Biogeography-based learning particle swarm optimization

Chen, Xu; Tianfield, Huaglory; Mei, Congli; Du, Wenli ; Liu, Guohai

Published in:
Soft Computing

DOI:
10.1007/s00500-016-2307-7

Publication date:
2017

Document Version
Author accepted manuscript

Citation for published version (Harvard):
Chen, X, Tianfield, H, Mei, C, Du, W & Liu, G 2017, 'Biogeography-based learning particle swarm optimization', *Soft Computing*, vol. 21, pp. 7519-7541. https://doi.org/10.1007/s00500-016-2307-7

Biogeography-Based Learning Particle Swarm Optimization

Xu Chen [1], · Huaglory Tianfield [2], · Congli Mei [1], · Wenli Du [3] and· Guohai Liu [1]

1 School of Electrical and Information Engineering, Jiangsu University, Zhenjiang 212013, China
2 School of Engineering and Built Environment, Glasgow Caledonian University, Glasgow G4 0BA, UK
3 Key Laboratory of Advanced Control and Optimization for Chemical Processes, Ministry of Education, East China University of Science and Technology, Shanghai 200237, China

Correspondence: Xu Chen xuchen@ujs.edu.cn

Abstract This paper explores biogeography-based learning particle swarm optimization (BLPSO). Specifically, based on migration of biogeography-based optimization (BBO), a new biogeography-based learning strategy is proposed for particle swarm optimization (PSO), whereby each particle updates itself by using the combination of its own personal best position and personal best positions of all other particles through the BBO migration. The proposed BLPSO is thoroughly evaluated on 30 benchmark functions from CEC 2014. The results are very promising, as BLPSO outperforms five well-established PSO variants and several other representative evolutionary algorithms.

Keywords particle swarm optimization; biogeography- based learning; exemplar generation; biogeography-based optimization; migration

1 Introduction

Particle swarm optimization (PSO) is a population-based optimization algorithm, originally proposed by Eberchart and Kennedy (1995a). PSO is inspired by the social interaction and communication such as bird flocking, fish schooling, and swarm insects searching for food. Owing to its easy implementation, rapid convergence rate, and good performance, PSO has become a widely adopted optimization algorithm (Qin et al. 2015).

In canonical PSO, all particles keep learning from the personal best experience and the global best experience of the entire swarm, which may lead to premature convergence (Gong et al. 2015). To improve the performance, numerous PSO variants have been proposed, which can be generally divided into four categories that focus on:

- **Parameters control**, including inertia weight and acceleration coefficients (Hu et al. 2013; Nickabadi et al. 2011; Shi and Eberhart 2001), and population size (Chen and Zhao 2009; Sheng-Ta et al. 2009).
- **Population topology and multi-swarm techniques,** including fully connected, wheel and Von Neumann (Kennedy 1999; Mendes et al. 2004), unified topology (Parsopoulos and Vrahatis 2004), cellular structured topology (Fang et al. 2016), adaptive time-varying topology (Lim and Isa 2014a), and dynamic multi-swarm (Liang and Suganthan 2005).
- **Hybrid with other algorithms**, including genetic algorithm (Gong et al. 2015; Robinson et al. 2002), differential evolution (Epitropakis et al. 2012), harmony search (Ouyang et al. 2016), and teaching-learning-based optimization (Lim and Isa 2014b).
- **New learning strategies,** including comprehensive learning strategy (Liang et al. 2006), orthogonal learning strategy (Zhan et al. 2011), self-learning strategy (Li et al. 2012), genetic learning strategy (Gong et al. 2015), competitive learning strategy (Cheng and Jin 2015a), and social learning strategy(Cheng and Jin 2015b).

An important direction is to design new learning strategies for PSO. Liang et al. (2006) proposed comprehensive learning PSO (CLPSO). It uses a novel comprehensive learning strategy (CLS) whereby all other particles' personal best positions are used to update a particle's velocity. The CLS can preserve the diversity of the swarm to discourage premature.

In a parallel line, Simon (2008) proposed a new evolutionary algorithm (EA), named biogeography-based optimization (BBO), inspired from biogeographic evolution. BBO mainly uses the biogeography-based migration operator to share the information among individuals.

