Krassimir Markov, Vitalii Velychko,
Lius Fernando de Mingo Lopez, Juan Casellanos
(editors)

New Trends
in
Information Technologies

I T H E A
SOFIA
2010

Krassimir Markov, Vitalii Velychko, Lius Fernando de Mingo Lopez, Juan Casellanos (ed.)
New Trends in Information Technologies

ITHEA®

Sofia, Bulgaria, 2010

ISBN 978-954-16-0044-9

First edition

Recommended for publication by The Scientific Concil of the Institute of Information Theories and Applications FOI ITHEA

This book maintains articles on actual problems of research and application of information technologies, especially the new approaches, models, algorithms and methods of membrane computing and transition P systems; decision support systems; discrete mathematics; problems of the interdisciplinary knowledge domain including informatics, computer science, control theory, and IT applications; information security; disaster risk assessment, based on heterogeneous information (from satellites and in-situ data, and modelling data); timely and reliable detection, estimation, and forecast of risk factors and, on this basis, on timely elimination of the causes of abnormal situations before failures and other undesirable consequences occur; models of mind, cognizers; computer virtual reality; virtual laboratories for computer-aided design; open social info-educational platforms; multimedia digital libraries and digital collections representing the European cultural and historical heritage; recognition of the similarities in architectures and power profiles of different types of arrays, adaptation of methods developed for one on others and component sharing when several arrays are embedded in the same system and mutually operated.

It is represented that book articles will be interesting for experts in the field of information technologies as well as for practical users.

General Sponsor: Consortium FOI Bulgaria (www.foibg.com).

Printed in Bulgaria

ISBN 978-954-16-0044-9

C\o Jusautor, Sofia, 2010

BENCHMARK OF PSO-DE USING BBOB 2010

Nuria Gómez Blas, Luis F. de Mingo

Abstract: *As an example, we benchmark the Particle Swarm Optimization algorithm with a Differential Evolution on the noisefree Black Box Optimization Benchmark 2010 testbed. Each candidate solution is sampled uniformly in* $[-5, 5]^D$ *, where D denotes the search space dimension, and the evolution is performed with a classical PSO algorithm and a classical DE/x/1 algorithm according to a random threshold. The maximum number of function evaluations is chosen as* 10^5 *times the search space dimension. This paper shows how to evaluate the performance of a given optimization algorithm a using the BBOB 2010.*

Keywords: *Benchmarking, Black-box optimization, Direct search, Evolutionary computation, Particle Swarm Optimizacin, Differential Evolution*

Categories: *G.1.6 [Numerical Analysis]: Optimization-global optimization, unconstrained optimization ; F.2.1 [Analysis of Algorithms and Problem Complexity]: Numerical Algorithms and Problems.*

Introduction

Particle swarm optimization (PSO) is a global optimization algorithm for dealing with problems in which a best solution can be represented as a point or surface in an n-dimensional space. Hypotheses are plotted in this space and seeded with an initial velocity, as well as a communication channel between the particles. Particles then move through the solution space, and are evaluated according to some fitness criterion after each timestep. Over time, particles are accelerated towards those particles within their communication grouping which have better fitness values. The main advantage of such an approach over other global minimization strategies such as simulated annealing is that the large number of members that make up the particle swarm make the technique impressively resilient to the problem of local minima [7, 8, 9].

Equations used in the particle swarm optimization training process are the following ones, where c1 and c2 are two positive constants, R1 and R2 are two random numbers belonging to [0, 1] and w is the inertia weight. This equations define how the genotype values are changing along iterations.

$$v_{in}(t+1) = wv_{in}(t) + c_1R_1(p_{in} - x_{in}(t)) + c_2R_2(p_{gn} - x_{in}(t))$$

$$x_{in}(t+1) = x_{in}(t) + v_{in}(t+1)$$

Previous equations will modified the network weights till a stop conditions is achieved, that is, a lower mean squared error or a maximum number of iterations is reached.

Differential Evolution (DE) is an evolutionary algorithm [10, 11, 12] that uses a differential mutation procedure that consists in the addition of the weighted difference of two population vectors to a third vector. Many variants of the differential mutation procedure exists. Choosing between these variants and setting parameters requires preliminary testing as [11] admits that the results of the algorithm are dependent on the chosen strategy and the choice of parameter. DE/local-to-best/1 is a variant where instead of the base vector x_{i1} being chosen in the population vector, it is chosen to lie between the vector considered and the best vector so far, thus the update of the velocity is written as follows, where F is a constant in the range [0, 2]:

