

Empirical Assessment of Human Learning Principles Inspired PSO Algorithms on Continuous Black-Box Optimization Testbed

M.R. Tanweer^(✉), Abdullah Al-Dujaili, and S. Suresh

School of Computer Engineering, Nanyang Technological University,
Singapore, Singapore

{muhammad170,aldujail001}@e.ntu.edu.sg, ssundaram@ntu.edu.sg

Abstract. This paper benchmarks the performance of one of the recent research directions in the performance improvement of particle swarm optimization algorithm; human learning principles inspired PSO variants. This article discusses and provides performance comparison of nine different PSO variants. The Comparing Continuous Optimizers (COCO) methodology has been adopted in comparing these variants on the noiseless BBOB testbed, providing useful insight regarding their relative efficiency and effectiveness. This study provides the research community a comprehensive account of suitability of a PSO variant in solving selective class of problems under different budget settings. Further, certain rectifications/extensions have also been suggested for the selected PSO variants for possible performance enhancement. Overall, it has been observed that SL-PSO and MePSO are most suited for expensive and moderate budget settings respectively. Further, iSRPSO and TPLPSO have provided better solutions under cheap budget settings where iSRPSO has shown robust behaviour (better solutions over dimensions). We hope this paper would mark a milestone in assessing the human learning principles inspired PSO algorithms and used as a baseline for performance comparison.

Keywords: PSO · Human learning principles inspired PSO variants · COCO methodology · Black-box optimization

1 Introduction

Particle Swarm Optimization (PSO) [10] is a population-based search optimization algorithm, inspired from the social behavior a swarm of bird. It has been widely used for solving numerous optimization problem [4, 11, 22] and has successfully provided solutions to the complex real-world optimization problems [15, 19, 21]. In the past two decades, its simplicity and computational efficiency has attracted the researchers. As a result, several research directions have been studied including parameter tuning, neighbourhood topology, learning strategies etc. [5, 13, 27]. The current state-of-the-art research in PSO includes a diverse

collection of modified PSO variants, with one variant performing better than other on a class of optimization problems. It has been shown in human learning psychology that human beings are better planners and possess intelligent information processing skills. This helps one to perform better self-regulation of the cognitive strategies and hence enhance the decision making abilities [14]. Therefore, algorithms developed using human learning principles have shown promising characteristics [17, 20]. Inspired from these findings, researchers have tried to design such PSO variants that can provide better solutions on various classes of problems by introducing human-like learning principles [18, 22, 24, 25]. This research direction has provided robust and efficient PSO variants capable of solving more classes of optimization problems. Recently, researchers have developed several human learning principles inspired PSO variants which has significantly enhanced the algorithms' performance. Therefore, it is required to benchmark the performance of these algorithms to come up with a baseline for performance comparison in assessing the human learning principles inspired PSO algorithms.

This paper aims towards comparing eight different human learning principles inspired PSO variants carefully selected to reflect different self and social principles applied to the PSO algorithm against the standard PSO algorithm. Through experimental evaluation, this paper marks the differences among them and investigates the suitability of the algorithms for solving optimization problems with different characteristics for different computational requirements. Furthermore, it draws several concluding points that can help the future development in the PSO research. To achieve the paper's goals, the experimental evaluation must be able to discover and distinguish the good features of the variants over others, and show their differences over different stages of the search for various goals and situations. In this paper, the Comparing Continuous Optimizer (COCO) methodology [7] has been adopted as it meets the requirements. It comes with a testbed of 24 scalable noiseless functions [8] addressing such real-world difficulties as ill-conditioning, multi-modality, and dimensionality. The selected PSO variants have diverse behaviour in terms of solution accuracy, convergence rate and robustness. Some require more time to converge [9, 23] whereas others are computationally efficient [12, 21, 24]. Therefore, the performance has been assessed over different settings of function evaluations to investigate the capability of the algorithms in solving computationally expensive, moderate and cheap budget optimization problems. This evaluation provide the researchers an insight of selection of an appropriate algorithm depending on what is known about the optimization problem in terms of evaluation budget, dimensionality and function structure.

The rest of the paper is organized as follows: Sect. 2 provides a brief description of the selected human learning principles inspired PSO variants. In Sect. 3, the numerical assessment of the algorithms is presented, including experimental setup, procedure for evaluating the algorithms' performance and discussion of the results. Section 4 summarizes the main conclusions from this study, and suggests possible extensions for further performance improvements.

2 Selected Human Learning Principles Inspired PSO Algorithms

Human learning principles inspired PSO variants broadly fall into three different categories: Self-learning principles based PSO, Social-learning principles based PSO and Combined self and social learning principles based PSO. In this paper, the following PSO variants carefully selected to represent different categories have been chosen for performance evaluation.

Self Learning Principles based PSO: The Self Regulating PSO (SRPSO) algorithm [21] and the Globally Adaptive Inertia Weight PSO (GAIWPSO) algorithm [1] are the two selected human self-learning inspired PSO variants. In SRPSO, human self-cognition in the form of regulation and perception has been incorporated in the basic PSO algorithm for enhanced exploration and intelligent exploitation of the search space. In GAIWPSO, a greedy approach for inertia weight has been adopted where a human-like adaptation strategy (increasing and decreasing according to fitness) has been incorporated which has successfully accelerated the convergence towards better solutions.

Social Learning Principles based PSO: The two selected algorithms inspired from human social learning principles are the Social Learning PSO (SL-PSO) algorithm [2] and the Competitive Swarm Optimizer (CSO) [3]. The SL-PSO algorithm introduced a new concept where a particle is allowed to learn from any particle that has a better performance instead of just the best particle whereas in CSO, a pairwise competition mechanism is introduced for the particles where all the losing particles learn from the winner particles.

Combined Self and Social Learning Principles based PSO: This category includes those PSO variants where the particles have a self-learning mechanism together with a guidance mechanism for better performance. These PSO variants have shown faster convergence characteristics with efficiency and robustness over a wide range of optimization problems. The four selected algorithms from this category are: the Mentoring based PSO (MePSO) algorithm [24], the Teaching and Peer-Learning PSO (TPLPSO) algorithm [12], the Example-based Learning PSO (ELPSO) algorithm [9] and the improved SRPSO (iSRPSO) algorithm [23]. In MePSO, the particles are divided into mentors, mentees and independent learners where the mentees are socially guided by the mentors and other particles perform independent search. Two phases of learning, the teaching and the peer learning phase are introduced in TPLPSO where the under performing particles from the teaching phase are guided through the exemplar in the peer-learning phase. Similarly, in ELPSO, multiple exemplars are selected forming a group of elite particles to guide other particles. A new directional update strategy for guidance of poorly performing particles has been introduced in the iSRPSO algorithm.

Furthermore, the basic PSO algorithm [10] has been selected as a baseline of performance. The detailed experimental procedures are given in the next section.

3 Numerical Assessment

3.1 Setup

The selected PSO variants are benchmarked on 24 functions (15 instances per function) of the BBOB testbed using $10^4 \times D$ function evaluations for different dimensions. The algorithms are implemented in MATLAB R2013b on Dell Precision T3600 machine having 16 Gb RAM and 64-bit operating system using 50 particles. The parameters of all the PSO variants are set to their standard values provided in the respective literature. A termination criteria has been used where the execution will stop if the target function value is achieved.

3.2 Performance Evaluation Procedure

The experiments are setup following the guidelines from [7], where each algorithm is evaluated on the functions [6, 8] multiple trials per function with specific target values. The algorithms are evaluated based on the number of function evaluations required to reach the target. The Expected Running Time (ERT) used in the figures and tables in this paper is dependent on a given target function value, $f_t = f_{\text{opt}} + \Delta f$. This ERT is computed over all relevant trials as the summation over all trails of the total number of executed function evaluations during each trail when the best function value has not achieved f_t , divided by the number of trials that actually reached f_t [7, 16]. For a given target Δf_t the rank-sum test has been used for providing the **Statistical significance**. For each trial, the test has been conducted using either the number of needed function evaluations to reach Δf_t (inverted and multiplied by -1), or, if the target was not reached, the best Δf -value achieved, measured only up to the smallest number of overall function evaluations for any unsuccessful trial under consideration.

3.3 Performance Evaluation Discussion

The experimental evaluations are reported in Figs. 1, 2, 3, 4 and 5; and in Tables 1 and 2. Overall, Fig. 1 shows that the ERT of all variants grows on a super-linear order with respect to the problem dimensionality. Furthermore, the Linear Slope function and the Griewank-Rosenbrock function appear to be the most challenging functions. From the rest of figures and tables, it has been observed that the performance gap between the algorithms varies over different budget settings of the function evaluations. Therefore, the discussion is based on three different budget settings of function evaluations, viz., expensive ($10^2 * D$ FEs), moderate ($10^3 * D$ FEs) and Cheap ($10^4 * D$ FEs). Under the three different budget settings, the following can be stated about the selected PSO variants.

