Automatic Tuning of Standard PSO Versions

Roberto Ugolotti
Dept. of Information Engineering
University of Parma, Italy
rob_ugo@ce.unipr.it

Stefano Cagnoni
Dept. of Information Engineering
University of Parma, Italy
cagnoni@ce.unipr.it

ABSTRACT

We used a meta-optimization environment to compare two reference versions of the Particle Swarm Optimization (PSO) algorithm, namely Standard PSO 2006 and Standard PSO 2011, on the CEC 2013 benchmark. We first compared the performances of the two standard PSO versions using the parameter sets suggested in the literature. Then, we automatically tuned the parameter values of both algorithms, to allow the two versions to perform at their best.

Categories and Subject Descriptors

I.2.8 [**Artificial Intelligence**]: Problem Solving, Control Methods, and Search—*Heuristic methods*

Keywords

Particle Swarm Optimization; Meta-Optimization

1. INTRODUCTION

We have performed an unbiased comparison between two standard Particle Swarm Optimization (PSO) versions on a well known function set. We used automatic tuning to optimize the parameter settings for both algorithms, in order to compare their best possible performance.

Three Standard PSO (SPSO) versions have been presented so far: SPSO2006, SPSO2007 and SPSO2011. The first two differ only very slightly, so we decided to compare just SPSO2006 and SPSO2011. Details on standard PSO versions can be found in [1].

2. META-OPTIMIZATION

Meta-Optimization (see Figure 1) is one of the most commonly used ways to automatically select the best parameter values for a Meta-Heuristic (MH).

The lower part of the figure represents a general optimization problem: a MH (Lower-Level MH, LL-MH) optimizes a function. The Tuner MH (above in the figure) works very similarly but optimizes the parameters of LL-MH rather than directly optimizing the solution to the problem at hand. This means that Tuner MH generates possible LL-MH configurations. For each set of parameters, an entire optimization process using LL-MH is repeated T times on the function(s) considered. An aggregated measure of the results (e.g., the average final fitness) represents the fitness value for the Tuner MH population.

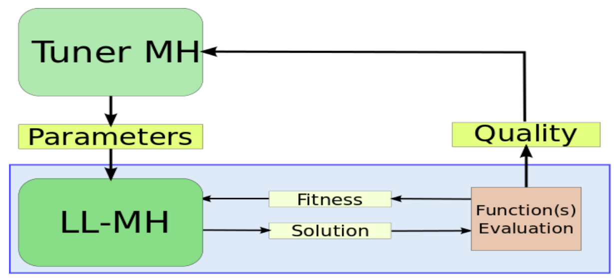

Figure 1: Scheme of Meta-Optimization.

SEPaT (Simple Evolutionary Parameter Tuning), an implementation of the Meta-Optimization paradigm first introduced in [3], has been used to tune the PSO versions, using Differential Evolution (DE) as Tuner MH.

We optimized numerical parameters (namely population size, inertia factor w, c_1, c_2, and neighborhood size K) as well as nominal parameters (namely *i.* topology, with three options: *adaptive random (ART)*, *probabilistic random (PRT)* and *ring*; and *ii.* particles' update rule, with SPSO2006 or SPSO2011 as options.

3. EXPERIMENTAL RESULTS

We first present the results of a direct comparison between SPSO2006 and SPSO2011. The parameter values used by the basic versions are the ones suggested in [1] and in [4] and reported in Table 1.

The third and fourth column of Table 2 show the medians obtained by SPSO2006 and SPSO2011 over the 28 functions that compose the CEC 2013 benchmark [2] in $D = 10$ dimensions. For each PSO version and each function, 51 independent runs were performed with a termination criterion of $1000 \cdot D$ fitness evaluations.

SPSO2011 performs statistically better than SPSO2006 on 13 functions, a tie occurs in 9 cases, while SPSO2006 is better in the other 6 cases.

SPSO2006 performs better on separable functions ($f5$, $f11$, $f22$) because it updates each dimension independently

GECCO'15 Companion, July 11–15, 2015, Madrid, Spain.

ACM ISBN 978-1-4503-3488-4/15/07..
DOI: http://dx.doi.org/10.1145/2739482.2764706

of the others and is therefore more suitable for solving this kind of problems. The same happens with non-rotated functions ($f14$, $f17$). In the other situations (multimodal, rotated, non-separable functions) SPSO2011 obtains better results.

In a subsequent step, we meta-optimized the parameters. We divided the CEC 2013 function set into a training set of 7 functions and a test set with the remaining 21. To avoid favoring any of the two versions a priori, the training set included three functions on which the two SPSO versions yield equal results ($f1$, $f3$, $f8$), two on which SPSO2006 performs better than SPSO2011 ($f11$, $f21$) and two for which SPSO2011 was better ($f10$, $f23$).

The parameters of the DE-based tuner were set as: 64 individuals, 40 generations, crossover rate 0.9, scale factor 0.5, target-to-best mutation, exponential crossover.

Table 1: PSO parameter values used in the standard versions and meta-optimized ones.

Parameter	Standard	TPSO
Population Size	40	246
Inertia Factor (w)	$\frac{1}{2 \cdot ln(2)}$	0.687378
c_1	$0.5 + ln(2)$	0.246448
c_2	$0.5 + ln(2)$	0.701503
K	3	6
Topology	*ART*	*ART*
Update	*SPSO20XX*	*SPSO2006*

The PSO parameters obtained by SEPaT are presented in Table 1. The most striking difference with respect to the standard parameter values regards population size, which confirmed the suggestion [4] that a small population may cause PSO to perform badly. At the same time, however, K does not increase as much, suggesting that the best strategy for PSO is to use many particles with little communication between one another and that strategies like niching and sub-swarming may therefore be beneficial.

