the first effort towards providing a baseline for a *fair* comparison of future PSO improvements.**

Keywords*—particle swarm optimization, SPSO-2011, CEC-2013, random topology, rotational invariance, benchmark testing, evolutionary computation, optimization.

I. Introduction

Particle Swarm Optimisation (PSO) is a population-based global optimisation technique inspired by the social behaviour of bird flocks looking for corn [1], [2]. This algorithm shares some similarities with other evolutionary computation techniques such as Genetic Algorithms (GA). In PSO, however, the solution-space is explored on the basis of individual- and neighbourhood-based best-known *particle positions*, with no presence of evolutionary operators such as crossover or selection [3].

Since it was first proposed in 1995 [1], [2], PSO has received a surge of attention in the literature given its flexibility, easy computational implementation (programming), low computational requirements, low number of adjustable parameters, and efficiency [4]–[6]. To date, numerous variants of the original PSO algorithm have been proposed in the literature, aimed at improving performance or tackling specific optimisation problems [4], [5]. More often than not, researchers claim to have compared their (improved) version of PSO to the "*standard*" PSO algorithm, but the standard itself seems to differ between different articles. More importantly, a clear need to establish a common benchmark in order to assess the performance of the numerous PSO variants appearing in the literature has been recently suggested [7]–[9]. Therefore, it would be very useful to define a baseline for the assessment of new PSO variants on the basis of a common benchmarking algorithm, which should remain unchanged for a few years at least.

As with any other optimisation algorithm, performance of PSO may vary depending on the class of problems investigated [10]. A comprehensive computational exercise on a wide class of "potential" response surfaces will therefore provide valuable insight on the algorithm's performance. This will facilitate the assessment of potential PSO improvements as well as the benchmarking of new PSO variants on challenging and well-known optimisation functions.

In this work, we take advantage of the new set of benchmark functions designed for CEC-2013, which comprise 28 minimisation problems representing different types of response surface (e.g. unimodal, multimodal, composition functions, separable, non-separable, shifted, rotated), to benchmark the latest Standard PSO [9], namely SPSO-2011. More importantly, we aim to provide a baseline for a *fair* assessment of future PSO improvements, on the basis of publicly available and reproducible algorithm [11], [12] and optimisation functions [13], [14].

Despite closely following the original PSO algorithm [1], [2], SPSO-2011 includes several improvements based on recent theoretical developments, the most important being an *adaptive* *random topology* and *rotational invariance*. SPSO-2011 is not intended to be the *best* PSO variant on the market (particularly, considering that swarm size and "learning" coefficients remain constant along iterations). Instead, it must be considered as the *reference level* to be outperformed by future PSO improvements.

The remainder of this article is arranged as follows: Section II provides details on the basics of PSO and the "Standard PSO" algorithm used for CEC-2013. Section III describes the benchmark functions, the experimental setup and algorithm settings used for solving each function. Results are presented in Section IV, and Section V concludes and summarises the main findings of this work.

II. PSO ALGORITHM

A. Canonical PSO

PSO is an evolutionary algorithm where each individual within the population, known as *particle* in PSO terminology, adjusts its *flying trajectory* in the multi-dimensional search space according to its own previous flying experience together with those of the neighbouring particles in the swarm [3]. The original PSO version, proposed in 1995 [1], [2], starts with a random initialisation of each particle's position and velocity (particle displacement) within the parameter space. When considering a D-dimensional search space, *position* and *velocity* for the i^{th} particle are represented by $\vec{X}_i = x_{i1}, x_{i2}, \ldots, x_{iD}$ and $\vec{V}_i = v_{i1}, v_{i2}, \ldots, v_{iD}$, respectively. The performance of each particle is assessed through a problem-specific *performance measure*, which is the basis for updating $\vec{X}_i$. The best-known position of the i^{th} particle, termed as *personal/previous best*, is represented by $\vec{P}_i = p_{i1}, p_{i2}, \ldots, p_{iD}$, whereas the best-known position within that particle's neighbourhood, termed as *local best* [15], is represented by $\vec{L} = l_1, l_2, \ldots, l_D$. Velocity and position of the i^{th} particle are updated according to the following equations,

$$\vec{V}_i^{t+1} = \omega \vec{V}_i^t + c_1 \vec{U}_1^t \otimes (\vec{P}_i^t - \vec{X}_i^t) + c_2 \vec{U}_2^t \otimes (\vec{L}^t - \vec{X}_i^t) \tag{1a}$$

$$\vec{X}_i^{t+1} = \vec{X}_i^t + \vec{V}_i^{t+1} \tag{1b}$$

where $i = 1, 2, \ldots, N$, with N equal to swarm size, and $t = 1, 2, \ldots, T$, with T equal to the maximum number of iterations. ω is the *inertia weight*, c_1 and c_2 are the *cognitive* and *social* acceleration coefficients, and $\vec{U}_1$ and $\vec{U}_2$ are independent and uniformly distributed random vectors within the range $[0, 1]$ (note that $\otimes$ denotes element-wise vector multiplication).

The inclusion of ω in equation (1a) [16] aims to prevent *swarm explosion*, i.e. an uncontrolled increase of particle velocity. In addition to this, [17] suggest limiting maximum velocity to the range $[-\vec{V}^{max}, \vec{V}^{max}]$ for each dimension, with $\vec{V}^{max} = \vec{X}^{max}$ and $\vec{X}^{max}$ as the limits of the search space. Equations (1a) and (1b) can be considered the present-day *canonical* PSO [18].

B. Topologies

Particles in the swarm interact by defining a common set of links, which controls the exchange of information between particles, known as *swarm topology*. The set of particles "informing" the i^{th} particle is defined as the particle's *neighbourhood*, which includes the particle itself as a member [7], [15], [19].