We find that there are some similarities between the productive operators of CLPSO and BBO. CLPSO uses personal best positions of many particles to construct the exemplars, while BBO uses a migration operator whereby many solutions contribute to producing an offspring. We believe that the migration of BBO can serve as a new exemplar generation method for CLPSO. Hence, our aim is to introduce the BBO migration into CLPSO, so as to propose a biogeography-based learning particle swarm optimization (BLPSO).

In this study, we will propose a biogeography-based learning strategy (BLS) whereby each particle updates itself by using the combination of its own personal best position and personal best position of other particles based on the BBO migration. Using BLS in place of CLS in CLPSO, a new PSO, i.e., biogeography-based learning particle swarm optimization (BLPSO), is proposed. The proposed BLPSO will be thoroughly evaluated on 30 benchmark functions from CEC 2014 on single-objective numerical optimization (Liang et al. 2013) and compared with previous representative PSO variants and some state-of-the-art EAs.

The remainder of the paper is arranged as follows: Section 2 introduces the CLPSO and BBO. Section 3 proposes the biogeography-based learning strategy and develops BLPSO algorithm. Section 4 presents thorough comparative simulation results. Lastly, Sect. 5 draws the conclusions.

2 Comprehensive learning particle swarm optimization and biogeography-based optimization

2.1 Comprehensive learning particle swarm optimization

PSO is swarm intelligence algorithm proposed by Eberhart and Kennedy (1995b) . In canonical PSO, each particle learns from its own personal best position (i.e., *pbest)* and the global best position found by the swarm (i.e., *gbest)* in order to update velocity and position. Let $v_i = (v_{i1}, v_{i2}, \cdots, v_{iD})$ and $x_i = (x_{i1}, x_{i2}, \cdots, x_{iD})$ represent the velocity and position of particle$_i$, respectively. Let $pbest_i = (pbesti_1, pbesti_2, \cdots, pbesti_D)$ denotes the personal best position of particle i and $gbest = (gbest_1, gbest_2, \cdots, gbest_D)$ denotes the global best position found by the swarm. The update equations for the d-th dimension of particle i are defined as follows:

$$v_{id} \leftarrow wv_{id}(t) + c_1 r_{1d}(pbest_{id} - x_{id}) + c_2 r_{2d}(gbest_d - x_{id}) \tag{1}$$

$$x_{id} \leftarrow x_{id} + v_{id} \tag{2}$$

where w is the inertia weight; c_1 and c_2 are the acceleration coefficients; r_{1d} and r_{2d} are two numbers randomly generated within [0, 1].

With the canonical learning scheme, PSO may easily get trapped in a local optimum when solving complex multimodal problems. To overcome the above shortcoming, Liang et al. (2006) proposed the CLPSO utilizing a new comprehensive learning strategy. The velocity updating equation in CLPSO can be described as follows:

$$v_{id} \leftarrow wv_{id} + cr_{id}(pbest_{f_i(d),d} - x_{id}) \tag{3}$$

where $f_i(d)$ means that, in the d-th dimension, particle i learns from the *pbest* of particle $f_i(d)$; $f_i = [f_i(1), f_i(2), \cdots, f_i(D)]$ defines the index vector of exemplars in all dimensions for particle i ; c is the acceleration coefficient.

From Eq. 3, it can be seen that each particle can learn from *pbest* of different particles for different dimensions in CLPSO. This raises a question: How to select exemplars in all dimensions for particle i ? In other words, how to generate the index vector of exemplars $f_i = [f_i(1), f_i(2), \cdots, f_i(D)]$ for particle i ?

The original CLPSO firstly assigns a learning probability Pci for each particle i using the equation below:

$$Pc_i = a + b * \frac{\exp\left(\frac{10 \cdot (i-1)}{N-1}\right) - 1}{\exp(10) - 1} \tag{4}$$

where a and b are two parameters to determine the maximum and minimum learning probability; N is the population size.

Then, the exemplar generation method for particle i is presented in Algorithm 1. Moreover, to ensure that a particle learns from good exemplars and to minimize the time wasted on poor directions, a refreshing gap number m is defined for evaluation, and a new $f_i = [f_i(1), f_i(2), \cdots, f_i(D)]$ will be generated if there is no improvement for m consecutive moves.