Interestingly, although SPSO2011 performed better using the standard parameters, SEPaT chose to update velocity and position of the particles according to SPSO2006. This choice suggests that a correct parameter setting may overturn the conclusions derived from the comparison of two algorithms based on standard parameter values.

Regarding the other parameters, the inertia factor w is close to the values commonly used in the literature, while c_1 and c_2 tend to be smaller than the ones usually suggested. Moreover, c_2 is larger than c_1, favoring communication between particles with respect to exploitation of their own findings.

Table 2 shows the results obtained on the 28 functions of the CEC 2013 benchmark. Columns $3-5$ of the table show the medians of the final fitness for the three PSO versions. The last column shows the version which performs better. It can be observed that the tuned version performs better not only on the function on which it was trained, but also on most functions which do not belong to the training set.

4. CONCLUSIONS AND FUTURE WORK

In this work, two standard versions of Particle Swarm Optimization, SPSO2006 and SPSO2011 have been investigated. Using standard parameter settings, SPSO2011 was shown to perform slightly better.

Table 2: Medians of the results obtained by the standard and tuned PSO versions in 10 dimensions. The last column indicates the best-performing version, according to the Wilcoxon signed- rank test ($p < 0.01$). T denotes the functions belonging to the training set.

Fun	Goal	SPSO2006	SPSO2011	TPSO	Best
f1	-1.400e+03	-1.400e+03	-1.400e+03	-1.400e+03	T
f2	-1.300e+03	1.006e+05	1.457e+04	1.381e+04	2011, TPSO
f3	-1.200e+03	-9.851e+02	1.108e+03	-1.198e+03	T
f4	-1.100e+03	6.998e+02	1.896e+02	1.350e+03	2011
f5	-1.000e+03	-1.000e+03	-1.000e+03	-1.000e+03	2006
f6	-9.000e+02	-8.902e+02	-8.902e+02	-8.902e+02	2006, TPSO
f7	-8.000e+02	-7.999e+02	-7.996e+02	-8.000e+02	TPSO
f8	-7.000e+02	-6.797e+02	-6.797e+02	-6.797e+02	T
f9	-6.000e+02	-5.978e+02	-5.979e+02	-5.987e+02	TPSO
f10	-5.000e+02	-4.999e+02	-4.999e+02	-5.000e+02	T
f11	-4.000e+02	-3.970e+02	-3.952e+02	-3.980e+02	T
f12	-3.000e+02	-2.909e+02	-2.956e+02	-2.970e+02	TPSO
f13	-2.000e+02	-1.908e+02	-1.945e+02	-1.961e+02	TPSO
f14	-1.000e+02	6.047e+01	5.378e+02	5.815e+01	2006, TPSO
f15	1.000e+02	9.261e+02	6.200e+02	4.160e+02	TPSO
f16	2.000e+02	2.008e+02	2.007e+02	2.007e+02	2011, TPSO
f17	3.000e+02	3.169e+02	3.173e+02	3.129e+02	TPSO
f18	4.000e+02	4.244e+02	4.183e+02	4.167e+02	TPSO
f19	5.000e+02	5.008e+02	5.008e+02	5.006e+02	TPSO
f20	6.000e+02	6.025e+02	6.025e+02	6.021e+02	-
f21	7.000e+02	1.100e+03	1.100e+03	1.100e+03	T
f22	8.000e+02	9.684e+02	1.268e+03	8.661e+02	TPSO
f23	9.000e+02	1.643e+03	1.245e+03	9.865e+02	T
f24	1.000e+03	1.210e+03	1.201e+03	1.200e+03	2011, TPSO
f25	1.100e+03	1.307e+03	1.301e+03	1.300e+03	TPSO
f26	1.200e+03	1.400e+03	1.308e+03	1.304e+03	2011, TPSO
f27	1.300e+03	1.700e+03	1.601e+03	1.600e+03	TPSO
f28	1.400e+03	1.700e+03	1.700e+03	1.700e+03	-

Then, a meta-optimization environment has been used to tune their parameters which was able to improve PSO performance not only on the training functions but also on the other functions in the CEC2013 benchmark.

The main conclusion is that PSO needs a larger population size than is usually set in the standard versions. Regarding topology, in the tuned version the larger population is sparsely connected, which suggests that approaches like sub-swarms and niching could improve PSO's performance.

Since our results are better than the ones obtained by standard settings of standard versions, we suggest that the parameters reported in Table 1 be used when comparing a novel version to the standard algorithm, especially when testing on the same standard benchmark we used.

5. REFERENCES

[1] M. Clerc. Standard particle swarm optimisation from 2006 to 2011. Technical report, Particle Swarm Central, 2011.

[2] J. Liang, B. Qu, and P. Suganthan. Problem definitions and evaluation criteria for the CEC 2013 special session on real-parameter optimization. Technical Report 201212, Zhengzhou University, 2013.

[3] R. Ugolotti, Y. S. G. Nashed, P. Mesejo, and S. Cagnoni. Algorithm configuration using GPU-based metaheuristics. In *Genetic and Evolutionary Computation Conference (GECCO) Companion Proceedings*, pages 221–222. ACM, 2013.

[4] M. Zambrano-Bigiarini, M. Clerc, and R. Rojas. Standard particle swarm optimisation 2011 at CEC-2013: A baseline for future pso improvements. In *IEEE Congress on Evolutionary Computation*, CEC'13, pages 2337–2344, June 2013.