Traditional topologies for the canonical PSO includes the *gbest* (star), *lbest* (circles), and *von Neumann* [15], [20]. In 2006 [21] proposed an *adaptive random* topology, where each particle randomly informs K particles and itself (the same particle may be chosen several times), with K usually set to 3 [9], [21]. In this topology the *connections* between particles randomly change when the global optimum shows no improvement.

C. Standard PSO 2011

To date, three successive *Standard PSO* algorithms have been defined, which take advantage of the latest theoretical analyses that were available at the time they were designed [8], [9], [22]. Until 2007, the velocity update was carried out *dimension by dimension* (see equation 1a), and the algorithm performance heavily depended on the coordinate system (*rotation sensitivity*) [22], [23]. Figure 1a illustrates this point clearly. It shows that for each particle and at each time step, the *distribution of all next possible positions* (DNPP) is a combination of two D-rectangles with sides parallel to the axes, with uniform probability distribution inside each rectangle.

In contrast, the latest Standard PSO (SPSO-2011) [9], [11] exploits the idea of *rotational invariance* (see Figure 1b). For each particle and at each time step, a centre of gravity ($\vec{G}_i$) is defined around three points: the current position ($\vec{X}_i^t$), a point a little "beyond" the best previous personal position ($\vec{p}_i^t$), and a point a little "beyond" the best previous position in the neighbourhood ($\vec{l}_i^t$), as follows:

$$\vec{p}_i^t = \vec{X}_i^t + c_1 \vec{U}_1^t \otimes (\vec{P}_i^t - \vec{X}_i^t) \tag{2a}$$

$$\vec{l}_i^t = \vec{X}_i^t + c_2 \vec{U}_2^t \otimes (\vec{L}^t - \vec{X}_i^t) \tag{2b}$$

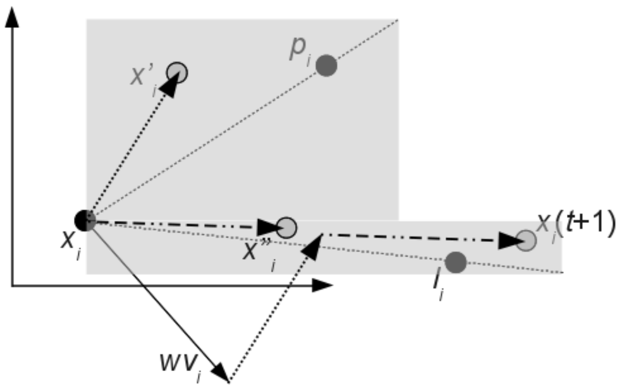
$$\vec{G}_i^t = \frac{(\vec{X}_i^t + \vec{p}_i^t + \vec{l}_i^t)}{3} \tag{2c}$$

A random point x' is then defined in the hypersphere $\mathcal{H}_i(\vec{G}_i^t, ||\vec{G}_i^t - \vec{X}_i^t||)$. Figure 1b shows that the support of the DNPP obtained in this case is a D-dimensional sphere, which is invariant by rotation around its centre, i.e. the DNPP is not modified when the surface response of the function is rotated. The search space resembles a D-dimensional window that allows us "to see" part of the surface response of the function. When that surface is rotated, certain points leave the original window while others appear on it, thus modifying the problem to solve [22]. Therefore, even for a rotational invariant algorithm the performances on both a given problem and its rotated version can be different, because these two problems, geometrically speaking, are usually different themselves.

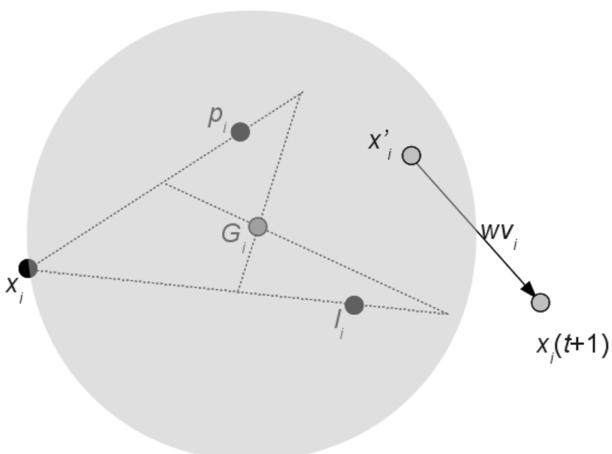
In SPSO-2011, velocity is updated as follows,

$$\vec{V}_i^{t+1} = \omega \vec{V}_i^t + \mathcal{H}_i(\vec{G}_i^t, ||\vec{G}_i^t - \vec{X}_i^t||) - \vec{X}_i^t \tag{3}$$

while particle position is updated following equation (1b).



(a) Geometrical interpretation of the Canonical and Standard PSO algorithms until 2007.



(b) Geometrical interpretation of the Standard PSO 2011 (SPSO-2011) algorithm.

Fig. 1. Distribution of all the next possible positions (DNPP) in PSO. (a) points x_i' and x_i'' are randomly chosen inside two hyper-parallelepipeds parallel to the axes. (b) In SPSO-2011 the point x_i' is chosen at random inside the hyper-sphere $\mathcal{H}_i(\vec{G}_i^t, ||\vec{G}_i^t - \vec{X}_i^t||)$.

In addition, in SPSO-2011 swarm size is user-defined, but a value of 40 is empirically suggested. The algorithm starts by connecting all the particles using a random topology ($K = 3$), while the other elements are initialised as follows:

$$X_i^0 = U(min_d, max_d) \tag{4a}$$

$$V_i^0 = \frac{U(min_d, max_d) - X_i^0}{2} \tag{4b}$$

$$p_i^0 = X_i^0 \tag{4c}$$

$$l_i^0 = min(f(p_i^0)) \tag{4d}$$

where superscript ^{0} refers to initial conditions, and $U(min_d, max_d)$ is a random number in $[min_d, max_d]$ drawn according to a uniform distribution. When a particle "flies" outside the $[min_d, max_d]$ range, each boundary of the search space acts as an absorbing wall, modifying the position of the particle to coincide with the reached boundary and resetting the particle's velocity to zero.