Expensive Budget Settings ($10^2 * D$ FEs): It has been stated in the respective literature that the MePSO [24], SL-PSO [2] and SRPSO [21] algorithms have faster convergence characteristics which is evident from the experimental results where these algorithms are performing better than others on expensive budget

settings. The performance of all the algorithms are coupled in the lower dimensions where SL-PSO has a marginally better performance. In higher dimensions, there is a significant performance gap where MePSO has a remarkable performance on the separable functions whereas TPLPSO and CSO are leading the moderate and ill-conditioned functions categories. Further, the SRPSO, SL-PSO and ELPSO algorithms are providing better solutions to the multi-modal and weakly structured multi-modal functions. Overall, SL-PSO is performing better on the expensive budget setting followed closely by CSO and GAIWPSO on lower dimensions and by MePSO, SRPSO and ELPSO on the higher dimensions. In limited budget, more participants are required to simultaneously contribute towards convergence i.e. perform more exploitation. Human strategies applied in SL-PSO favours exploitation over exploration, therefore, SL-PSO is performing better.

Moderate Budget Settings ($10^3 * D$ FEs): The maximum gap among the performances of the algorithms has been observed under moderate budget settings. Here, MePSO and TPLPSO have a coupled performance in all the dimensions and are successfully providing better performance on most of the functions from each category with a remarkable performance gap. PSO, being the baseline of performance has the worst performance with a significant gap compared to the other algorithms. Both the MePSO and TPLPSO algorithms belongs to the same category of human learning principles where there is intelligent exploration and balanced exploitation of the search space. The peer learning strategy is present in both algorithms that has significantly contributed towards convergence by diverging the particles towards better search areas in the solution space.

Cheap Budget Settings ($10^4 * D$ FEs): In cheap budget setting, the particles are given fair amount of time for exploration of the search space. As a results, the performance of all the algorithms are coupled on lower dimensions. However, there is a slight gap observed in the higher dimensions where the performance of iSRPSO, TPLPSO and SRPSO are better than the others. This suggest that the concept of guidance present in iSRPSO and TPLPSO that first allows the particles to explore the search space and then guide the lesser performing particles towards optimum solution has a better convergence characteristic under the cheap budget settings.

To summarize, it can be stated that none of the PSO variants is capable enough of solving all the functions and it is also evident from the “No free lunch theorems” [26]. But it has been observed that an algorithm is exceptionally better than another on a class of problem (MePSO-separable functions, ELPSO-moderate functions and TPLPSO-weakly structured multi-modal functions). Also, different algorithms have varying performance on different budget settings. MePSO, SL-PSO and TPLPSO have performed better under limited and moderate budget settings whereas iSRPSO and TPLPSO have better performances under expensive budget settings. Further, robust solutions have also been observed by ELPSO (separable and moderate functions) and iSRPSO algorithms. Overall, SL-PSO can be termed as better performer under expensive budget settings followed by MePSO. Next, MePSO can be termed as better per-

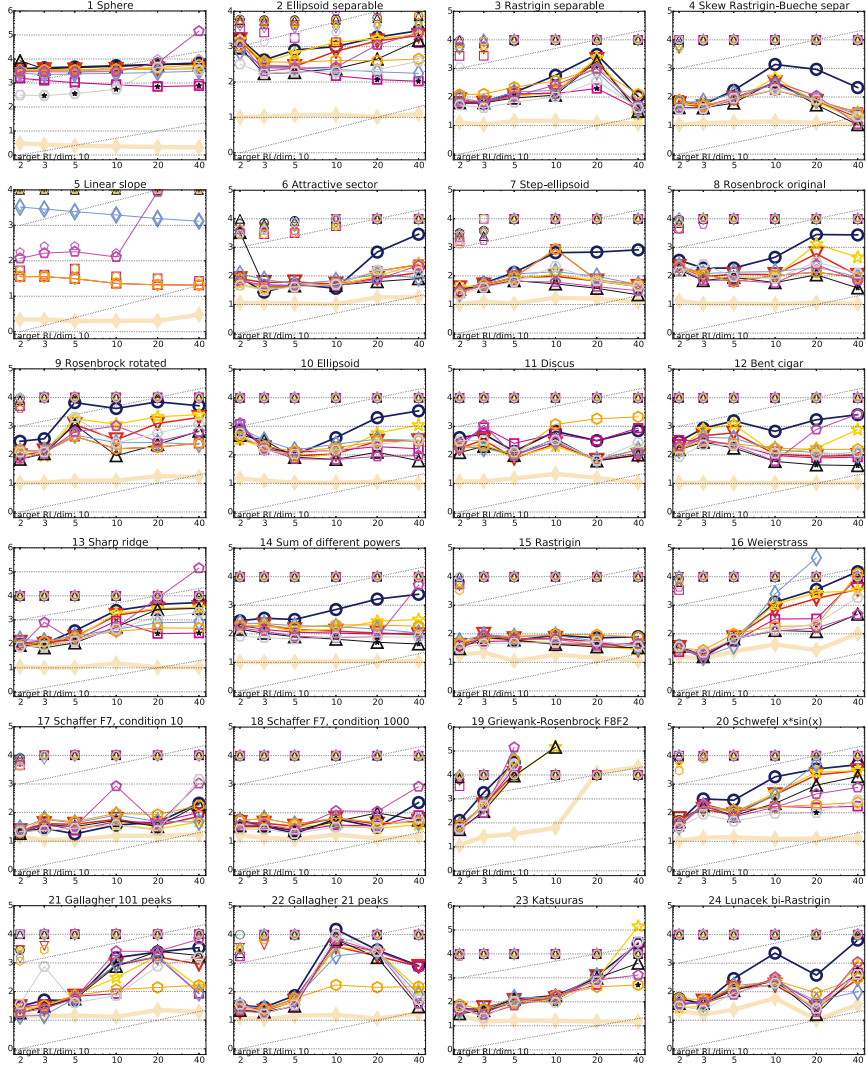


Fig. 1. Expected running time (ERT in number of f -evaluations as \log_{10} value) divided by dimension versus dimension. The target function value is chosen such that the best-GECCO2009 artificial algorithm just failed to achieve an ERT of $10 \times \text{DIM}$. Different symbols correspond to different algorithms given in the legend of f_1 and f_{24} . Light symbols give the maximum number of function evaluations from the longest trial divided by dimension. Black stars indicate a statistically better result compared to all other algorithms with $p < 0.01$ and Bonferroni correction number of dimensions (six). Legend: \circ :PSO, ∇ :SRPSO, \star :iSRPSO, \square :MePSO, \triangle :SL – PSO, \diamond :ELPSO, \circ :TPLPSO, \diamond :CSO, \circ :GAIWPSO

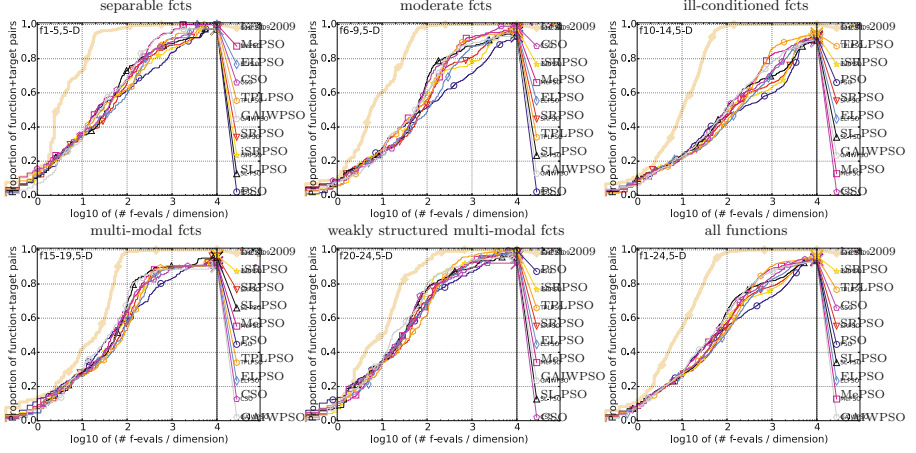


Fig. 2. Bootstrapped empirical cumulative distribution of the number of objective function evaluations divided by dimension (FEvals/DIM) for all functions and sub-groups in 5-D. The targets are chosen from $10^{[-8..2]}$ such that the bestGECCO2009 artificial algorithm just not reached them within a given budget of $k \times \text{DIM}$, with $k \in \{0.5, 1.2, 3, 10, 50\}$. The “best 2009” line corresponds to the best ERT observed during BBOB 2009 for each selected target