$$v_i = x_i + F(x_{\text{best}} - x_i) + F(x_{i_2} - x_{i_3}),$$

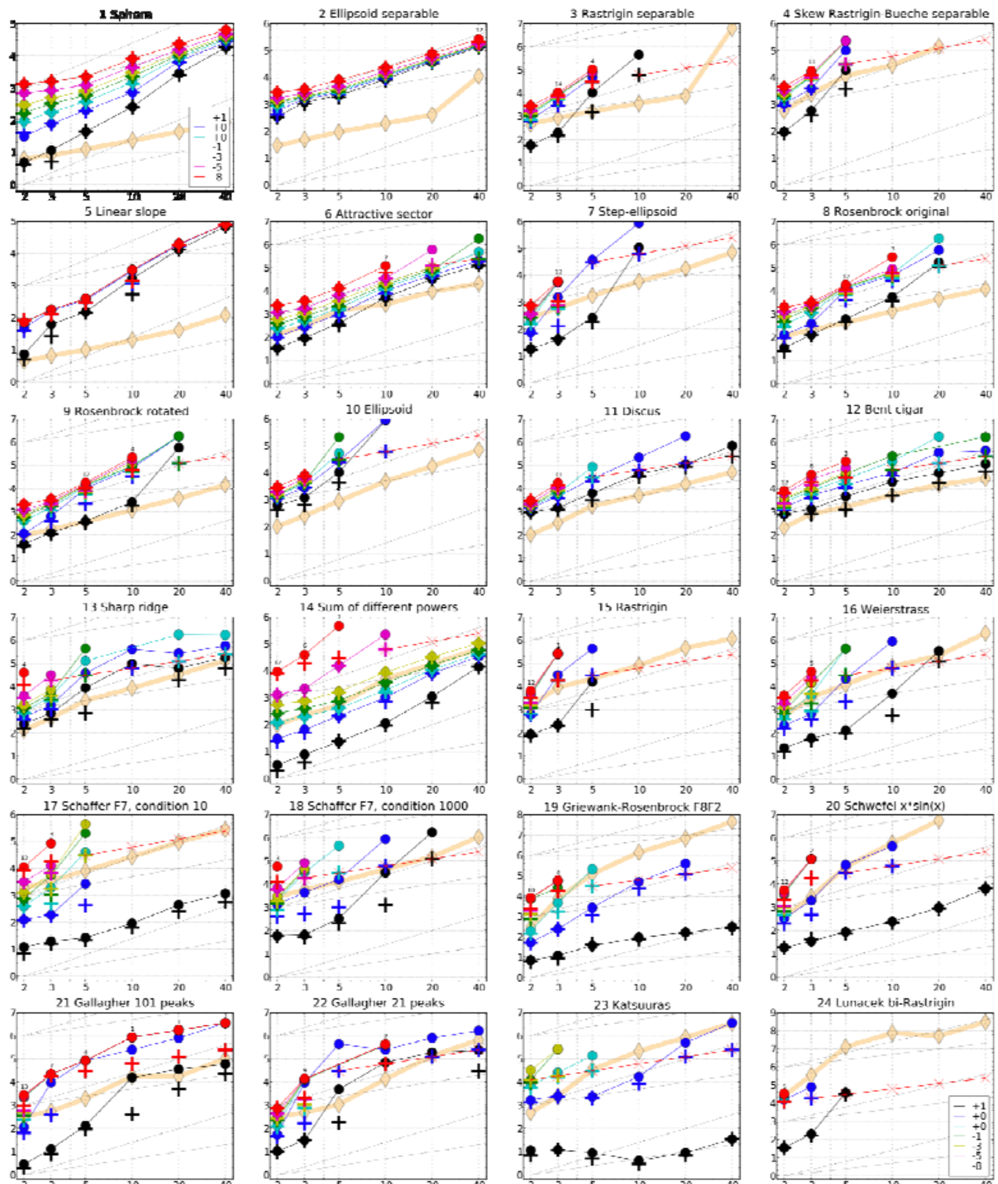

Figure 1: Expected Running Time (ERT, ●) to reach $f_{opt} + \Delta f$ and median number of f-evaluations from successful trials (+), for $\Delta f = 10^{\{+1,0,-1,-2,-3,-5,-8\}}$ (the exponent is given in the legend of f_1 and f_{24}) versus dimension in log-log presentation. For each function and dimension, ERT(Δf) equals to #FEs(Δf) divided by the number of successful trials, where a trial is successful if $f_{opt} + \Delta f$ was surpassed. The #FEs(Δf) are the total number (sum) of f-evaluations while $f_{opt} + \Delta f$ was not surpassed in the trial, from all (successful and unsuccessful) trials, and f_{opt} is the optimal function value. Crosses (×) indicate the total number of f-evaluations, #FEs($-\infty$), divided by the number of trials. Numbers above ERT-symbols indicate the number of successful trials. Y-axis annotations are decimal logarithms. The thick light line with diamonds shows the single best results from BBOB-2009 for $\Delta f = 10^{-8}$. Additional grid lines show linear and quadratic scaling.

Method

We have used a uniform sampling in $[-5, 5]^D$, where D denotes the dimension of the search space. The experiments according to [3] on the benchmark functions given in [2, 4] have been conducted using a C-code. A maximum of $10^5 \times D$ function evaluations has been used.

Δf	f_1 in 5-D, N=15, mFE=6062 #	ERT	10%	90%	RT_{succ}	f_1 in 20-D, N=15, mFE=35078 #	ERT	10%	90%	RT_{succ}
10	15	4.3e1	1.1e1	8.5e1	4.3e1	15	2.6e3	1.1e3	5.2e3	2.6e3
1	15	2.0e2	9.0e1	2.9e2	2.0e2	15	6.7e3	5.2e3	9.4e3	6.7e3
1e−1	15	3.8e2	2.8e2	5.2e2	3.8e2	15	9.0e3	7.0e3	1.1e4	9.0e3
1e−3	15	7.0e2	6.8e2	9.1e2	7.0e2	15	1.2e4	9.5e3	1.6e4	1.2e4
1e−5	15	1.2e3	1.1e3	1.5e3	1.2e3	15	1.6e4	1.3e4	1.8e4	1.6e4
1e−8	15	2.2e3	2.0e3	2.5e3	2.2e3	15	2.2e4	2.1e4	2.6e4	2.2e4