The SPSO-2011 algorithm can be summarised as follows:

for $i = 1$ to N **do** {for each particle in the swarm}
  Initialise particles' positions ($\vec{X}_i$) and velocities ($\vec{V}_i$)
  Initialise personal/previous best, $\vec{P}_i$, and local best, $\vec{G}$
end for
repeat
  for $i = 1$ to N **do**
    Update particle's velocity using equation 3
    Update particle's position using equation 1b
    if $f(\vec{X}_i) < f(\vec{P}_i)$ **then** {minimization of f}
      Update particle's best-known position $\vec{P}_i = \vec{X}_i$
      if $f(\vec{P}_i) < f(\vec{L})$ **then** {minimization of f}
        Update the neighbourhood's best-known position $\vec{L} = \vec{P}_i$
      end if
    end if
  end for
until [nr. of iterations (T) or tolerance error is met]

III. Experimental Setup

The performance of SPSO-2011 was benchmarked against the new set of test functions designed for CEC-2013, which comprise five unimodal (f1-f5), fifteen multimodal (f6-f20), and eight composition (f21-f28) functions, totalling 28 diverse and difficult minimisation problems. The composition functions were specifically designed for this competition, and comprise the sum of three or five unimodal and/or multimodal functions, leading to very challenging properties: multimodal, non-separable, asymmetrical and with different properties around different local optima.

Experiments were carried out for 10-, 30- and 50-dimensional versions of each test function, with a maximum number of function evaluations ($MaxFES$) equal to $10000 * D$. Each experiment was repeated 51 times, in order to (partially) take into account the stochastic nature of evolutionary algorithms [8]. The search space for all the 28 test functions was set to [-100,100]. Function error values ($f(x) - f(x*)$) were stored for every iteration of each optimisation run, however, they were only reported for $(0.01, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0) * MaxFES$ for each experiment ([13]). Error values smaller than $1E-08$ were considered as zero. Each run of the algorithm was designed to finish when either $MaxFES$ was met or the error value was lower than $1E-08$ [13], even in the case of premature convergence. In total, approximately 1.3E+09 function evaluations were computed (28 test functions $\times$ 3 dimensions $\times 10000*D$ function evaluations $\times$ 51 repetitions). For a complete description of the test functions and evaluation criteria used in this work we refer the reader to [13].

Parameter settings for SPSO-2011 were those defined in [9], [11] and consider: a swarm size equal to 40 particles for 10-, 30- and 50-D; random initialisation of particle positions and velocities; random topology with K=3 informants; acceleration coefficients c_1 and c_2 equal to $0.5 + ln(2)$; particle velocity constrained to the range $[-\vec{X}^{max}, \vec{X}^{max}]$; and a constant inertia weight equal to $\omega = 1/(2 * ln(2))$.

Experiments were performed in a 64-bit Intel machine, 64-bit GNU/Linux operative system (RHEL 6, kernel 2.6.32-279.22.1.el6.x86_64), using the R statistical environment 2.15.2 [24]. The SPSO-2011 was run using the open source

TABLE I. SUMMARY STATISTICS FOR THE 10-DIMENSIONAL CASE (D10)

Function	$f(x*)$	Min	Median	Max	Std
f1	-1.400E+03	-1.400E+03	-1.400E+03	-1.400E+03	0.000E+00
f2	-1.300E+03	7.853E+02	3.504E+04	4.755E+05	7.356E+04
f3	-1.200E+03	-1.200E+03	2.670E+05	8.251E+07	1.656E+07
f4	-1.100E+03	2.454E+02	7.769E+03	1.856E+04	4.556E+03
f5	-1.000E+03	-1.000E+03	-1.000E+03	-1.000E+03	3.142E-05
f6	-9.000E+02	-9.000E+02	-8.902E+02	-8.898E+02	4.974E+00
f7	-8.000E+02	-7.974E+02	-7.789E+02	-7.434E+02	1.327E+01
f8	-7.000E+02	-6.798E+02	-6.797E+02	-6.796E+02	6.722E-02
f9	-6.000E+02	-5.987E+02	-5.952E+02	-5.929E+02	1.499E+00
f10	-5.000E+02	-4.999E+02	-4.997E+02	-4.989E+02	2.713E-01
f11	-4.000E+02	-3.970E+02	-3.891E+02	-3.731E+02	5.658E+00
f12	-3.000E+02	-2.970E+02	-2.861E+02	-2.682E+02	6.560E+00
f13	-2.000E+02	-1.946E+02	-1.792E+02	-1.523E+02	9.822E+00
f14	-1.000E+02	2.228E+02	7.338E+02	1.109E+03	2.335E+02
f15	1.000E+02	4.372E+02	8.743E+02	1.705E+03	2.507E+02
f16	2.000E+02	2.002E+02	2.005E+02	2.014E+02	2.457E-01
f17	3.000E+02	3.104E+02	3.189E+02	3.416E+02	5.873E+00
f18	4.000E+02	4.125E+02	4.178E+02	4.365E+02	4.534E+00
f19	5.000E+02	5.003E+02	5.009E+02	5.019E+02	3.886E-01
f20	6.000E+02	6.020E+02	6.034E+02	6.040E+02	4.194E-01
f21	7.000E+02	1.100E+03	1.100E+03	1.100E+03	0.000E+00
f22	8.000E+02	1.206E+03	1.706E+03	2.388E+03	3.431E+02
f23	9.000E+02	1.016E+03	1.810E+03	2.776E+03	3.596E+02
f24	1.000E+03	1.162E+03	1.214E+03	1.222E+03	9.166E+00
f25	1.100E+03	1.300E+03	1.309E+03	1.320E+03	5.943E+00
f26	1.200E+03	1.307E+03	1.400E+03	1.520E+03	5.513E+01
f27	1.300E+03	1.602E+03	1.636E+03	1.898E+03	7.359E+01
f28	1.400E+03	1.500E+03	1.700E+03	2.009E+03	8.362E+01