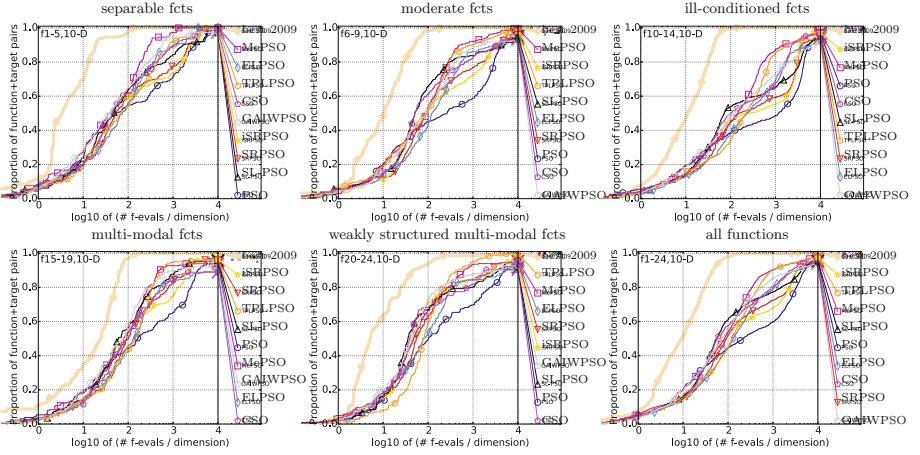


Fig. 3. Bootstrapped empirical cumulative distribution of the number of objective function evaluations divided by dimension (FEvals/DIM) for all functions and sub-groups in 10-D. The targets are chosen from $10^{[-8..2]}$ such that the bestGECCO2009 artificial algorithm just not reached them within a given budget of $k \times \text{DIM}$, with $k \in \{0.5, 1.2, 3, 10, 50\}$. The “best 2009” line corresponds to the best ERT observed during BBOB 2009 for each selected target

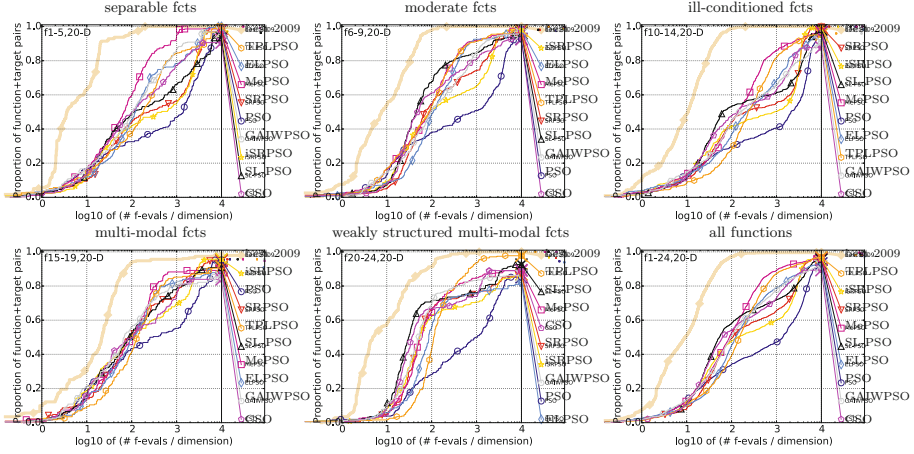


Fig. 4. Bootstrapped empirical cumulative distribution of the number of objective function evaluations divided by dimension (FEvals/DIM) for all functions and subgroups in 20-D. The targets are chosen from $10^{[-8..2]}$ such that the bestGECCO2009 artificial algorithm just not reached them within a given budget of $k \times \text{DIM}$, with $k \in \{0.5, 1.2, 3, 10, 50\}$. The “best 2009” line corresponds to the best ERT observed during BBOB 2009 for each selected target

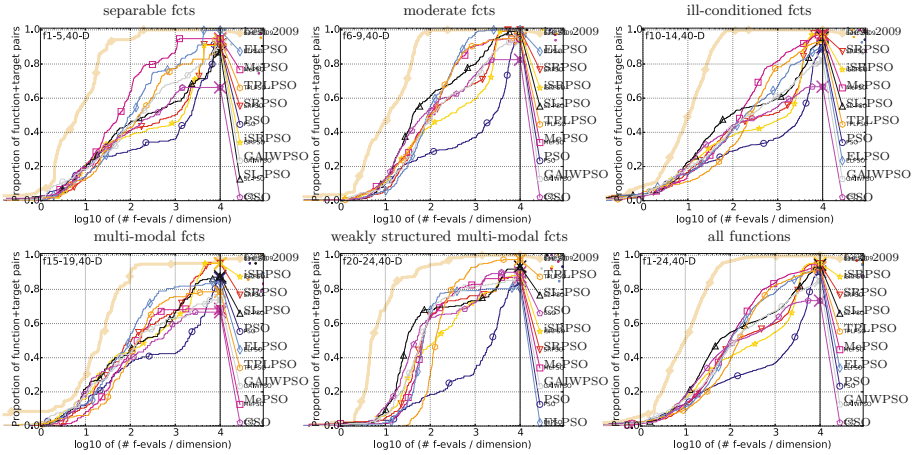


Fig. 5. Bootstrapped empirical cumulative distribution of the number of objective function evaluations divided by dimension (FEvals/DIM) for all functions and subgroups in 40-D. The targets are chosen from $10^{[-8..2]}$ such that the bestGECCO2009 artificial algorithm just not reached them within a given budget of $k \times \text{DIM}$, with $k \in \{0.5, 1.2, 3, 10, 50\}$. The “best 2009” line corresponds to the best ERT observed during BBOB 2009 for each selected target

Table 1. Expected running time (ERT in number of function evaluations) divided by the respective best ERT measured during BBOB-2009 in dimension 5. The ERT and in braces, as dispersion measure, the half difference between 90 and 10 %-tile of bootstrapped run lengths appear for each algorithm and run-length based target, the corresponding best ERT (preceded by the target Δf -value in *italics*) in the first row. #succ is the number of trials that reached the target value of the last column. The median number of conducted function evaluations is additionally given in *italics*, if the target in the last column was never reached. Entries, succeeded by a star, are statistically significantly better (according to the rank-sum test) when compared to all other algorithms of the table, with $p = 0.05$ or $p = 10^{-k}$ when the number k following the star is larger than 1, with Bonferroni correction by the number of instances.