Δf	f_2 in 5-D, N=15, mFE=21317 #	ERT	10%	90%	RT_{succ}	f_2 in 20-D, N=15, mFE=115843 #	ERT	10%	90%	RT_{succ}
10	15	2.1e3	1.2e3	3.5e3	2.1e3	15	3.6e4	2.8e4	5.2e4	3.6e4
1	15	2.6e3	1.8e3	3.9e3	2.6e3	15	4.1e4	3.3e4	5.4e4	4.1e4
1e−1	15	3.1e3	2.2e3	4.2e3	3.1e3	15	4.3e4	3.5e4	5.8e4	4.3e4
1e−3	15	4.0e3	3.1e3	5.5e3	4.0e3	15	4.9e4	4.0e4	6.4e4	4.9e4
1e−5	15	4.8e3	3.7e3	7.1e3	4.8e3	15	5.5e4	4.5e4	7.0e4	5.5e4
1e−8	15	8.1e3	6.1e3	9.2e3	8.1e3	15	7.8e4	6.5e4	1.1e5	7.8e4

Δf	f_3 in 5-D, N=15, mFE=30780 #	ERT	10%	90%	RT_{succ}	f_3 in 20-D, N=15, mFE=123040 #	ERT	10%	90%	RT_{succ}
10	12	1.1e4	2.3e3	3.2e4	8.9e3	0	*15e+1*	*38e+0*	*39e+1*	3.0e4
1	7	4.9e4	3.1e3	1.1e5	1.3e4	.	.	.	.	.
1e−1	5	7.8e4	6.3e3	2.2e5	1.7e4	.	.	.	.	.
1e−3	5	8.0e4	1.1e4	2.2e5	1.6e4	.	.	.	.	.
1e−5	5	8.2e4	1.0e4	1.7e5	2.0e4	.	.	.	.	.
1e−8	4	1.0e5	1.3e4	2.6e5	2.0e4	.	.	.	.	.

Δf	f_4 in 5-D, N=15, mFE=30780 #	ERT	10%	90%	RT_{succ}	f_4 in 20-D, N=15, mFE=123040 #	ERT	10%	90%	RT_{succ}
10	11	1.9e4	1.5e3	3.4e4	7.4e3	0	*25e+1*	*13e+1*	*32e+1*	6.3e4
1	4	1.0e5	1.3e4	3.0e5	1.6e4	.	.	.	.	.
1e−1	2	2.3e5	2.0e4	5.5e5	2.7e4	.	.	.	.	.
1e−3	2	2.3e5	3.0e4	4.4e5	2.7e4	.	.	.	.	.
1e−5	2	2.3e5	3.0e4	5.2e5	2.8e4	.	.	.	.	.
1e−8	0	*59e-1*	*65e-7*	*25e+0*	2.5e4	.	.	.	.	.

Δf	f_5 in 5-D, N=15, mFE=2413 #	ERT	10%	90%	RT_{succ}	f_5 in 20-D, N=15, mFE=38803 #	ERT	10%	90%	RT_{succ}
10	15	1.5e2	3.0e1	2.1e2	1.5e2	15	1.3e4	8.5e3	2.1e4	1.3e4
1	15	3.6e2	1.7e2	7.8e2	3.6e2	15	1.9e4	1.2e4	3.0e4	1.9e4
1e−1	15	3.0e2	2.1e2	7.4e2	3.0e2	15	2.0e4	1.2e4	2.7e4	2.0e4
1e−3	15	3.0e2	2.1e2	6.2e2	3.0e2	15	2.0e4	1.0e4	2.7e4	2.0e4
1e−5	15	3.0e2	2.1e2	6.7e2	3.0e2	15	2.0e4	1.4e4	2.7e4	2.0e4
1e−8	15	3.9e2	2.1e2	7.4e2	3.9e2	15	2.0e4	1.2e4	2.7e4	2.0e4

Δf	f_6 in 5-D, N=15, mFE=23270 #	ERT	10%	90%	RT_{succ}	f_6 in 20-D, N=15, mFE=123040 #	ERT	10%	90%	RT_{succ}
10	15	4.0e2	2.1e2	5.8e2	4.0e2	15	3.3e4	2.6e4	4.5e4	3.3e4
1	15	1.0e3	4.7e2	2.0e3	1.0e3	15	4.7e4	3.7e4	5.9e4	4.7e4
1e−1	15	1.8e3	7.1e2	3.5e3	1.8e3	15	6.3e4	5.2e4	7.4e4	6.3e4
1e−3	15	4.0e3	2.6e3	5.5e3	4.0e3	14	1.0e5	7.2e4	1.2e5	9.5e4
1e−5	15	6.6e3	5.1e3	9.6e3	6.6e3	8	6.9e5	1.1e5	1.8e6	1.1e5
1e−8	15	1.3e4	1.1e4	1.9e4	1.3e4	0	*31e-6*	*56e-7*	*89e-6*	1.2e5

Δf	f_7 in 5-D, N=15, mFE=30780 #	ERT	10%	90%	RT_{succ}	f_7 in 20-D, N=15, mFE=123040 #	ERT	10%	90%	RT_{succ}
10	15	2.6e2	6.2e1	7.2e2	2.6e2	0	*59e+0*	*48e+0*	*13e+1*	3.7e4
1	7	3.6e4	4.2e2	9.3e4	1.1e3	.	.	.	.	.
1e−1	0	*16e-1*	*31e-2*	*23e-1*	2.6e3	.	.	.	.	.
1e−3	.	.	.	.	.	.	.	.	.	.
1e−5	.	.	.	.	.	.	.	.	.	.
1e−8	.	.	.	.	.	.	.	.	.	.