TABLE II. SUMMARY STATISTICS FOR THE 30-DIMENSIONAL CASE (D30)

Function	$f(x*)$	Min	Median	Max	Std
f1	-1.400E+03	-1.400E+03	-1.400E+03	-1.400E+03	1.875E-13
f2	-1.300E+03	6.790E+04	3.075E+05	7.317E+05	1.667E+05
f3	-1.200E+03	1.172E+06	1.188E+08	2.866E+09	5.243E+08
f4	-1.100E+03	2.616E+04	3.804E+04	5.324E+04	6.702E+03
f5	-1.000E+03	-1.000E+03	-1.000E+03	-1.000E+03	4.909E-05
f6	-9.000E+02	-8.998E+02	-8.717E+02	-8.210E+02	2.825E+01
f7	-8.000E+02	-7.498E+02	-7.131E+02	-6.618E+02	2.107E+01
f8	-7.000E+02	-6.793E+02	-6.791E+02	-6.790E+02	5.893E-02
f9	-6.000E+02	-5.789E+02	-5.716E+02	-5.617E+02	4.426E+00
f10	-5.000E+02	-4.999E+02	-4.997E+02	-4.993E+02	1.478E-01
f11	-4.000E+02	-3.493E+02	-2.916E+02	-2.139E+02	2.740E+01
f12	-3.000E+02	-2.512E+02	-2.055E+02	-9.006E+01	3.539E+01
f13	-2.000E+02	-8.763E+01	-2.322E+00	1.088E+02	3.862E+01
f14	-1.000E+02	2.856E+03	3.923E+03	5.887E+03	6.194E+02
f15	1.000E+02	2.042E+03	3.904E+03	5.256E+03	6.938E+02
f16	2.000E+02	2.004E+02	2.014E+02	2.021E+02	3.588E-01
f17	3.000E+02	3.727E+02	4.152E+02	4.852E+02	2.018E+01
f18	4.000E+02	4.768E+02	5.168E+02	5.711E+02	2.460E+01
f19	5.000E+02	5.028E+02	5.090E+02	5.305E+02	4.418E+00
f20	6.000E+02	6.105E+02	6.140E+02	6.145E+02	1.109E+00
f21	7.000E+02	9.000E+02	1.000E+03	1.144E+03	6.796E+01
f22	8.000E+02	3.683E+03	5.151E+03	6.694E+03	7.670E+02
f23	9.000E+02	4.021E+03	5.663E+03	7.883E+03	8.227E+02
f24	1.000E+03	1.238E+03	1.264E+03	1.293E+03	1.246E+01
f25	1.100E+03	1.377E+03	1.400E+03	1.424E+03	1.045E+01
f26	1.200E+03	1.400E+03	1.540E+03	1.594E+03	8.240E+01
f27	1.300E+03	2.058E+03	2.326E+03	2.571E+03	1.119E+02
f28	1.400E+03	1.500E+03	1.700E+03	4.276E+03	4.761E+02

implementation provided by the *hydroPSO* R package v0.3-0 [12], and benchmarking functions were evaluated using the *cec2013* R package v0.1-4 [14]. The latter provides R wrappers to the original C code available on the CEC-2013 web page (http://www.ntu.edu.sg/home/EPNSugan/index_files/CEC2013/CEC2013.htm). Random numbers were generated in R with the Mersenne-Twister pseudo random number generator [25].

IV. RESULTS

Table I reports basic summary statistics for the optimum values found by SPSO-2011 for all functions in the 10-dimensional case. From this table we can see that SPSO-2011 succeeded in finding, at least once, the true optimum value for only the unimodal functions f1, f3, f5; the multimodal function f6; and none of the composition functions. However, for the multimodal functions f10, f16, f19 and f20 the optimum values found by SPSO-2011 were reasonably close to the true optimum. At the same time, the lowest standard deviation values were achieved for functions f1, f5, f8, f10, f16, f19, f20 and f21. It is worth noting that SPSO-2011 was both accurate and precise for the multimodal functions f10, f16, f19 and f20, even it if did not succeed in finding the true optimum within the requested error tolerance ($1.0E-08$). For the composite function f21, SPSO-2011 was only precise, converging into a sub-optimal region far from the true optimum.

Figure 2 shows the evolution of the median error ($f(x) - f(x*)$) against the number of function evaluations for D-10, where the median was computed over the 51 repetitions of each experiment. This figure shows that SPSO-2011 quickly converged to the true optimum value for the unimodal and separable functions f1 and f5 (ca. 10E+03 function evaluations). However, convergence was much faster for the shifted sphere (f1) than for f5, which is most likely due to the effect of the different sensitivities of the variables in f5. For the unimodal and non-separable functions (f2, f3, f4), SPSO-2011 also showed an initial fast convergence to the region of the global optimum but, after approximately 5E+03 function evaluations, only minor improvements in the optimised values were observed, thus failing to reach the tolerance error before $MaxFES$ was achieved. For the multimodal functions, the best performance of SPSO-2011 was achieved for f10, f16, f19 and f20 (with two of them belonging to the Griewank family), while the worst results were obtained for f14 and f15 (indistinguishable from each other), which belong to the Schwefel's family. For the shifted-rotated Ackley function (f8), which is characterised by a true optimum located in an extremely narrow valley, SPSO-2011 did not move properly after the initial random sampling, thus failing to achieve better values in successive function evaluations. The poorest performance of SPSO-2011 was obtained for all composition functions (f21-f28) where the true optimum value was never reached within the given error tolerance ($1E-08$). However, functions f24, f25 and f26 were much closer to the global optimum compared to the remaining composition functions.