#	#FEID	0.5	1.2	3	10	50	#FEID	0.5	1.2	3	10	50	#FEID	0.5	1.2	3	10	50	#FEID	0.5	1.2	3	10	50
1	PSO	2.3(7.18)	1.6(1.76)	1.6(1.12)	1.6(1.12)	1.6(1.12)	PS13	1.6(1.78)	2.3(2.74)	4.5(7.17)	1.6(1.12)	1.6(1.12)	PS14	1.6(1.78)	2.3(2.74)	4.5(7.17)	1.6(1.12)	1.6(1.12)	PS15	1.6(1.78)	2.3(2.74)	4.5(7.17)	1.6(1.12)	1.6(1.12)
2	SRPSO	2.4(1.3)	2.1(1.3)	1.4(1.04)	1.4(1.04)	1.4(1.04)	SRPSO	1.5(1.2)	3.4(3.1)	7.9(5.1)	1.5(1.2)	1.5(1.2)	SRPSO	1.5(1.2)	3.4(3.1)	7.9(5.1)	1.5(1.2)	1.5(1.2)	SRPSO	1.5(1.2)	3.4(3.1)	7.9(5.1)	1.5(1.2)	1.5(1.2)
3	MRPSO	1.8(1.2)	1.4(1.3)	1.4(1.04)	1.4(1.04)	1.4(1.04)	MRPSO	1.5(1.2)	3.4(3.1)	7.9(5.1)	1.5(1.2)	1.5(1.2)	MRPSO	1.5(1.2)	3.4(3.1)	7.9(5.1)	1.5(1.2)	1.5(1.2)	MRPSO	1.5(1.2)	3.4(3.1)	7.9(5.1)	1.5(1.2)	1.5(1.2)
4	SL-PSO	2.4(1.4)	4.4(4.4)	4.2(2.4)	4.2(2.4)	4.2(2.4)	SL-PSO	2.1(1.2)	2.4(1.3)	3.9(2.5)	1.8(1.2)	1.8(1.2)	SL-PSO	2.1(1.2)	2.4(1.3)	3.9(2.5)	1.8(1.2)	1.8(1.2)	SL-PSO	2.1(1.2)	2.4(1.3)	3.9(2.5)	1.8(1.2)	1.8(1.2)
5	TLPSO	1.2(1.0)	1.2(1.0)	1.2(1.0)	1.2(1.0)	1.2(1.0)	TLPSO	1.2(1.0)	1.2(1.0)	1.2(1.0)	1.2(1.0)	1.2(1.0)	TLPSO	1.2(1.0)	1.2(1.0)	1.2(1.0)	1.2(1.0)	1.2(1.0)	TLPSO	1.2(1.0)	1.2(1.0)	1.2(1.0)	1.2(1.0)	1.2(1.0)
6	CPSO	3.4(3.1)	6.7(8.1)	1.1(0.55)	1.1(0.55)	1.1(0.55)	CPSO	2.6(1.4)	4.4(4.4)	4.4(4.4)	2.6(1.4)	2.6(1.4)	CPSO	2.6(1.4)	4.4(4.4)	4.4(4.4)	2.6(1.4)	2.6(1.4)	CPSO	2.6(1.4)	4.4(4.4)	4.4(4.4)	2.6(1.4)	2.6(1.4)
7	GAIWPSO	1.6(1.4)	4.1(3.1)	1.4(1.0)	1.4(1.0)	1.4(1.0)	GAIWPSO	1.6(1.4)	4.1(3.1)	1.4(1.0)	1.6(1.4)	1.6(1.4)	GAIWPSO	1.6(1.4)	4.1(3.1)	1.4(1.0)	1.6(1.4)	1.6(1.4)	GAIWPSO	1.6(1.4)	4.1(3.1)	1.4(1.0)	1.6(1.4)	1.6(1.4)
8	PSO	1.6(1.78)	2.3(2.74)	1.6(1.12)	1.6(1.12)	1.6(1.12)	PS16	1.6(1.78)	2.3(2.74)	4.5(7.17)	1.6(1.12)	1.6(1.12)	PS17	1.6(1.78)	2.3(2.74)	4.5(7.17)	1.6(1.12)	1.6(1.12)	PS18	1.6(1.78)	2.3(2.74)	4.5(7.17)	1.6(1.12)	1.6(1.12)
9	SRPSO	1.3(1.0)	1.8(1.0)	1.8(1.0)	1.8(1.0)	1.8(1.0)	SRPSO	1.3(1.0)	1.8(1.0)	1.8(1.0)	1.3(1.0)	1.3(1.0)	SRPSO	1.3(1.0)	1.8(1.0)	1.8(1.0)	1.3(1.0)	1.3(1.0)	SRPSO	1.3(1.0)	1.8(1.0)	1.8(1.0)	1.3(1.0)	1.3(1.0)
10	MRPSO	1.8(1.2)	1.4(1.3)	1.4(1.04)	1.4(1.04)	1.4(1.04)	MRPSO	1.5(1.2)	3.4(3.1)	7.9(5.1)	1.5(1.2)	1.5(1.2)	MRPSO	1.5(1.2)	3.4(3.1)	7.9(5.1)	1.5(1.2)	1.5(1.2)	MRPSO	1.5(1.2)	3.4(3.1)	7.9(5.1)	1.5(1.2)	1.5(1.2)
11	SL-PSO	2.4(1.4)	4.4(4.4)	4.2(2.4)	4.2(2.4)	4.2(2.4)	SL-PSO	2.1(1.2)	2.4(1.3)	3.9(2.5)	1.8(1.2)	1.8(1.2)	SL-PSO	2.1(1.2)	2.4(1.3)	3.9(2.5)	1.8(1.2)	1.8(1.2)	SL-PSO	2.1(1.2)	2.4(1.3)	3.9(2.5)	1.8(1.2)	1.8(1.2)
12	TLPSO	1.2(1.0)	1.2(1.0)	1.2(1.0)	1.2(1.0)	1.2(1.0)	TLPSO	1.2(1.0)	1.2(1.0)	1.2(1.0)	1.2(1.0)	1.2(1.0)	TLPSO	1.2(1.0)	1.2(1.0)	1.2(1.0)	1.2(1.0)	1.2(1.0)	TLPSO	1.2(1.0)	1.2(1.0)	1.2(1.0)	1.2(1.0)	1.2(1.0)
13	CPSO	3.4(3.1)	6.7(8.1)	1.1(0.55)	1.1(0.55)	1.1(0.55)	CPSO	2.6(1.4)	4.4(4.4)	4.4(4.4)	2.6(1.4)	2.6(1.4)	CPSO	2.6(1.4)	4.4(4.4)	4.4(4.4)	2.6(1.4)	2.6(1.4)	CPSO	2.6(1.4)	4.4(4.4)	4.4(4.4)	2.6(1.4)	2.6(1.4)
14	GAIWPSO	1.6(1.4)	4.1(3.1)	1.4(1.0)	1.4(1.0)	1.4(1.0)	GAIWPSO	1.6(1.4)	4.1(3.1)	1.4(1.0)	1.6(1.4)	1.6(1.4)	GAIWPSO	1.6(1.4)	4.1(3.1)	1.4(1.0)	1.6(1.4)	1.6(1.4)	GAIWPSO	1.6(1.4)	4.1(3.1)	1.4(1.0)	1.6(1.4)	1.6(1.4)
15	PSO	1.6(1.78)	2.3(2.74)	1.6(1.12)	1.6(1.12)	1.6(1.12)	PS19	1.6(1.78)	2.3(2.74)	4.5(7.17)	1.6(1.12)	1.6(1.12)	PS20	1.6(1.78)	2.3(2.74)	4.5(7.17)	1.6(1.12)	1.6(1.12)	PS21	1.6(1.78)	2.3(2.74)	4.5(7.17)	1.6(1.12)	1.6(1.12)
16	SRPSO	1.3(1.0)	1.8(1.0)	1.8(1.0)	1.8(1.0)	1.8(1.0)	SRPSO	1.3(1.0)	1.8(1.0)	1.8(1.0)	1.3(1.0)	1.3(1.0)	SRPSO	1.3(1.0)	1.8(1.0)	1.8(1.0)	1.3(1.0)	1.3(1.0)	SRPSO	1.3(1.0)	1.8(1.0)	1.8(1.0)	1.3(1.0)	1.3(1.0)
17	MRPSO	1.8(1.2)	1.4(1.3)	1.4(1.04)	1.4(1.04)	1.4(1.04)	MRPSO	1.5(1.2)	3.4(3.1)	7.9(5.1)	1.5(1.2)	1.5(1.2)	MRPSO	1.5(1.2)	3.4(3.1)	7.9(5.1)	1.5(1.2)	1.5(1.2)	MRPSO	1.5(1.2)	3.4(3.1)	7.9(5.1)	1.5(1.2)	1.5(1.2)
18	SL-PSO	2.4(1.4)	4.4(4.4)	4.2(2.4)	4.2(2.4)	4.2(2.4)	SL-PSO	2.1(1.2)	2.4(1.3)	3.9(2.5)	1.8(1.2)	1.8(1.2)	SL-PSO	2.1(1.2)	2.4(1.3)	3.9(2.5)	1.8(1.2)	1.8(1.2)	SL-PSO	2.1(1.2)	2.4(1.3)	3.9(2.5)	1.8(1.2)	1.8(1.2)
19	TLPSO	1.2(1.0)	1.2(1.0)	1.2(1.0)	1.2(1.0)	1.2(1.0)	TLPSO	1.2(1.0)	1.2(1.0)	1.2(1.0)	1.2(1.0)	1.2(1.0)	TLPSO	1.2(1.0)	1.2(1.0)	1.2(1.0)	1.2(1.0)	1.2(1.0)	TLPSO	1.2(1.0)	1.2(1.0)	1.2(1.0)	1.2(1.0)	1.2(1.0)
20	CPSO	3.4(3.1)	6.7(8.1)	1.1(0.55)	1.1(0.55)	1.1(0.55)	CPSO	2.6(1.4)	4.4(4.4)	4.4(4.4)	2.6(1.4)	2.6(1.4)	CPSO	2.6(1.4)	4.4(4.4)	4.4(4.4)	2.6(1.4)	2.6(1.4)	CPSO	2.6(1.4)	4.4(4.4)	4.4(4.4)	2.6(1.4)	2.6(1.4)
21	GAIWPSO	1.6(1.4)	4.1(3.1)	1.4(1.0)	1.4(1.0)	1.4(1.0)	GAIWPSO	1.6(1.4)	4.1(3.1)	1.4(1.0)	1.6(1.4)	1.6(1.4)	GAIWPSO	1.6(1.4)	4.1(3.1)	1.4(1.0)	1.6(1.4)	1.6(1.4)	GAIWPSO	1.6(1.4)	4.1(3.1)	1.4(1.0)	1.6(1.4)	1.6(1.4)