Δf	f_8 in 5-D, N=15, mFE=30780 #	ERT	10%	90%	RT_{succ}	f_8 in 20-D, N=15, mFE=123040 #	ERT	10%	90%	RT_{succ}
10	15	6.1e2	3.7e2	1.2e3	6.1e2	6	1.7e5	2.8e4	3.5e5	5.1e4
1	12	1.1e4	6.8e2	3.5e4	3.4e3	3	5.7e5	7.2e4	1.5e6	8.1e4
1e−1	12	1.3e4	6.7e2	3.6e4	5.4e3	1	1.8e6	3.5e5	4.5e6	1.1e5
1e−3	12	1.3e4	3.8e3	4.0e4	7.3e3	0	*98e-1*	*39e-2*	*12e+0*	1.2e5
1e−5	12	1.6e4	5.0e3	3.6e4	8.4e3	.	.	.	.	.
1e−8	12	1.8e4	7.5e3	4.3e4	1.1e4	.	.	.	.	.

Δf	f_9 in 5-D, N=15, mFE=30780 #	ERT	10%	90%	RT_{succ}	f_9 in 20-D, N=15, mFE=123040 #	ERT	10%	90%	RT_{succ}
10	15	3.9e2	1.8e2	6.8e2	3.9e2	3	5.8e5	9.3e4	1.3e6	9.6e4
1	12	1.0e4	5.0e2	3.4e4	3.5e3	1	1.8e6	1.8e5	3.5e6	5.0e4
1e−1	12	1.2e4	1.0e3	3.6e4	4.7e3	1	1.9e6	1.9e5	3.7e6	9.2e4
1e−3	12	1.3e4	4.4e3	3.6e4	7.0e3	0	*11e+0*	*69e-1*	*13e+0*	1.2e5
1e−5	12	1.4e4	5.5e3	4.6e4	7.0e3	.	.	.	.	.
1e−8	12	1.5e4	5.5e3	4.1e4	1.0e4	.	.	.	.	.

Δf	f_{10} in 5-D, N=15, mFE=30780 #	ERT	10%	90%	RT_{succ}	f_{10} in 20-D, N=15, mFE=123040 #	ERT	10%	90%	RT_{succ}
10	14	1.0e4	1.6e3	2.3e4	8.2e3	0	*65e+1*	*66e+0*	*59e+1*	1.2e5
1	11	2.7e4	3.9e3	5.6e4	1.6e4	.	.	.	.	.
1e−1	7	5.4e4	4.4e3	1.2e5	1.9e4	.	.	.	.	.
1e−3	0	*15e-3*	*27e-4*	*30e-3*	3.1e4	.	.	.	.	.
1e−5	.	.	.	.	.	.	.	.	.	.
1e−8	.	.	.	.	.	.	.	.	.	.

Δf	f_{11} in 5-D, N=15, mFE=30780 #	ERT	10%	90%	RT_{succ}	f_{11} in 20-D, N=15, mFE=123040 #	ERT	10%	90%	RT_{succ}
10	15	6.4e3	1.2e3	1.5e4	6.4e3	12	1.1e5	5.1e4	2.0e5	8.0e4
1	11	2.6e4	3.7e3	7.5e4	1.3e4	1	1.8e6	2.2e5	4.0e6	8.2e4
1e−1	8	3.5e4	2.0e4	1.9e5	2.4e4	0	*39e-1*	*16e-1*	*36e+0*	1.2e5
1e−3	0	*61e-3*	*18e-3*	*17e-2*	3.1e4	.	.	.	.	.
1e−5	.	.	.	.	.	.	.	.	.	.
1e−8	.	.	.	.	.	.	.	.	.	.

Δf	f_{12} in 5-D, N=15, mFE=30780 #	ERT	10%	90%	RT_{succ}	f_{12} in 20-D, N=15, mFE=123040 #	ERT	10%	90%	RT_{succ}
10	15	4.8e3	1.0e3	1.0e4	4.8e3	12	4.7e4	1.4e4	1.4e5	1.8e4
1	13	1.4e4	1.2e3	3.2e4	9.3e3	4	3.8e5	1.7e4	7.8e5	1.6e4
1e−1	11	2.8e4	3.9e3	4.7e4	1.8e4	1	1.7e6	5.4e4	3.4e6	2.2e4
1e−3	7	5.3e4	1.3e4	1.1e5	1.8e4	0	*29e-1*	*35e-2*	*40e+0*	1.2e5
1e−5	5	7.8e4	1.4e4	1.7e5	1.8e4	.	.	.	.	.
1e−8	3	1.5e5	2.0e4	3.6e5	2.4e4	.	.	.	.	.

Δf	f_{13} in 5-D, N=15, mFE=30780 #	ERT	10%	90%	RT_{succ}	f_{13} in 20-D, N=15, mFE=123040 #	ERT	10%	90%	RT_{succ}
10	12	8.0e3	4.5e2	2.2e3	1.2e3	11	6.2e4	1.3e4	1.4e5	1.7e4
1	7	3.8e4	1.0e3	9.7e4	2.8e3	4	2.7e5	2.3e4	7.7e5	2.8e4
1e−1	3	1.3e5	3.3e3	3.2e5	4.9e3	1	1.8e6	1.3e5	3.4e6	2.9e4
1e−3	0	*31e-1*	*19e-3*	*69e+0*	2.9e4	0	*36e-1*	*30e-2*	*26e+0*	1.2e5
1e−5	.	.	.	.	.	.	.	.	.	.
1e−8	.	.	.	.	.	.	.	.	.	.