Table II reports basic summary statistics for the optimum values found by SPSO-2011 for all functions in the 30-dimensional case. From this table it is clear that SPSO-2011 found, at least once, the true optimum value (see [13]) for only the unimodal functions f1 and f5; and for none of the multimodal or composite functions. However, for the multimodal functions f6, f10, f16, f19 and f20, the optimum values found were, again, close to the true optimum. At the same time, the lowest standard deviation values were achieved for functions f1, f5, f8, f10, f16, f19 and f20.These results are consistent with those obtained in the 10-dimensional case.

Similarly, Table III presents summary statistics for the 50-dimensional case. These results are consistent with those obtained for the 10- and 30-dimensional cases. However, the multimodal functions f19 (Expanded Griewank plus Rosenbrock) and f20 (Expanded Scaffer's F6) showed a lower level of performance.

Figures 3 and 4 show that for the 30- and 50-dimensional problems the convergence of SPSO-2011 is similar to the 10-dimensional case. This indicates a *"scalable"* behaviour of SPSO-2011 up to D=50, i.e. the ability to maintain a similar level of performance for higher dimensions without increasing the population size. This highlights the efficiency of SPSO-2011 for up to 50-dimensional problems, which is particularly relevant for computationally expensive function evaluations (e.g. environmental models). This scalable behaviour is consistent with results reported in [12] for 10-, 20- and 30-dimensional problems, using a different set of benchmarking functions.

The algorithm's complexity [13] is shown in Table IV for 10, 30 and 50 dimensions. The latter provides an indication of the additional time required for optimisation with an increasing number of dimensions.

V. Conclusions

In this work we benchmarked the latest version of the Standard Particle Swarm Optimisation algorithm, namely SPSO-2011, against the 28 functions designed for the Special Session on Real-Parameter Single Objective Optimisation at CEC-2013. We aimed to provide a minimum reference level for assessing the effects of future improvements to the PSO algorithm, on the basis of the most recent and challenging benchmark functions available and the latest standard PSO.

The best performance of SPSO-2011 was obtained for unimodal and separable functions, while values close to the true optimum were achieved for five multimodal functions. In contrast, the weakest performance was obtained for all of the composition functions, and the f14 and f15 multimodal functions. These latter two are characterised by being non-separable, asymmetrical, with a large number of local minima, and with a second-best local minima located far from the global optimum.

Results indicate that SPSO-2011 is able to quickly converge towards the region of the global optimum, even for shifted and rotated functions. For most of the multimodal and composition functions, however, SPSO-2011 achieves its best possible value within a "limited" number of function evaluations (ca. 10E+03). Despite this fast convergence, all the particles in the swarm continue to "fly" over the attraction zone, without converging towards the true global optimum (stagnation). Unnecessary model runs can be easily avoided by defining a small relative error as an additional stopping criterion.

SPSO-2011 showed a desirable "scalable" behaviour, i.e. maintaining a similar performance across all dimensions (up to $D = 50$) without increasing the population size. The latter renders SPSO-2011 highly suitable for higher dimensional problems where computational time might be a constraint. This has been verified in [12] against other global optimisation algorithms such as Shuffled Complex Evolution [26] and Differential Evolution [27]. It is worth noting that the number of particles used in this work (40, as suggested in [9]), is not large enough for exploring high-dimensional parameter spaces ($D > 30$). Therefore, future research assessing the performance of SPSO-2011 using different numbers of particles in several dimensional spaces may prove to be useful for a more rigorous definition of scalability.

New developments of SPSO-2011 should be targeted to effectively address non-separable, asymmetrical and multimodal functions as well as to allow the algorithm to "escape" from stagnation once premature convergence towards sub-optimal solutions has been detected. It is worth noting that strategies (partially) addressing these problems have been already proposed in the literature, and some of them are readily available in current software packages (e.g. *hydroPSO* R package [12]). However, to the best of our knowledge, these improvements have not yet been tested with SPSO-2011.

The 28 benchmark functions used in this work are newly designed and therefore results obtained in this article cannot be compared directly to any other previous work. We hope the results obtained here will provide a common ground to facilitate the comparison between future PSO improvements or other optimisation algorithms.

TABLE III. Summary statistics for the 50-dimensional case (D50)

Function	$f(x*)$	Min	Median	Max	Std
f1	-1.400E+03	-1.400E+03	-1.400E+03	-1.400E+03	3.183E-13
f2	-1.300E+03	3.776E+05	6.785E+05	1.126E+06	1.873E+05
f3	-1.200E+03	1.995E+07	4.365E+08	5.711E+09	9.471E+08
f4	-1.100E+03	3.113E+04	4.987E+04	7.704E+04	8.717E+03
f5	-1.000E+03	-1.000E+03	-1.000E+03	-1.000E+03	5.405E-05
f6	-9.000E+02	-8.816E+02	-8.565E+02	-7.541E+02	2.405E+01
f7	-8.000E+02	-7.439E+02	-7.136E+02	-6.731E+02	1.527E+01
f8	-7.000E+02	-6.790E+02	-6.789E+02	-6.788E+02	4.254E-02
f9	-6.000E+02	-5.548E+02	-5.460E+02	-5.327E+02	6.744E+00
f10	-5.000E+02	-4.999E+02	-4.996E+02	-4.986E+02	2.376E-01
f11	-4.000E+02	-2.498E+02	-1.702E+02	-3.983E+01	4.183E+01
f12	-3.000E+02	-1.378E+02	-6.519E+01	5.022E+01	4.870E+01
f13	-2.000E+02	1.196E+02	2.284E+02	3.969E+02	6.219E+01
f14	-1.000E+02	5.408E+03	7.161E+03	8.935E+03	8.526E+02
f15	1.000E+02	5.778E+03	8.022E+03	1.128E+04	1.140E+03
f16	2.000E+02	2.014E+02	2.020E+02	2.033E+02	3.865E-01
f17	3.000E+02	5.080E+02	6.106E+02	7.975E+02	6.616E+01
f18	4.000E+02	5.695E+02	6.914E+02	8.625E+02	6.240E+01
f19	5.000E+02	5.170E+02	5.372E+02	5.635E+02	1.198E+01
f20	6.000E+02	6.199E+02	6.227E+02	6.245E+02	1.194E+00
f21	7.000E+02	9.000E+02	1.536E+03	1.822E+03	3.042E+02
f22	8.000E+02	7.356E+03	9.718E+03	1.297E+04	1.404E+03
f23	9.000E+02	8.670E+03	1.126E+04	1.413E+04	1.350E+03
f24	1.000E+03	1.306E+03	1.344E+03	1.381E+03	1.688E+01
f25	1.100E+03	1.458E+03	1.502E+03	1.578E+03	2.048E+01
f26	1.200E+03	1.400E+03	1.628E+03	1.676E+03	9.063E+01
f27	1.300E+03	2.548E+03	2.976E+03	3.343E+03	1.638E+02
f28	1.400E+03	1.800E+03	1.800E+03	5.746E+03	1.304E+03