22	PSO	1.6(1.78)	2.3(2.74)	1.6(1.12)	1.6(1.12)	1.6(1.12)	PS22	1.6(1.78)	2.3(2.74)	4.5(7.17)	1.6(1.12)	1.6(1.12)	PS23	1.6(1.78)	2.3(2.74)	4.5(7.17)	1.6(1.12)	1.6(1.12)	PS24	1.6(1.78)	2.3(2.74)	4.5(7.17)	1.6(1.12)	1.6(1.12)
23	SRPSO	1.3(1.0)	1.8(1.0)	1.8(1.0)	1.8(1.0)	1.8(1.0)	SRPSO	1.3(1.0)	1.8(1.0)	1.8(1.0)	1.3(1.0)	1.3(1.0)	SRPSO	1.3(1.0)	1.8(1.0)	1.8(1.0)	1.3(1.0)	1.3(1.0)	SRPSO	1.3(1.0)	1.8(1.0)	1.8(1.0)	1.3(1.0)	1.3(1.0)
24	MRPSO	1.8(1.2)	1.4(1.3)	1.4(1.04)	1.4(1.04)	1.4(1.04)	MRPSO	1.5(1.2)	3.4(3.1)	7.9(5.1)	1.5(1.2)	1.5(1.2)	MRPSO	1.5(1.2)	3.4(3.1)	7.9(5.1)	1.5(1.2)	1.5(1.2)	MRPSO	1.5(1.2)	3.4(3.1)	7.9(5.1)	1.5(1.2)	1.5(1.2)
25	SL-PSO	2.4(1.4)	4.4(4.4)	4.2(2.4)	4.2(2.4)	4.2(2.4)	SL-PSO	2.1(1.2)	2.4(1.3)	3.9(2.5)	1.8(1.2)	1.8(1.2)	SL-PSO	2.1(1.2)	2.4(1.3)	3.9(2.5)	1.8(1.2)	1.8(1.2)	SL-PSO	2.1(1.2)	2.4(1.3)	3.9(2.5)	1.8(1.2)	1.8(1.2)
26	TLPSO	1.2(1.0)	1.2(1.0)	1.2(1.0)	1.2(1.0)	1.2(1.0)	TLPSO	1.2(1.0)	1.2(1.0)	1.2(1.0)	1.2(1.0)	1.2(1.0)	TLPSO	1.2(1.0)	1.2(1.0)	1.2(1.0)	1.2(1.0)	1.2(1.0)	TLPSO	1.2(1.0)	1.2(1.0)	1.2(1.0)	1.2(1.0)	1.2(1.0)
27	CPSO	3.4(3.1)	6.7(8.1)	1.1(0.55)	1.1(0.55)	1.1(0.55)	CPSO	2.6(1.4)	4.4(4.4)	4.4(4.4)	2.6(1.4)	2.6(1.4)	CPSO	2.6(1.4)	4.4(4.4)	4.4(4.4)	2.6(1.4)	2.6(1.4)	CPSO	2.6(1.4)	4.4(4.4)	4.4(4.4)	2.6(1.4)	2.6(1.4)
28	GAIWPSO	1.6(1.4)	4.1(3.1)	1.4(1.0)	1.4(1.0)	1.4(1.0)	GAIWPSO	1.6(1.4)	4.1(3.1)	1.4(1.0)	1.6(1.4)	1.6(1.4)	GAIWPSO	1.6(1.4)	4.1(3.1)	1.4(1.0)	1.6(1.4)	1.6(1.4)	GAIWPSO	1.6(1.4)	4.1(3.1)	1.4(1.0)	1.6(1.4)	1.6(1.4)
29	PSO	1.6(1.78)	2.3(2.74)	1.6(1.12)	1.6(1.12)	1.6(1.12)	PS25	1.6(1.78)	2.3(2.74)	4.5(7.17)	1.6(1.12)	1.6(1.12)	PS26	1.6(1.78)	2.3(2.74)	4.5(7.17)	1.6(1.12)	1.6(1.12)	PS27	1.6(1.78)	2.3(2.74)	4.5(7.17)	1.6(1.12)	1.6(1.12)
30	SRPSO	1.3(1.0)	1.8(1.0)	1.8(1.0)	1.8(1.0)	1.8(1.0)	SRPSO	1.3(1.0)	1.8(1.0)	1.8(1.0)	1.3(1.0)	1.3(1.0)	SRPSO	1.3(1.0)	1.8(1.0)	1.8(1.0)	1.3(1.0)	1.3(1.0)	SRPSO	1.3(1.0)	1.8(1.0)	1.8(1.0)	1.3(1.0)	1.3(1.0)
31	MRPSO	1.8(1.2)	1.4(1.3)	1.4(1.04)	1.4(1.04)	1.4(1.04)	MRPSO	1.5(1.2)	3.4(3.1)	7.9(5.1)	1.5(1.2)	1.5(1.2)	MRPSO	1.5(1.2)	3.4(3.1)	7.9(5.1)	1.5(1.2)	1.5(1.2)	MRPSO	1.5(1.2)	3.4(3.1)	7.9(5.1)	1.5(1.2)	1.5(1.2)
32	SL-PSO	2.4(1.4)	4.4(4.4)	4.2(2.4)	4.2(2.4)	4.2(2.4)	SL-PSO	2.1(1.2)	2.4(1.3)	3.9(2.5)	1.8(1.2)	1.8(1.2)	SL-PSO	2.1(1.2)	2.4(1.3)	3.9(2.5)	1.8(1.2)	1.8(1.2)	SL-PSO	2.1(1.2)	2.4(1.3)	3.9(2.5)	1.8(1.2)	1.8(1.2)
33	TLPSO	1.2(1.0)	1.2(1.0)	1.2(1.0)	1.2(1.0)	1.2(1.0)	TLPSO	1.2(1.0)	1.2(1.0)	1.2(1.0)	1.2(1.0)	1.2(1.0)	TLPSO	1.2(1.0)	1.2(1.0)	1.2(1.0)	1.2(1.0)	1.2(1.0)	TLPSO	1.2(1.0)	1.2(1.0)	1.2(1.0)	1.2(1.0)	1.2(1.0)
34	CPSO	3.4(3.1)	6.7(8.1)	1.1(0.55)	1.1(0.55)	1.1(0.55)	CPSO	2.6(1.4)	4.4(4.4)	4.4(4.4)	2.6(1.4)	2.6(1.4)	CPSO	2.6(1.4)	4.4(4.4)	4.4(4.4)	2.6(1.4)	2.6(1.4)	CPSO	2.6(1.4)	4.4(4.4)	4.4(4.4)	2.6(1.4)	2.6(1.4)
35	GAIWPSO	1.6(1.4)	4.1(3.1)	1.4(1.0)	1.4(1.0)	1.4(1.0)	GAIWPSO	1.6(1.4)	4.1(3.1)	1.4(1.0)	1.6(1.4)	1.6(1.4)	GAIWPSO	1.6(1.4)	4.1(3.1)	1.4(1.0)	1.6(1.4)	1.6(1.4)	GAIWPSO	1.6(1.4)	4.1(3.1)	1.4(1.0)	1.6(1.4)	1.6(1.4)
36	PSO	1.6(1.78)	2.3(2.74)	1.6(1.12)	1.6(1.12)	1.6(1.12)	PS28	1.6(1.78)	2.3(2.74)	4.5(7.17)	1.6(1.12)	1.6(1.12)	PS29	1.6(1.78)	2.3(2.74)	4.5(7.17)	1.6(1.12)	1.6(1.12)	PS30	1.6(1.78)	2.3(2.74)	4.5(7.17)	1.6(1.12)	1.6(1.12)
37	SRPSO	1.3(1.0)	1.8(1.0)	1.8(1.0)	1.8(1.0)	1.8(1.0)	SRPSO	1.3(1.0)	1.8(1.0)	1.8(1.0)	1.3(1.0)	1.3(1.0)	SRPSO	1.3(1.0)	1.8(1.0)	1.8(1.0)	1.3(1.0)	1.3(1.0)	SRPSO	1.3(1.0)	1.8(1.0)	1.8(1.0)	1.3(1.0)	1.3(1.0)
38	MRPSO	1.8(1.2)	1.4(1.3)	1.4(1.04)	1.4(1.04)	1.4(1.04)	MRPSO	1.5(1.2)	3.4(3.1)	7.9(5.1)	1.5(1.2)	1.5(1.2)	MRPSO	1.5(1.2)	3.4(3.1)	7.9(5.1)	1.5(1.2)	1.5(1.2)	MRPSO	1.5(1.2)	3.4(3.1)	7.9(5.1)	1.5(1.2)	1.5(1.2)
39	SL-PSO	2.4(1.4)	4.4(4.4)	4.2(2.4)	4.2(2.4)	4.2(2.4)	SL-PSO	2.1(1.2)	2.4(1.3)	3.9(2.5)	1.8(1.2)	1.8(1.2)	SL-PSO	2.1(1.2)	2.4(1.3)	3.9(2.5)	1.8(1.2)	1.8(1.2)	SL-PSO	2.1(1.2)	2.4(1.3)	3.9(2.5)	1.8(1.2)	1.8(1.2)
40	TLPSO	1.2(1.0)	1.2(1.0)	1.2(1.0)	1.2(1.0)	1.2(1.0)	TLPSO	1.2(1.0)	1.2(1.0)	1.2(1.0)	1.2(1.0)	1.2(1.0)	TLPSO	1.2(1.0)	1.2(1.0)	1.2(1.0)	1.2(1.0)	1.2(1.0)	TLPSO	1.2(1.0)	1.2(1.0)	1.2(1.0)	1.2(1.0)	1.2(1.0)
41	CPSO	3.4(3.1)	6.7(8.1)	1.1(0.55)	1.1(0.55)	1.1(0.55)	CPSO	2.6(1.4)	4.4(4.4)	4.4(4.4)	2.6(1.4)	2.6(1.4)	CPSO	2.6(1.4)	4.4(4.4)	4.4(4.4)	2.6(1.4)	2.6(1.4)	CPSO	2.6(1.4)	4.4(4.4)	4.4(4.4)	2.6(1.4)	2.6(1.4)
42	GAIWPSO	1.6(1.4)	4.1(3.1)	1.4(1.0)	1.4(1.0)	1.4(1.0)	GAIWPSO	1.6(1.4)	4.1(3.1)	1.4(1.0)	1.6(1.4)	1.6(1.4)	GAIWPSO	1.6(1.4)	4.1(3.1)	1.4(1.0)	1.6(1.4)	1.6(1.4)	GAIWPSO	1.6(1.4)	4.1(3.1)	1.4(1.0)	1.6(1.4)	1.6(1.4)
43	PSO	1.6(1.78)	2.3(2.74)	1.6(1.12)	1.6(1.12)	1.6(1.12)	PS31	1.6(1.78)	2.3(2.74)	4.5(7.17)	1.6(1.12)	1.6(