Since the inception of CLPSO, several improved CLPSO variants have been proposed, including enhanced version (ECLPSO) (Yu and Zhang 2014), parallel version (PCLPSO) (Gulcu and Kodaz 2015), heterogeneous version (HCLPSO) (Lynn and Suganthan 2015), and multi-objective version (MOCLPSO) (Huang et al. 2006). However, almost all these CLPSO variants use the similar exemplar generation method as that in the original CLPSO.

Algorithm 1. Exemplar generation method for particle i in CLPSO

Input: Fitness of personal best positions *fit*(*pbesti*) , learning probability Pc_i (i = 1, , N)
Output: Exemplar vector index $f_i = [\, f_i(1), f_i(2), \cdots , f(D)]$
for d = 1 **to** D **do**
    Randomly generate a real number *rand* within [0,1]
    if $rand \geq Pc_i$ **then** // learn from its own *pbest*
        $f_i(d) \leftarrow i$;
    else // learn from *pbest* of other particle by a tournament selection
        Randomly select two particle a and b ($a \neq b \neq i$)
        If *fit*(*pbesta*) < *fit*(*pbestb*)
            $f_i(d) \leftarrow a$;
        else
            $f_i(d) \leftarrow b$;
        end if
    end if
end for
if $f_i(d) == i$ (d = 1, , D) in all dimensions

    Randomly select a particle j($j \neq i$) ;
    Randomly select a dimension l ;
        $f_i(l) \leftarrow j$;
end if

2.2 Biogeography-based optimization

BBO is biogeography-inspired evolutionary algorithm proposed by Simon (2008). In BBO, each individual is considered as a "island" with a habitat suitability index (HSI) to measure the individual, and the individual components are analogous to a set of suitability index variables (SIV). A good individual is analogous to an island with a high HSI, and a poor individual indicates an island with a low HSI. BBO uses two main operators, migration and mutation, to modify the individuals, but the core operator is migration.

Assume that there are N individuals $x_i = (x_{i1}, x_{i2}, \cdots , x_{iD})$, $i = 1, \cdots , N$. In the BBO migration, each individual has its own immigration rate λ and emigration rate μ. A

good individual has a higher emigration rate μ and a lower immigration rate λ, and vice versa, a bad individual has a lower μ and a higher λ. The immigration and the emigration rates are functions of the ranking value of the individuals. They can be calculated as follows:

$$\lambda_k = \left(1 - \frac{k}{N}\right) \cdot I \tag{5}$$

$$\mu_k = \left(\frac{k}{N}\right) \cdot E \tag{6}$$

where I and E are the maximum possible immigration and emigration rates, commonly set as $I = E = 1$; N is the population size; k is the index of the individual with rank k, where $k = 1$ refers to the worst individual and $k = N$ refers to the best individual.

Equations 5 and 6 are the linear migration model proposed in the original BBO. Figure 1 plots the immigration and emigration curves. Besides the linear migration model, Ma (2010) gave a total of six migration models for BBO, as presented in "Appendix 1."

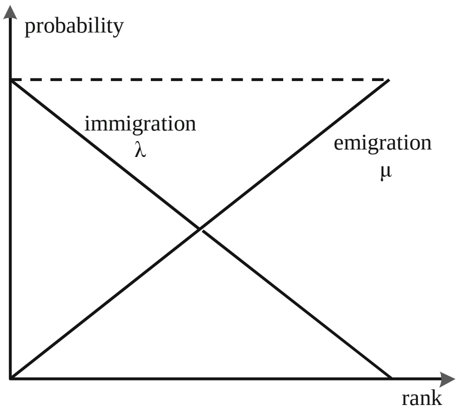

Figure 1 The immigration and emigration curves

The BBO migration for individual *xi* can be presented in Algorithm 2. From Algorithm 2, it can be seen that a good individual has a low immigration rate λ, and thus, the solution modifies its components with low probability; a poor individual has a high immigration rate λ, and thus, the solution modifies its components with high probability. At the same time, a good individual has a high emigration rate μ, so that other solutions can obtain more features from that good individual; a poor individual has a low emigration rate μ, so that other solutions obtain few features from the poor individual. This migration makes BBO be good at exploiting the information of the current population.