TABLE IV. Algorithm complexity

Dimension	T0	T1	$\hat{T}2$	$(\hat{T}2$-T1)/T0
10	3.4116	169.4895	187.1347	5.1720
30	3.4330	172.9095	192.9653	5.8421
50	3.4353	175.5773	196.6957	6.1474

Acknowledgment

The authors would like to thank Yasser Fernández for providing the R wrappers for the 28 benchmark functions defined for CEC-2013, which were originally implemented for C and Matlab only.

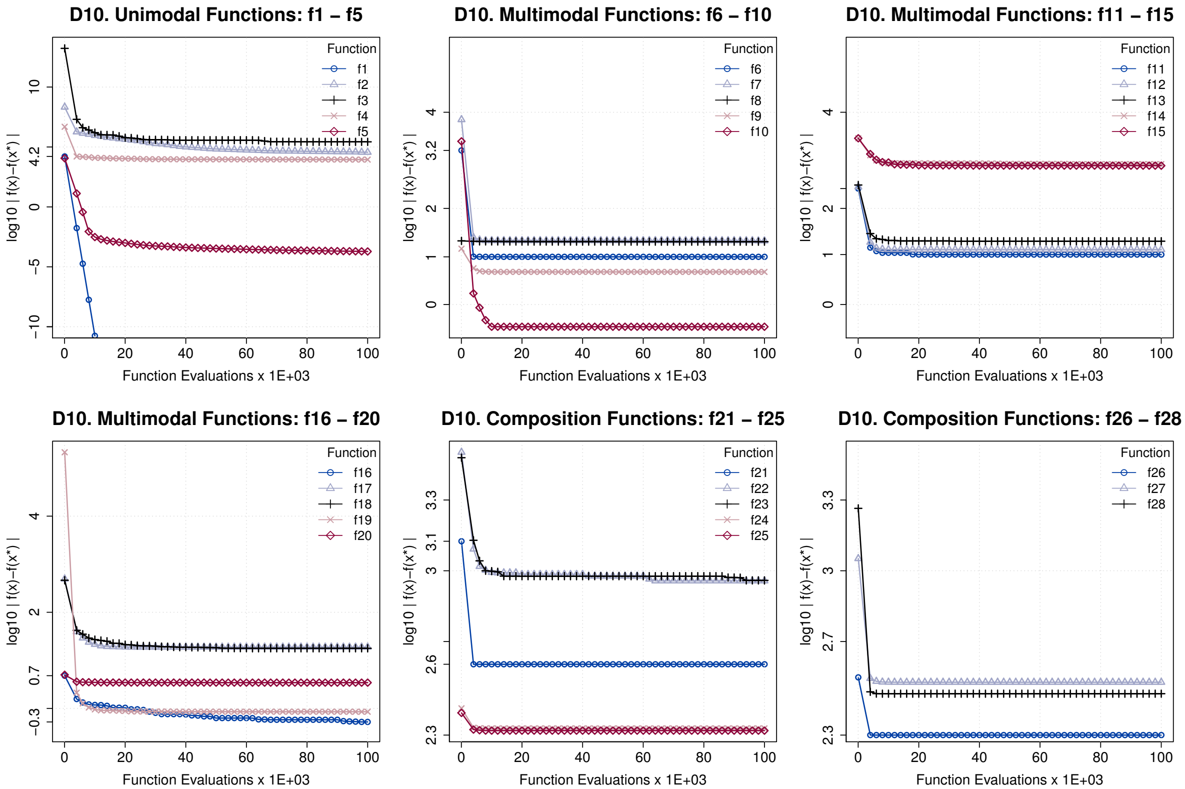


Fig. 2. Evolution of the median error $(f(x) - f(x*))$ against the number of function evaluations in the 10-dimensional case (D10). All multimodal functions share a common vertical scale, as do all composition functions.