Table 2. Expected running time (ERT in number of function evaluations) divided by the respective best ERT measured during BBOB-2009 in dimension 20. The ERT and in braces, as dispersion measure, the half difference between 90 and 10%-tile of bootstrapped run lengths appear for each algorithm and run-length based target, the corresponding best ERT (preceded by the target Δf -value in *italics*) in the first row. #succ is the number of trials that reached the target value of the last column. The median number of conducted function evaluations is additionally given in *italics*, if the target in the last column was never reached. Entries, succeeded by a star, are statistically significantly better (according to the rank-sum test) when compared to all other algorithms of the table, with $p = 0.05$ or $p = 10^{-k}$ when the number k following the star is larger than 1, with Bonferroni correction by the number of instances.

#FFs/D	0.5	1.2	3	10	50	#succ	#FFs/D	0.5	1.2	3	10	50	#succ
F													
PSO	8.5(4)	9.3(4)	2769(60)	2769(88)	2769(80)	15/15	PSO	6.0(2)	8.7(5)	56(55)	418(34)	167(32)	8/15
SRPSO	3(0)	1.7(0)	1799(21)	1799(16)	1799(20)	15/15	SRPSO	1(0)	1(0)	1(0)	205(9)	59(0)	8/15
ISRPFO	100(5)	8.8(1)	1799(10)	1799(10)	1799(30)	15/15	ISRPFO	7.9(1)	7.5(2)	12(2)	280(12)	71(9)	12/15
MaPSO	10(2)	10(1)	328(18)	328(23)	328(31)	15/15	MaPSO	6.2(4)	6(2)	14(2)	255(17)	108(20)	8/15
SL-PSO	10(0)	1.6(1)	2026(43)	2026(43)	2026(21)	15/15	SL-PSO	7.9(1)	7.9(1)	8.7(2)	258(38)	113(17)	10/15
TLPSO	17(7)	16(0)	1281(23)	1281(50)	1281(47)	15/15	TLPSO	1(0)	1(0)	1(0)	258(38)	113(17)	10/15
GAIWPSO	20(15)	10(5)	2026(43)	2026(43)	2026(40)	15/15	GAIWPSO	1(0)	1(0)	1(0)	258(38)	113(17)	10/15
CSO	3.0e+4	2.0(1)	16.4(1)	16.4(1)	16.4(1)	15/15	CSO	3.0e+4	2.0(1)	16(4)	16(4)	16(4)	15/15
GAIWPSO	8.9(4)	32(5)	2102(2359)	2102(2324)	2102(1197)	13/15	GAIWPSO	6.7(4)	7.2(1)	10(2)	707(956)	346(222)	4/15
GAIWPSO	8.9(4)	32(5)	2102(2359)	2102(2324)	2102(1197)	13/15	GAIWPSO	6.7(4)	7.2(1)	10(2)	707(956)	346(222)	4/15
F													
PSO	1.4e+129	4.6e+129	1.7e+129	1.7e+129	1.7e+129	1.7e+129	PSO	1.4e+129	4.6e+129	1.7e+129	1.7e+129	1.7e+129	1.7e+129
PSO	3.0(0.7)	1.4(0.6)	109(33)	109(33)	109(34)	15/15	PSO	10(5)	10(3)	13(3)	150(41)	164(10)	15/15
SRPSO	1.5(2)	1.8(1)	26(10)	26(10)	26(10)	15/15	SRPSO	8.7(2)	8.7(2)	8.7(2)	10(1)	88(3)	15/15
ISRPFO	2.5(3)	2.9(3)	35(22)	35(22)	35(22)	15/15	ISRPFO	11(4)	6.4(3)	6.2(1)	23(14)	68(4)	15/15
MaPSO	1.6(1)	1.6(1)	16(4)	16(4)	16(4)	15/15	MaPSO	10(5)	8.6(3)	8.2(3)	4(0)	84(7)	15/15
SL-PSO	1.3(0)	1.3(1)	21(1)	21(1)	21(1)	15/15	SL-PSO	13(3)	7.4(4)	6.0(2)	11(1)	19(1)	15/15
TLPSO	1.7(7)	2(0)	21(1)	21(1)	21(1)	15/15	TLPSO	1(0)	1(0)	1(0)	258(38)	113(17)	10/15
GAIWPSO	1.6(3)	2.0(1)	21(1)	21(1)	21(1)	15/15	GAIWPSO	1(0)	1(0)	1(0)	258(38)	113(17)	10/15
CSO	1.3e+4	2.0(1)	16.4(1)	16.4(1)	16.4(1)	15/15	CSO	1.3e+4	2.0(1)	16(4)	16(4)	16(4)	15/15
GAIWPSO	8.9(4)	32(5)	2102(2359)	2102(2324)	2102(1197)	13/15	GAIWPSO	6.7(4)	12(5)	19(3)	127(32)	196(20)	9/15
GAIWPSO	8.9(4)	32(5)	2102(2359)	2102(2324)	2102(1197)	13/15	GAIWPSO	6.7(4)	12(5)	19(3)	127(32)	196(20)	9/15
F													
PSO	2.0(1)	2.0(1)	148(25)	148(25)	148(25)	15/15	PSO	2.0(1)	2.0(1)	2.0(1)	2.0(1)	2.0(1)	15/15
SRPSO	5.2(2)	7.8(3)	41(22)	41(22)	41(22)	15/15	SRPSO	6.2(4)	4.6(4)	4.3(3)	50(23)	77(25)	15/15
ISRPFO	4.0(4)	4.0(4)	148(25)	148(25)	148(25)	15/15	ISRPFO	5.1(7)	4.3(2)	3.5(2)	37(19)	47(19)	15/15
MaPSO	3.1(3)	7.3(2)	20(6)	16(3)	16(3)	15/15	MaPSO	6.7(2)	4.2(1)	2.6(1)	7.0(7)	4(1)	15/15
SL-PSO	3.1(3)	7.2(2)	14(10)	14(10)	14(10)	15/15	SL-PSO	5.3(2)	3.8(1)	3.4(1)	12(4)	38(14)	15/15
TLPSO	10(7)	22(0)	70(18)	77(15)	38(16)	15/15	TLPSO	1(0)	1(0)	1(0)	258(38)	113(17)	10/15
GAIWPSO	5.2(2)	7.4(5)	24(11)	28(10)	12(3)	15/15	GAIWPSO	6.2(4)	4.6(4)	4.3(3)	50(23)	77(25)	15/15
GAIWPSO	5.2(2)	7.4(5)	24(11)	28(10)	12(3)	15/15	GAIWPSO	6.2(4)	4.6(4)	4.3(3)	50(23)	77(25)	15/15
F													
PSO	6.6e+122	4.6e+122	1.7e+122	1.7e+122	1.7e+122	1.7e+122	PSO	6.6e+122	4.6e+122	1.7e+122	1.7e+122	1.7e+122	1.7e+122
PSO	13(0)	19(1)	74(18)	148(25)	148(25)	15/15	PSO	13(0)	13(1)	13(0)	13(0)	13(0)	15/15
SRPSO	8.0(5)	9.8(2)	5.6(3)	38(36)	31(2)	15/15	SRPSO	1.6(4)	11(3)	12(0)	64(32)	54(49)	15/15
ISRPFO	7.4(2)	8.1(2)	31(12)	31(12)	31(12)	15/15	ISRPFO	4.9(8)	32(27)	10(17)	10(12)	44(4)	15/15
MaPSO	3(5)	7.6(5)	5.9(1)	10(3)	4.6(3)	15/15	MaPSO	4.4(4)	20(10)	12(16)	12(16)	44(43)	15/15