Δf	f_{14} in 5-D, N=15, mFE=30780 #	ERT	10%	90%	RT_{succ}	f_{14} in 20-D, N=15, mFE=123040 #	ERT	10%	90%	RT_{succ}
10	15	2.3e1	3.0e0	6.0e1	2.3e1	15	1.1e3	3.7e2	4.1e3	1.1e3
1	15	2.1e2	1.1e2	2.9e2	2.1e2	15	7.9e3	6.0e3	1.1e4	7.9e3
1e−1	15	4.0e2	2.8e2	4.9e2	4.0e2	15	1.1e4	9.1e3	1.4e4	1.1e4
1e−3	15	1.7e3	1.2e3	3.3e3	1.7e3	15	3.3e4	3.1e4	4.1e4	3.3e4
1e−5	14	1.8e4	4.8e3	2.5e4	1.2e4	0	*69e-6*	*71e-6*	*90e-6*	1.2e5
1e−8	1	4.8e5	6.1e4	1.1e6	3.1e4	.	.	.	.	.

Δf	f_{15} in 5-D, N=15, mFE=30780 #	ERT	10%	90%	RT_{succ}	f_{15} in 20-D, N=15, mFE=123040 #	ERT	10%	90%	RT_{succ}
10	10	1.6e4	6.1e2	6.3e4	7.7e3	0	*25e+1*	*32e+1*	*38e+1*	8.0e4
1	1	4.3e5	3.3e4	9.8e5	2.0e3	.	.	.	.	.
1e−1	0	*79e-1*	*90e-1*	*00e+0*	7.0e3	.	.	.	.	.
1e−3	.	.	.	.	.	.	.	.	.	.
1e−5	.	.	.	.	.	.	.	.	.	.
1e−8	.	.	.	.	.	.	.	.	.	.

Δf	f_{16} in 5-D, N=15, mFE=30780 #	ERT	10%	90%	RT_{succ}	f_{16} in 20-D, N=15, mFE=123040 #	ERT	10%	90%	RT_{succ}
10	15	1.3e2	1.2e1	2.7e2	1.3e2	4	3.4e5	3.1e3	8.0e5	3.1e4
1	9	3.2e4	7.1e2	6.3e4	1.3e4	0	*14e+0*	*68e-1*	*10e+0*	1.2e5
1e−1	1	4.3e5	3.3e4	1.1e6	2.0e4	.	.	.	.	.
1e−3	0	*76e-2*	*12e-2*	*26e-1*	3.0e4	.	.	.	.	.
1e−5	.	.	.	.	.	.	.	.	.	.
1e−8	.	.	.	.	.	.	.	.	.	.

Δf	f_{17} in 5-D, N=15, mFE=30780 #	ERT	10%	90%	RT_{succ}	f_{17} in 20-D, N=15, mFE=123040 #	ERT	10%	90%	RT_{succ}
10	15	2.7e1	6.0e0	6.8e1	2.7e1	15	4.5e2	1.0e2	6.8e2	4.5e2
1	14	3.7e3	1.2e2	9.8e2	4.0e2	0	*55e-1*	*38e-1*	*75e-1*	1.2e5
1e−1	7	4.0e4	6.1e2	9.3e4	5.1e3	.	.	.	.	.
1e−3	1	4.4e5	7.3e4	1.0e6	1.2e4	.	.	.	.	.
1e−5	0	*36e-3*	*40e-4*	*73e-3*	2.9e4	.	.	.	.	.
1e−8	.	.	.	.	.	.	.	.	.	.

Δf	f_{18} in 5-D, N=15, mFE=30780 #	ERT	10%	90%	RT_{succ}	f_{18} in 20-D, N=15, mFE=123040 #	ERT	10%	90%	RT_{succ}
10	15	2.5e2	9.5e1	2.9e2	2.5e2	1	1.7e6	1.3e5	3.8e6	3.4e4
1	10	1.8e4	4.2e3	3.2e4	9.0e3	0	*15e+0*	*11e+0*	*66e+0*	1.2e5
1e−1	1	4.8e5	3.5e4	1.1e6	2.4e4	.	.	.	.	.
1e−3	0	*30e-2*	*11e-2*	*26e-1*	2.0e4	.	.	.	.	.
1e−5	.	.	.	.	.	.	.	.	.	.
1e−8	.	.	.	.	.	.	.	.	.	.

Δf	f_{19} in 5-D, N=15, mFE=30780 #	ERT	10%	90%	RT_{succ}	f_{19} in 20-D, N=15, mFE=123040 #	ERT	10%	90%	RT_{succ}
10	15	3.7e1	1.6e1	5.3e1	3.7e1	15	1.8e2	6.5e1	2.5e2	1.8e2
1	15	2.7e3	3.9e2	6.3e3	2.7e3	4	4.0e5	6.3e4	9.7e5	5.7e4
1e−1	2	2.1e5	1.6e4	4.5e5	1.0e4	0	*12e-1*	*79e-2*	*24e-1*	1.2e5
1e−3	0	*36e-3*	*79e-3*	*85e-3*	3.1e4	.	.	.	.	.
1e−5	.	.	.	.	.	.	.	.	.	.
1e−8	.	.	.	.	.	.	.	.	.	.

Δf	f_{20} in 5-D, N=15, mFE=30780 #	ERT	10%	90%	RT_{succ}	f_{20} in 20-D, N=15, mFE=123040 #	ERT	10%	90%	RT_{succ}
10	15	6.8e1	4.8e1	1.7e2	6.8e1	15	9.7e2	3.7e2	1.3e3	9.7e2
1	5	6.5e4	6.3e2	1.5e5	6.6e3	0	*16e-1*	*17e-1*	*30e-1*	1.1e5
1e−1	0	*15e-1*	*49e-2*	*18e-1*	1.0e4	.	.	.	.	.
1e−3	.	.	.	.	.	.	.	.	.	.
1e−5	.	.	.	.	.	.	.	.	.	.
1e−8	.	.	.	.	.	.	.	.	.	.