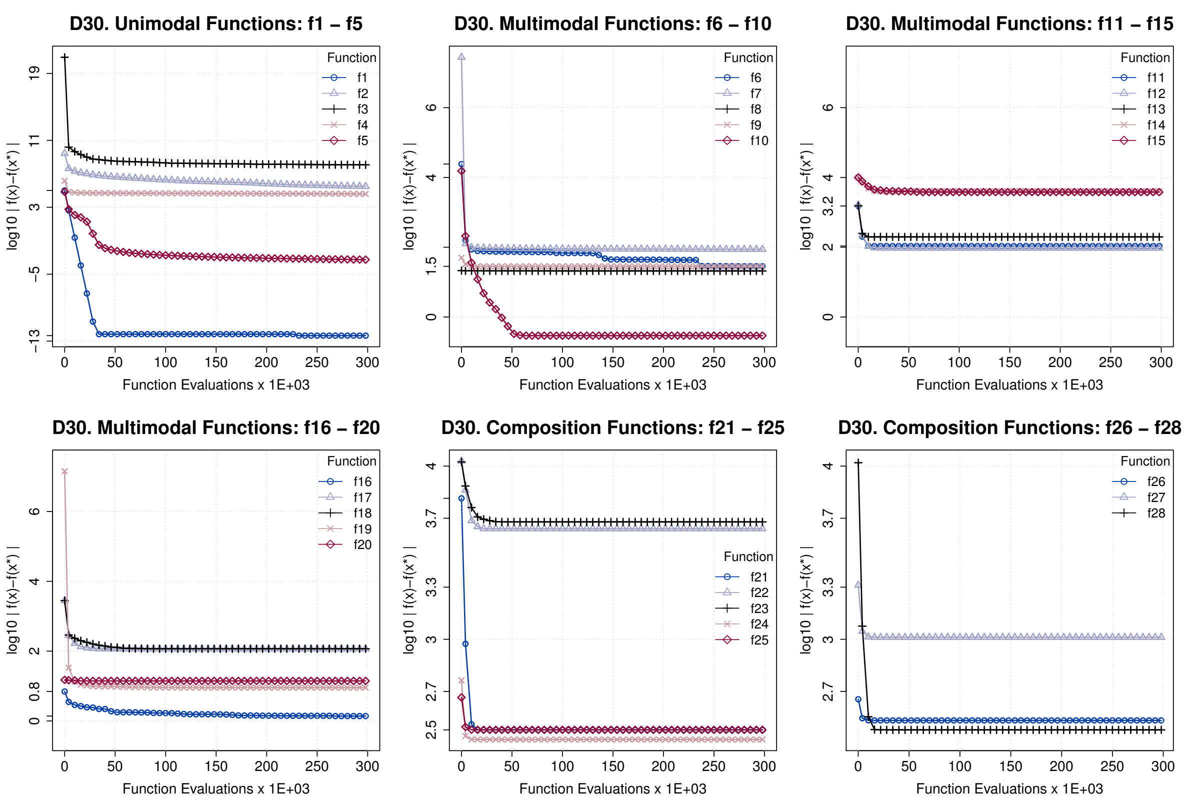


Fig. 3. Evolution of the median error $(f(x) - f(x*))$ against the number of function evaluations in the 30-dimensional case (D30). All multimodal functions share a common vertical scale, as do all composition functions.

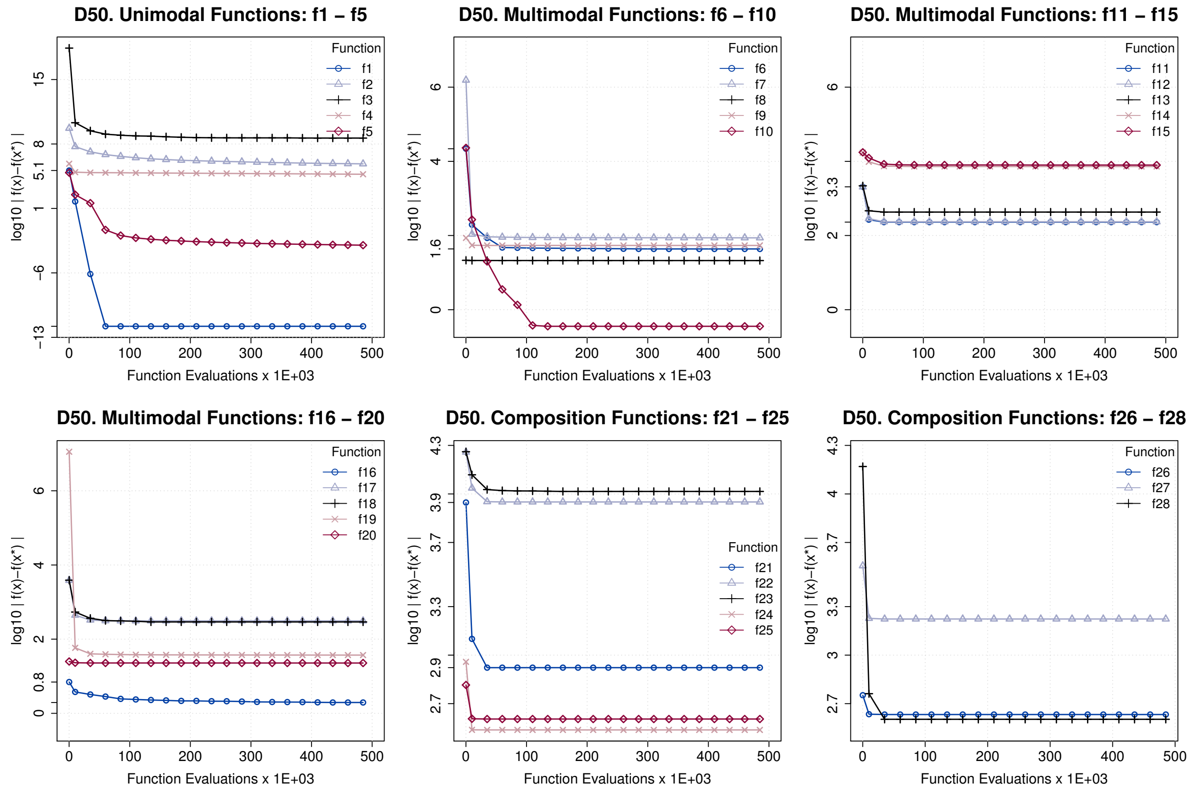


Fig. 4. Evolution of the median error $(f(x) - f(x*))$ against the number of function evaluations in the 10-dimensional case (D50). All multimodal functions share a common vertical scale, as do all composition functions.

References

[1] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings IEEE International Conference on Neural Networks, 1995*, vol. 4, nov/dec 1995, pp. 1942–1948.

[2] R. Eberhart and J. Kennedy, "A new optimizer using particle swarm theory," in *Micro Machine and Human Science, 1995. MHS '95., Proceedings of the Sixth International Symposium on*, oct 1995, pp. 39–43.