SL-PSO	8.8(3)	8.3(3)	3.0(4)	10(3)	4.1(2)	15/15	SL-PSO	4.8(3)	8.1(2)	4.4(2)	4(2)	7(2)	15/15
TLPSO	11(0)	7.6(2)	8.0(2)	14(4)	30(3)	15/15	TLPSO	4.0(4)	46(7)	170(2120)	120(2120)	8(6)	15/15
CSO	6.8(4)	8.7(1)	8.8(5)	34(23)	35(58)	15/15	CSO	2.8(3)	5(0)	5.9(5)	5.9(5)	6.2(3)	15/15
GAIWPSO	5.8(5)	6.0(1)	6.0(1)	10(3)	10(3)	15/15	GAIWPSO	5.8(5)	10(5)	10(5)	5.1(18)	5.1(18)	15/15
GAIWPSO	5.8(5)	6.0(1)	6.0(1)	10(3)	10(3)	15/15	GAIWPSO	5.8(5)	10(5)	10(5)	5.1(18)	5.1(18)	15/15
F													
PSO	1.6e+124	4.6e+124	1.7e+124	1.7e+124	1.7e+124	1.7e+124	PSO	1.6e+124	4.6e+124	1.7e+124	1.7e+124	1.7e+124	1.7e+124
PSO	11(1)	20(8)	∞	∞	∞	15/15	PSO	11(1)	11(1)	11(1)	11(1)	11(1)	15/15
SRPSO	11(1)	20(8)	∞	∞	∞	15/15	SRPSO	11(1)	11(1)	11(1)	11(1)	11(1)	15/15
ISRPFO	11(1)	20(8)	∞	∞	∞	15/15	ISRPFO	11(1)	11(1)	11(1)	11(1)	11(1)	15/15
MaPSO	2.4(1)	8.2(0.2)	10(2)	10(4)	10(4)	15/15	MaPSO	2.4(1)	8.2(0.2)	10(2)	10(4)	10(4)	15/15
SL-PSO	11(1)	20(8)	∞	∞	∞	15/15	SL-PSO	11(1)	11(1)	11(1)	11(1)	11(1)	15/15
TLPSO	11(1)	20(8)	∞	∞	∞	15/15	TLPSO	11(1)	11(1)	11(1)	11(1)	11(1)	15/15
GAIWPSO	11(1)	20(8)	∞	∞	∞	15/15	GAIWPSO	11(1)	11(1)	11(1)	11(1)	11(1)	15/15
CSO	11(1)	20(8)	∞	∞	∞	15/15	CSO	11(1)	11(1)	11(1)	11(1)	11(1)	15/15
GAIWPSO	11(1)	20(8)	∞	∞	∞	15/15	GAIWPSO	11(1)	11(1)	11(1)	11(1)	11(1)	15/15
F													
PSO	1.6e+124	4.6e+124	1.7e+124	1.7e+124	1.7e+124	1.7e+124	PSO	1.6e+124	4.6e+124	1.7e+124	1.7e+124	1.7e+124	1.7e+124
PSO	13(0)	19(1)	74(18)	148(25)	148(25)	15/15	PSO	13(0)	13(1)	13(0)	13(0)	13(0)	15/15
SRPSO	8.0(5)	9.8(2)	5.6(3)	38(36)	31(2)	15/15	SRPSO	1.6(4)	11(3)	12(0)	64(32)	54(49)	15/15
ISRPFO	7.4(2)	8.1(2)	31(12)	31(12)	31(12)	15/15	ISRPFO	4.9(8)	32(27)	10(17)	10(12)	44(4)	15/15
MaPSO	3(5)	7.6(5)	5.9(1)	10(3)	4.6(3)	15/15	MaPSO	4.4(4)	20(10)	12(16)	12(16)	44(43)	15/15
SL-PSO	8.8(3)	8.3(3)	3.0(4)	10(3)	4.1(2)	15/15	SL-PSO	4.8(3)	8.1(2)	4.4(2)	4(2)	7(2)	15/15
TLPSO	11(0)	7.6(2)	8.0(2)	14(4)	30(3)	15/15	TLPSO	4.0(4)	46(7)	170(2120)	120(2120)	8(6)	15/15
CSO	6.8(4)	8.7(1)	8.8(5)	34(23)	35(58)	15/15	CSO	2.8(3)	5(0)	5.9(5)	5.9(5)	6.2(3)	15/15
GAIWPSO	5.8(5)	6.0(1)	6.0(1)	10(3)	10(3)	15/15	GAIWPSO	5.8(5)	10(5)	10(5)	5.1(18)	5.1(18)	15/15
GAIWPSO	5.8(5)	6.0(1)	6.0(1)	10(3)	10(3)	15/15	GAIWPSO	5.8(5)	10(5)	10(5)	5.1(18)	5.1(18)	15/15
F													
PSO	1.6e+124	4.6e+124	1.7e+124	1.7e+124	1.7e+124	1.7e+124	PSO	1.6e+124	4.6e+124	1.7e+124	1.7e+124	1.7e+124	1.7e+124
PSO	13(0)	19(1)	74(18)	148(25)	148(25)	15/15	PSO	13(0)	13(1)	13(0)	13(0)	13(0)	15/15
SRPSO	8.0(5)	9.8(2)	5.6(3)	38(36)	31(2)	15/15	SRPSO	1.6(4)	11(3)	12(0)	64(32)	54(49)	15/15
ISRPFO	7.4(2)	8.1(2)	31(12)	31(12)	31(12)	15/15	ISRPFO	4.9(8)	32(27)	10(17)	10(12)	44(4)	15/15
MaPSO	3(5)	7.6(5)	5.9(1)	10(3)	4.6(3)	15/15	MaPSO	4.4(4)	20(10)	12(16)	12(16)	44(43)	15/15
SL-PSO	8.8(3)	8.3(3)	3.0(4)	10(3)	4.1(2)	15/15	SL-PSO	4.8(3)	8.1(2)	4.4(2)	4(2)	7(2)	15/15
TLPSO	11(0)	7.6(2)	8.0(2)	14(4)	30(3)	15/15	TLPSO	4.0(4)	46(7)	170(2120)	120(2120)	8(6)	15/15
CSO	6.8(4)	8.7(1)	8.8(5)	34(23)	35(58)	15/15	CSO	2.8(3)	5(0)	5.9(5)	5.9(5)	6.2(3)	15/15
GAIWPSO	5.8(5)	6.0(1)	6.0(1)	10(3)	10(3)	15/15	GAIWPSO	5.8(5)	10(5)	10(5)	5.1(18)	5.1(18)	15/15
GAIWPSO	5.8(5)	6.0(1)	6.0(1)	10(3)	10(3)	15/15	GAIWPSO	5.8(5)	10(5)	10(5)	5.1(18)	5.1(18)	15/15
F													
PSO	1.6e+124	4.6e+124	1.7e+124	1.7e+124	1.7e+124	1.7e+124	PSO	1.6e+124	4.6e+124	1.7e+124	1.7e+124	1.7e+124	1.7e+124
PSO	13(0)	19(1)	74(18)	148(25)	148(25)	15/15	PSO	13(0)	13(1)	13(0)	13(0)	13(0)	15/15
SRPSO	8.0(5)	9.8(2)	5.6(3)	38(36)	31(2)	15/15	SRPSO	1.6(4)	11(3)	12(0)	64(32)	54(49)	15/15
ISRPFO	7.4(2)	8.1(2)	31(12)	31(12)	31(12)	15/15	ISRPFO	4.9(8)	32(27)	10(17)	10(12)	44(4)	15/15
MaPSO	3(5)	7.6(5)	5.9(1)	10(3)	4.6(3)	15/15	MaPSO	4.4(4)	20(10)	12(16)	12(16)	44(43)	15/15
SL-PSO	8.8(3)	8.3(3)	3.0(4)	10(3)	4.1(2)	15/15	SL-PSO	4.8(3)	8.1(2)	4.4(2)	4(2)	7(2)	15/15
TLPSO	11(0)	7.6(2)	8.0(2)	14(4)	30(3)	15/15	TLPSO	4.0(4)	46(7)	170(2120)	120(2120)	8(6)	15/15
CSO	6.8(4)	8.7(1)	8.8(5)	34(23)	35(58)	15/15	CSO	2.8(3)	5(0)	5.9(5)	5.9(5)	6.2(3)	15/15
GAIWPSO	5.8(5)	6.0(1)	6.0(1)	10(3)	10(3)	15/15	GAIWPSO	5.8(5)	10(5)	10(5)	5.1(18)	5.1(18)	15/15
GAIWPSO	5.8(5)	6.0(1)	6.0(1)	10(3)	10(3)	15/15	GAIWPSO	5.8(5)	10(5)	10(5)	5.1(18)	5.1(18)	15/15
F													
PSO	1.6e+124	4.6e+124	1.7e+124	1.7e+124	1.7e+124	1.7e+124	PSO	1.6e+124	4.6e+124	1.7e+124	1.7e+124	1.7e+124	1.7e+124
PSO	13(0)	19(1)	74(18)	148(25)	148(25)	15/15							