Δf	f_{21} in 5-D, N=15, mFE=30780 #	ERT	10%	90%	RT_{succ}	f_{21} in 20-D, N=15, mFE=123040 #	ERT	10%	90%	RT_{succ}
10	15	1.3e2	5.3e1	4.3e2	1.3e2	12	3.8e4	8.3e2	1.3e5	5.1e3
1	4	3.3e4	1.2e2	2.2e5	2.1e2	2	5.1e5	7.9e3	1.7e6	6.3e3
1e−1	4	6.3e4	4.3e2	1.6e5	4.4e2	1	1.7e6	1.3e5	3.8e6	8.4e3
1e−3	4	8.3e4	6.3e2	2.6e5	7.8e2	1	1.7e6	1.4e5	4.0e6	1.3e4
1e−5	4	8.6e4	1.3e3	1.9e5	1.3e3	1	1.7e6	1.4e5	3.6e6	1.6e4
1e−8	4	8.6e4	1.7e3	2.8e5	1.6e3	1	1.8e6	1.5e5	4.5e6	2.8e4

Δf	f_{22} in 5-D, N=15, mFE=30780 #	ERT	10%	90%	RT_{succ}	f_{22} in 20-D, N=15, mFE=123040 #	ERT	10%	90%	RT_{succ}
10	13	4.9e3	7.7e1	3.1e4	2.0e3	8	1.9e5	4.0e3	5.6e5	6.3e3
1	1	4.3e5	3.1e4	9.4e5	4.5e2	2	8.1e5	1.3e5	1.7e6	7.3e3
1e−1	0	*91e-1*	*35e-1*	*15e+0*	4.0e3	0	*15e+0*	*66e-2*	*39e+0*	4.0e4
1e−3	.	.	.	.	.	.	.	.	.	.
1e−5	.	.	.	.	.	.	.	.	.	.
1e−8	.	.	.	.	.	.	.	.	.	.

Δf	f_{23} in 5-D, N=15, mFE=30780 #	ERT	10%	90%	RT_{succ}	f_{23} in 20-D, N=15, mFE=123040 #	ERT	10%	90%	RT_{succ}
10	15	3.3e0	1.0e0	3.8e1	5.9e0	15	9.1e0	4.0e0	2.9e1	9.1e0
1	15	2.4e3	1.3e3	3.5e3	2.4e3	5	8.0e5	1.5e4	1.5e6	1.2e4
1e−1	2	1.4e5	9.3e3	3.1e5	1.3e4	0	*12e-1*	*79e-2*	*15e-1*	1.2e5
1e−3	0	*26e-2*	*36e-3*	*56e-2*	5.0e4	.	.	.	.	.
1e−5	.	.	.	.	.	.	.	.	.	.
1e−8	.	.	.	.	.	.	.	.	.	.

Δf	f_{24} in 5-D, N=15, mFE=30780 #	ERT	10%	90%	RT_{succ}	f_{24} in 20-D, N=15, mFE=123040 #	ERT	10%	90%	RT_{succ}
10	7	3.3e4	1.4e3	9.5e4	2.8e3	0	*18e+1*	*73e+0*	*18e+1*	1.7e5
1	0	*11e+0*	*79e-1*	*15e+0*	2.2e4	.	.	.	.	.
1e−1	.	.	.	.	.	.	.	.	.	.
1e−3	.	.	.	.	.	.	.	.	.	.
1e−5	.	.	.	.	.	.	.	.	.	.
1e−8	.	.	.	.	.	.	.	.	.	.

Table 1: Shown are, for a given target difference to the optimal function value Δf: the number of successful trials (#); the expected running time to surpass $f_{opt} + \Delta f$ (ERT, see Figure 1); the 10%-tile and 90%-tile of the bootstrap distribution of ERT; the average number of function evaluations in successful trials or, if none was successful, as last entry the median number of function evaluations to reach the best function value (RT_{succ}). If $f_{opt} + \Delta f$ was never reached, figures in *italics* denote the best achieved Δf-value of the median trial and the 10% and 90%-tile trial. Furthermore, N denotes the number of trials, and mFE denotes the maximum of number of function evaluations executed in one trial. See Figure 1 for the names of functions.

The simulations for 2; 5; 10; 20 and 40 D were done with the C-code and took 2 hours and a half. No parameter tuning was done and the crafting effort CrE [3] is computed to zero.