[3] R. C. Eberhart and Y. Shi, "Comparison between genetic algorithms and particle swarm optimization," in *Evolutionary Programming VII*, V. Porto, N. Saravanan, D. Waagen, and A. Eiben, Eds. Springer Berlin / Heidelberg, 1998, vol. 1447, pp. 611–616.

[4] T. Chen and T. Chi, "On the improvements of the particle swarm optimization algorithm," *Advances in Engineering Software*, vol. 41, no. 2, pp. 229–239, 2010.

[5] R. Poli, J. Kennedy, and T. Blackwell, "Particle swarm optimization," *Swarm Intelligence*, vol. 1, no. 1, pp. 33–57, 2007.

[6] R. Eberhart and Y. Shi, "Particle swarm optimization: developments, applications and resources," in *Proceedings of the 2001 Congress on Evolutionary Computation*, vol. 1, 2001, pp. 81–86.

[7] M. Clerc, "Back to random topology," Tech. Rep., 2007, clerc.maurice.free.fr/pso/random_topology.pdf. [Online. Last accessed 03-Sep-2012].

[8] ——, "A method to improve Standard PSO," Tech. Rep. MC2009-03-13, 2009, http://clerc.maurice.free.fr/pso/Design_efficient_PSO.pdf. [Online. Last accessed: 14-Mar-2013].

[9] ——, "Standard Particle Swarm Optimisation," Particle Swarm Central, Tech. Rep., 2012, http://clerc.maurice.free.fr/pso/SPSO_descriptions.pdf. [Online. Last accessed 24-Sep-2012].

[10] ——, "From theory to practice in Particle Swarm Optimization," in *Handbook of Swarm Intelligence*, ser. Adaptation, Learning, and Optimization, B. K. Panigrahi, Y. Shi, M.-H. Lim, L. M. Hiot, and Y. S. Ong, Eds. Springer Berlin Heidelberg, 2010, vol. 8, pp. 3–36. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-17390-5_1

[11] PSC, "Particle Swarm Central," 2013, http://www.particleswarm.info/. [Online. Last accessed 14-Mar-2013].

[12] M. Zambrano-Bigiarini and R. Rojas, "A model-independent Particle Swarm Optimisation software for model calibration," *Environmental Modelling & Software*, vol. 43, pp. 5–25, 2013.

[13] J. J. Liang, B.-Y. Qu, P. N. Suganthan, and A. G. Hernández-Díaz, "Problem definitions and evaluation criteria for the CEC 2013 special session and competition on real-parameter optimization," Technical Report 201212, Computational Intelligence Laboratory. Zhengzhou University, Zhengzhou China and Technical Report, Nanyang Technological University, Singapore, Tech. Rep., Jan 2013, http://www.ntu.edu.sg/home/EPNSugan/index_files/CEC2013/CEC2013.htm. [Last accessed 12-Feb-2013].

[14] M. Zambrano-Bigiarini and Y. G. Fernandez, *cec2013: Benchmark Functions for the Special Session and Competition on Real-Parameter Single Objective Optimization at CEC-2013*, 2013, R package version 0.1-4. [Online]. Available: http://CRAN.R-project.org/package=cec2013

[15] J. Kennedy, "Small worlds and mega-minds: effects of neighborhood topology onparticle swarm performance," in *Proceedings of the 1999 Congress on Evolutionary Computation, 1999*, vol. 3, 1999, pp. 3 vol. (xxxvii+2348).

[16] Y. Shi and R. Eberhart, "A modified particle swarm optimizer," in *Evolutionary Computation Proceedings, 1998. IEEE World Congress on Computational Intelligence., The 1998 IEEE International Conference on*, 1998, pp. 69–73.

[17] R. Eberhart and Y. Shi, "Comparing inertia weights and constriction factors in particle swarm optimization," in *Evolutionary Computation, 2000. Proceedings of the 2000 Congress on*, vol. 1, 2000, pp. 84–88.

[18] J. Kennedy, "Swarm intelligence," in *Handbook of Nature-Inspired and Innovative Computing*, A. Zomaya, Ed. Springer US, 2006, pp. 187–219.

[19] J. Kennedy, R. Eberhart, and Y. Shi, *Swarm Intelligence*, ser. The Morgan Kaufmann Series in Evolutionary Computation. San Francisco,

CA 94104-3205, USA: Morgan Kaufmann Publishers Inc., 2001, ch. Variations and Comparisons.

[20] J. Kennedy and R. Mendes, "Population structure and particle swarm performance," in *Proceedings of the 2002 Congress on Evolutionary Computation, CEC '02*, May 2002, pp. 1671–1676.

[21] M. Clerc, *Particle Swarm Optimization*. ISTE (International Scientific and Technical Encyclopedia), 2006.

[22] ——, "Beyond standard particle swarm optimisation," *International Journal of Swarm Intelligence Research*, vol. 1, no. 4, pp. 46–61, 2010.

[23] W. M. Spears, D. T. Green, and D. F. Spears, "Biases in Particle Swarm Optimization," *International Journal of Swarm Intelligence Research*, vol. 1, no. 2, pp. 34–57, 2010.

[24] R Core Team, *R: A Language and Environment for Statistical Computing*, R Foundation for Statistical Computing, Vienna, Austria, 2013, ISBN 3-900051-07-0. [Online]. Available: http://www.R-project.org/

[25] M. Matsumoto and T. Nishimura, "Mersenne Twister: a 623-dimensionally equidistributed uniform pseudo-random number generator," *ACM Transactions on Modeling and Computer Simulation*, vol. 8, no. 1, pp. 3–30, 1998.

[26] Q. Duan, S. Sorooshian, and H. Gupta, "Effective and efficient global optimization fro conceptual ranfall–runoff models," *Water Resources Research*, vol. 28, no. 4, pp. 1015–1031, 1992.

[27] R. Storn and K. Price, "Differential Evolution- a simple and efficient adaptive scheme for global optimization over continuous spaces," in *Technical Report TR-95-012*, 1995.