former under moderate budget settings followed by TPLPSO. Finally, iSRPSO can be termed as the better performing algorithm on the functions followed by TPLPSO under cheap budget settings.

4 Conclusion

This paper provides an extensive comparison of eight human learning principles inspired PSO variants on the noiseless BBOB testbed. Based on the results, SL-PSO and MePSO are performing better on expensive budget settings, MePSO and TPLPSO are performing better on moderate budget settings and iSRPSO and TPLPSO are performing better under cheap budget settings. Further, iSRPSO has shown the most promising performance with robustness. From the results, it can be inferred that MePSO and ELPSO are the most suited algorithms for separable functions, TPLPSO and SRPSO are most suited for low dimensional ill-conditioned and high dimensional ill-conditioned functions respectively. TPLPSO and iSRPSO are most suited for multi-modal and weakly structured multi-modal functions. Further, SRPSO and SL-PSO can also be chosen for multi-modal functions.

It has been observed that the top performing algorithm, iSRPSO has not been able to locate the optimum solutions on the separable functions. The algorithm can be further investigated for providing better solutions on the separable functions. Similarly, TPLPSO can be further investigated for performance enhancement on moderate and multi-modal functions. The performance of ELPSO and MePSO can be further enhanced by considering modifications in the algorithm to tackle with ill-conditioned and multi-modal functions.

Acknowledgement. The authors wish to extend their thanks to the ATMRI:2014-R8, Singapore, for providing financial support to conduct this study.

References

1. Arya, M., Deep, K., Bansal, J.C.: A nature inspired adaptive inertia weight in particle swarm optimisation. *Int. J. AI Soft Comput.* **4**(2–3), 228–248 (2014)
2. Cheng, R., Jin, Y.: A social learning particle swarm optimization algorithm for scalable optimization. *Inf. Sci.* **291**, 43–60 (2015)
3. Cheng, R., Jin, Y.: A competitive swarm optimizer for large scale optimization. *IEEE Trans. Cybern.* **45**(2), 191–204 (2015)
4. Epitropakis, M., Plagianakos, V., Vrahatis, M.: Evolving cognitive and social experience in particle swarm optimization through differential evolution: a hybrid approach. *Inf. Sci.* **216**(1), 50–92 (2012)
5. Eslami, M., Shareef, H., Khajehzadeh, M., Mohamed, A.: A survey of the state of the art in particle swarm optimization. *Res. J. Appl. Sci. Eng. Technol.* **4**(9), 1181–1197 (2012)
6. Finck, S., Hansen, N., Ros, R., Auger, A.: Real-parameter black-box optimization benchmarking 2009: presentation of the noiseless functions. Technical Report 2009/20, Research Center PPE (2009). Updated, February 2010

7. Hansen, N., Auger, A., Finck, S., Ros, R.: Real-parameter black-box optimization benchmarking 2012: experimental setup. Technical report, INRIA (2012)
8. Hansen, N., Finck, S., Ros, R., Auger, A.: Real-parameter black-box optimization benchmarking 2009: noiseless functions definitions. Technical report RR-6829, INRIA (2009). Updated February 2010
9. Huang, H., Qin, H., Hao, Z., Lim, A.: Example-based learning particle swarm optimization for continuous optimization. *Inf. Sci.* **182**(1), 125–138 (2012)
10. Kennedy, J., Eberhart, R.: Particle swarm optimization. In: *Proceedings of IEEE International Conference on Neural Networks*, pp. 1942–1948 (1995)
11. Liang, J., Qin, A., Suganthan, P., Baskar, S.: Comprehensive learning particle swarm optimizer for global optimization of multimodal functions. *IEEE Trans. Evol. Comput.* **10**(3), 281–295 (2006)
12. Lim, W., Isa, N.: Teaching and peer-learning particle swarm optimization. *Appl. Soft Comput.* **18**, 39–58 (2014)
13. Lynn, N., Suganthan, P.: Heterogeneous comprehensive learning particle swarm optimization with enhanced exploration and exploitation. *Swarm Evol. Comput.* **24**, 11–24 (2015)
14. Nelson, T., Narens, L.: Metamemory: a theoretical framework and new findings. *Psychol. Learn. Motiv.* **26**, 125–141 (1990)
15. Poli, R.: Analysis of the publications on the applications of particle swarm optimization. *Artif. Evol. Appl.* **28**, 1–10 (2008)
16. Price, K.: Differential evolution vs. the functions of the second ICEO. In: *Proceedings of the IEEE International CEC*, pp. 153–157 (1997)
17. Shi, Y.: Brain storm optimization algorithm. In: Tan, Y., Shi, Y., Chai, Y., Wang, G. (eds.) *ICSI 2011. LNCS*, vol. 6728, pp. 303–309. Springer, Heidelberg (2011). doi:[10.1007/978-3-642-21515-5_36](https://doi.org/10.1007/978-3-642-21515-5_36)
18. Sun, S., Li, J.: A two-swarm cooperative particle swarms optimization. *Swarm Evol. Comput.* **15**, 1–18 (2014)
19. Suresh, S., Sujit, P., Rao, A.: Particle swarm optimization approach for multi-objective composite box-beam design. *Compos. Struct.* **81**(4), 598–605 (2007)
20. Tanweer, M.R., Suresh, S., Sundararajan, N.: Human meta-cognition inspired collaborative search algorithm for optimization. In: *IEEE MFI*, pp. 1–6 (2014)
21. Tanweer, M.R., Suresh, S., Sundararajan, N.: Self regulating particle swarm optimization algorithm. *Inf. Sci.* **294**, 182–202 (2014)
22. Tanweer, M.R., Suresh, S., Sundararajan, N.: Dynamic mentoring and self-regulation based particle swarm optimization algorithm for solving complex real-world optimization problems. *Inf. Sci.* **326**, 1–24 (2015)
23. Tanweer, M.R., Suresh, S., Sundararajan, N.: Improved SRPSO algorithm for solving CEC 2015 computationally expensive numerical optimization problems. In: *IEEE CEC*, pp. 1943–1949 (2015)
24. Tanweer, M.R., Suresh, S., Sundararajan, N.: Mentoring based particle swarm optimization algorithm for faster convergence. In: *IEEE CEC*, pp. 196–203 (2015)
25. Wang, H., Qiao, Z., Xia, C., Li, L.: Self-regulating and self-evolving particle swarm optimizer. *Eng. Opt.* **47**(1), 129–147 (2015)
26. Wolpert, D.H., Macready, W.G.: No free lunch theorems for optimization. *IEEE Trans. Evol. Comput.* **1**(1), 67–82 (1997)
27. Zhang, Y., Wang, S., Ji, G.: A comprehensive survey on particle swarm optimization algorithm and its applications. *Math. Prob. Eng.* **501** (2015) 931256

Swarm, Evolutionary, and Memetic Computing
6th International Conference, SEMCCO 2015,
Hyderabad, India, December 18-19, 2015, Revised
Selected Papers

Panigrahi, B.K.; Suganthan, P.N.; Das, S.; Satapathy,
S.C. (Eds.)

2016, XI, 288 p. 117 illus., Softcover

ISBN: 978-3-319-48958-2