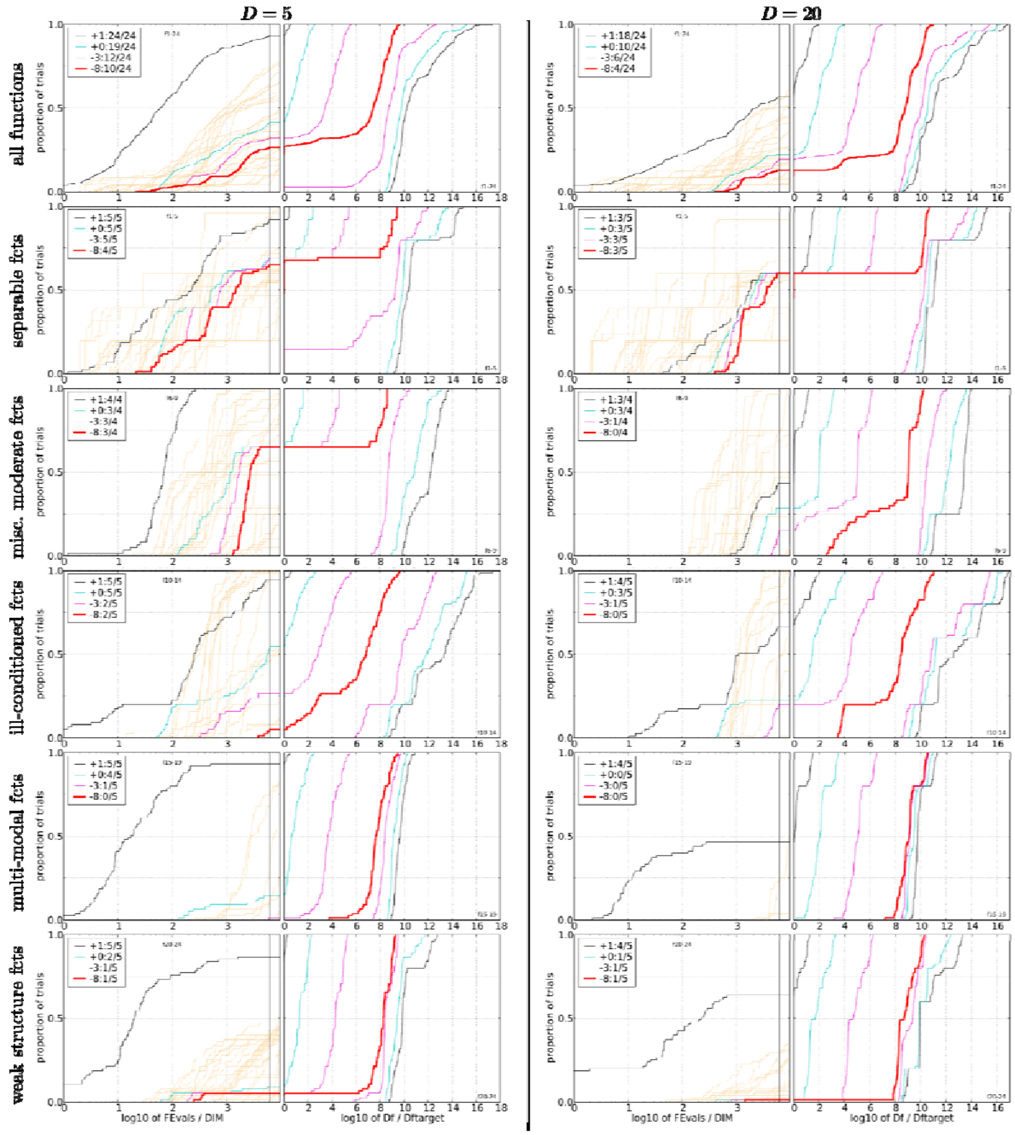

Figure 2: Empirical cumulative distribution functions (ECDFs), plotting the fraction of trials versus running time (left subplots) or versus Δf (right subplots). The thick red line represents the best achieved results. Left subplots: ECDF of the running time (number of function evaluations), divided by search space dimension D, to fall below $f_{opt} + \Delta f$ with $\Delta f = 10^k$, where k is the first value in the legend. Right subplots: ECDF of the best achieved Δf divided by 10^k (upper left lines in continuation of the left subplot), and best achieved Δf divided by 10^{-8} for running times of $D, 10\,D, 100\,D\ldots$ function evaluations (from right to left cycling black-cyan-magenta). The legends indicate the number of functions that were solved in at least one trial. FEvals denotes number of function evaluations, D and DIM denote search space dimension, and Δf and Df denote the difference to the optimal function value. Light brown lines in the background show ECDFs for target value 10^{-8} of all algorithms benchmarked during BBOB-2009.

Results

Results from experiments according to [2] on the benchmarks functions given in [1, 3] are presented in Figures 1, 2 and 3 and in Tables 1 and 2. The algorithm solves some of the moderate functions f1, f2, f5, f6, f14 and f21. Else, f8, f9, f11, f12, f13 are partially solved for dimensions 20.

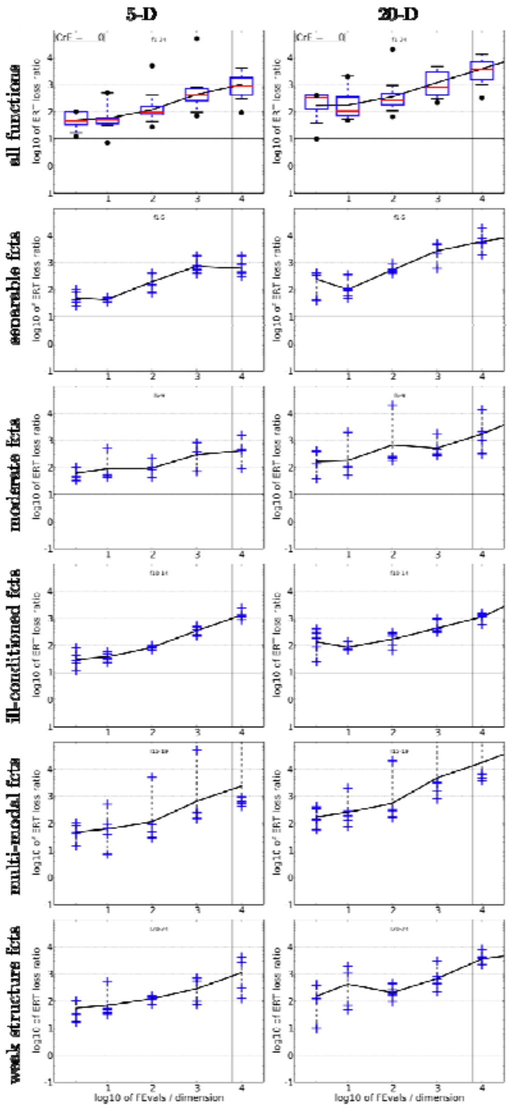

Figure 3: ERT loss ratio versus given budget FEvals. The target value f_t for ERT (see Figure 1) is the smallest (best) recorded function value such that ERT(f_t) ≤ FEvals for the presented algorithm. Shown is FEvals divided by the respective best ERT(f_t) from BBOB-2009 for functions f_1–f_{24} in 5-D and 20-D. Each ERT is multiplied by exp(CrE) correcting for the parameter crafting effort. Line: geometric mean. Box-Whisker error bar: 25-75%-ile with median (box), 10-90%-ile (caps), and minimum and maximum ERT loss ratio (points). The vertical line gives the maximal number of function evaluations in this function subset.