Further details about BBO can be found in (Gong et al. 2010b; Simon 2008) and on webpage of BBO.[1]

Algorithm 2. BBO migration for individual x_i

Input: Individual $x_i = (x_{i1}, x_{i2}, \cdots, x_{iD})$, immigration rate λ_i , and emigration rate μ_i
Output: Modified individual $\overline{x}_i = (\overline{x}_{i1}, \overline{x}_{i2}, \cdots, \overline{x}_{iD})$
for $d = 1$ **to** D **do**
  Randomly generate a number *rand* within [0,1]
  if $rand < \lambda_i$ **then** // whether to immigrate?
    Select an individual x_j with probability $\propto \mu_j$ // which individual to emigrate?
    $\overline{x}_{id} \leftarrow x_{jd}$;
  else
    $\overline{x}_{id} \leftarrow x_{id}$;
  end if
end for

3 Proposed biogeography-based learning particle swarm optimization

3.1 Biogeography-based learning strategy

The core idea of biogeography-based learning strategy**(**BLS) is to generate the exemplar vector index $f_i = [f_i(1), f_i(2), \cdots, f_i(D)]$ based on the BBO migration. The steps of the exemplar generation in BLS can be described as follows:

First, all particles are ranked based on the quality of their *pbest* . For minimization problem, assume

$$fit(pbest_{s_1}) \le fit(pbest_{s_2}) \le \cdots \le fit(pbest_{s_N}) \tag{7}$$

where s_1 is the subscript of the particle with the best *pbest* , s_2 is the subscript of the particle with the second best *pbest* , and s_N is the subscript of the particle with the worst *pbest* .

Then, the rankings of particles are assigned as below:

$$rank(s_1) = N-1, rank(s_2) = N-2, \cdots, rank(s_N) = 0 \tag{8}$$

According to Eq. 8, the particle with the best *pbest* obtains the highest ranking value, and the particle with the worst

[1] webpage of BBO http://embeddedlab.csuohio.edu/BBO/.

pbest obtains the lowest ranking value.

Second, immigration and emigration rates are assigned for all particles. Here, we use the linear migration model as an example. The immigration and emigration rates for all particles can be calculated as follows:

$$\begin{cases}\lambda(s_1) = 1 - \frac{N-1}{N} \\ \mu(s_1) = \frac{N-1}{N}\end{cases}, \begin{cases}\lambda(s_2) = 1 - \frac{N-2}{N} \\ \mu(s_2) = \frac{N-2}{N}\end{cases}, \dots, \begin{cases}\lambda(s_N) = 1 - \frac{0}{N} \\ \mu(s_N) = \frac{0}{N}\end{cases} \tag{9}$$

According to Eq. 9, the solution x_{s1} with the best $pbest_{s1}$ will have the lowest immigration rate $\lambda(s_1)$ and highest emigration rate $\mu(s_1)$; the solution x_{sN} with the worst $pbest_{sN}$ will have the highest immigration rate $\lambda(s_N)$ and lowest emigration rate $\mu(s_N)$.

Third, the biogeography-based exemplar generation method for particle *i* can be generated as in Algorithm 2. From Algorithm 2, it can be seen that:

(1) particle x_{s1} with the best $pbest_{s1}$ has the lowest immigration rate $\lambda(s_1)$, so it will learn more from its own $pbest_{s1}$;
(2) particle x_{sN} with the worst $pbest_{sN}$ has the highest immigration rate $\lambda(s_N)$, so it will learn more from others' *pbest*;
(3) particle *xs*1 with the best $pbest_{s1}$ has the highest emigration rate, so its $pbest_{s1}$ will tend more to be learned by other particles; and
(4) particle *xsN* with the worst $pbest_{sN}$ has the lowest emigration rate, so its $pbest_{sN}$ will tend less to be learned by other particles.

3.2 Biogeography-based learning particle swarm optimization (BLPSO) algorithm

Using BLS in place of CLS in the original CLPSO, a biogeography-based learning particle swarm optimization (BLPSO) can be proposed as in Algorithm 4.