Table 2: ERT loss ratio (see Figure 3) compared to the respective best result from BBOB-2009 for budgets given in the first column. The last row RL_{US}/D gives the number of function evaluations in unsuccessful runs divided by dimension. Shown are the smallest, 10%-ile, 25%-ile, 50%-ile, 75%-ile and 90%-ile value (smaller values are better).

	f_1–f_{24} in 5-D, maxFE/D=6158					
#FEs/D	best	10%	25%	med	75%	90%
2	1.2	1.7	3.1	4.4	10	10
10	0.72	3.2	3.5	4.9	5.8	50
100	2.8	4.3	7.5	9.5	15	40
1e3	7.1	9.4	24	38	70	91
1e4	9.1	29	43	87	1.7e2	4.0e2
RL_{US}/D	6e3	6e3	6e3	6e3	6e3	6e3

	f_1–f_{24} in 20-D, maxFE/D=6152					
#FEs/D	best	10%	25%	med	75%	90%
2	1.0	3.6	11	31	40	40
10	4.7	5.1	7.2	10	32	2.0e2
100	6.4	11	18	26	45	2.8e2
1e3	22	29	41	80	3.0e2	4.7e2
1e4	32	94	1.4e2	3.7e2	6.7e2	1.4e3
1e5	99	2.0e2	3.5e2	6.2e2	3.7e3	6.2e3
RL_{US}/D	6e3	6e3	6e3	6e3	6e3	6e3

Conclusion

We have presented the results of the Particle Swarm Optimization algorithm with a Differential Evolution term, that does use information gathered during search for guiding its next stops following a social behavior not a genetic one. Those results provide a baseline comparison that every adaptive algorithm should outperform. Results have been obtained using the Black Box Optimization Benchmark 2010, which provides useful tools to analyze data in a graphical way.

Bibliography

[1] S. H. Brooks. A discussion of random methods for seeking maxima. Operations Research, 6:244– 251, 1958.

[2] S. Finck, N. Hansen, R. Ros, and A. Auger. Real-parameter black-box optimization benchmarking 2009: Presentation of the noiseless functions. Technical Report 2009/20, Research Center PPE, 2009.

[3] N. Hansen, A. Auger, S. Finck, and R. Ros. Real-parameter black-box optimization benchmarking 2009: Experimental setup. Technical Report RR-6828, INRIA, 2009.

[4] N. Hansen, S. Finck, R. Ros, and A. Auger. Real-parameter black-box optimization benchmarking 2009: Noiseless functions definitions. Technical Report RR-6829, INRIA, 2009.

[5] M. J. D. Powell. The NEWUOA software for unconstrained optimization without derivatives. Large Scale Nonlinear Optimization, pages 255–297, 2006.

[6] J. Nelder and R. Mead. The downhill simplex method. Computer Journal, 7:308–313, 1965.

[7] T Jayabarathi, Sandeep Chalasani, Zameer Ahmed Shaik, Nishchal Deep Kodali; "Hybrid Differential Evolution and Particle Swarm Optimization Based Solutions to Short Term Hydro Thermal Scheduling", WSEAS Transactions on Power Systems Issue 11, Volume 2, pp. , ISSN: 1790-5060, 2007.

[8] Piao Haiguo, Wang Zhixin, Zhang Huaqiang, "Cooperative-PSO-Based PID Neural Network Integral Control Strategy and Simulation Research with Asynchronous Motor Controller Design", WSEAS Transactions on Circuits and Systems Volume 8, pp. 136-141, ISSN: 1109-2734, 2009.

[9] Lijia Ren, Xiuchen Jiang, Gehao Sheng, Wu B;"A New Study in Maintenance for Transmission Lines", WSEAS Transactions on Circuits and Systems Volume 7, pp. 53-37, ISSN: 1109-2734, 2008.

[10] Kenneth Price. Differential evolution vs. the functions of the second ICEO. In Proceedings of the IEEE International Congress on Evolutionary Computation, pages 153–157, 1997.

[11] Kenneth Price, Rainer M. Storn, and Jouni A. Lampinen. Differential Evolution: A Practical Approach to Global Optimization (Natural Computing Series). Springer- Verlag New York, Inc., 2005. ISBN 3540209506. URL http://portal.acm.org/citation.cfm?id=1121631.

[12] K.V. Price. Differential evolution: a fast and simple numerical optimizer. In Fuzzy Information Processing Society, 1996. NAFIPS. 1996 Biennial Conference of the North American, pages 524–527, 1996. doi: {10.1109/NAFIPS.1996.534790}.

Authors' Information

Nuria Gómez Blas *– Associate professor U.P.M Crtra Valencia km 7, Madrid-28031, Spain; e-mail: ngomez@eui.upm.es*

Research: DNA computing, Membrane computing, Education on Applied Mathematics and Informatics

Luis F. de Mingo *– Associate professor U.P.M Crtra Valencia km 7, Madrid-28031, Spain; e-mail: lfmingo@eui.upm.es*

Research:,Artificial Intelligence, Social Intelligence, Education on Applied Mathematics and Informatics