Algorithm 3. Biogeography-based exemplar generation method for particle *i*

Input: Rank value $rank(i)$, immigration rate $\lambda_{rank(i)}$, and emigration rate $\mu_{rank(i)}$
Output: Exemplar vector index $f_i = [f_i(1), f_i(2), \cdots, f_i(D)]$
for $k = 1$ **to** D **do**
    if $rand < \lambda_{rank(i)}$ // learn from other particle
        Utilize a roulette wheel to select a particle index j with probability $\propto \mu_{rank(j)}$;
        $f_i(k) \leftarrow j$;
    else // learn from itself
        $f_i(k) \leftarrow i$;
    end If
end for
if $f_i(k) == i$ $(i = 1, \cdots, D)$
    Randomly select a particle index $j (j \neq i)$;
    Randomly select a dimension l;
    $f_i(l) \leftarrow j$;
end if

Algorithm 4. BLPSO

for each particle $i \in \{1, 2, \cdots, ps\}$ // initialize the population
    Randomly initialize the position x_i and velocity v_i;
    Evaluate the position $fit(x_i)$;
    Store the personal best position $pbest_i$;
end for
while the halting criterion is not satisfied **do** // main loop of BLPSO
    for each particle $i \in \{1, 2, \cdots, ps\}$ **do**
        Assign the ranking values for all particles based on the fitness of their *pbest*;
        Assign the immigration and emigration rates for all particles;
        Generate the exemplar vector index $f_i = [f_i(1), f_i(2), \cdots, f_i(D)]$ using Algorithm 3;

Update the velocity v_i according to Eq.(3);
Update the position x_i according to Eq.(2);
Evaluate the new position $fit(x_i)$;
Update the personal best position $pbest_i$;
end for
end while
Output the best solution.

The main differences between the proposed BLPSO and the original CLPSO can be summarized as follows:

(1) The original CLPSO uses learning probability and tournament selection to generate exemplars for particles, whereas BLPSO uses the migration of BBO to generate exemplars.
(2) Given a specific problem, there are two difficult-to-tune parameters in the original CLPSO, i.e., the learning probability Pc_i and the refreshing gap*m*. On the contrary, there is no parameter to tune in BLPSO, only except that the maximum possible immigration and emigration rates can be trivially set as $I = E = 1$. What needs to be done is select an appropriate migration model from the pool of migration models as listed in "Appendix 1."
(3) The adaptation of the original CLPSO is limited, as the learning probability Pc_i is usually assigned a value just based on the particle index i , and the Pc value usually remains unchanged during the whole process of optimization. Differently, BLPSO ranks the particles based on their *pbest* and assigns new immigration and emigration rates in each generation, and thus, BLPSO may have a stronger adaption ability.

It is also worth pointing out that the structure of BLPSO is as simple as CLPSO. Compared with other state-of-the-art CLPSO variants such as ECLPSO (Yu and Zhang 2014), PCLPSO (Gulcu and Kodaz 2015), and HCLPSO (Lynn and Suganthan 2015), BLPSO is relatively easy to realize.

Remark 1 The core idea of BLPSO is that the "ranking" technique and "migration" technique are used simultaneously so that the particles can learn from different personal best positions. Recently, Cheng and Jin (2015b) proposed a social learning PSO (SL-PSO) in which 'swarm sorting' and 'learning probability' are utilized. However, there are important differences between BLPSO and SL-PSO. Firstly, there are no historical personal best positions in SL-PSO, so the "swarm sorting" is conducted on the current swarm, whereas in BLPSO the ranking technique is conducted on personal best positions. Secondly, the particles in SL-PSO have the "learning probability," which is similar to the "immigration rate" in BLPSO, whereas BLPSO has an "emigration rate," i.e., the probability learned by the others. Thirdly, SLPSO uses the three-term formula for behavior correction, while BLPSO uses the two-term formula for velocity updating. Finally, the learning mechanism of SL-PSO is inspired from a social phenomenon called imitation, while the learning mechanism of BLPSO is inspired from biogeographic migration.

Remark 2 Gong et al. (2015) proposed a generalized learning paradigm for PSO, called the "*L-PSO." The paradigm is composed of two cascading layers, the first for exemplar generation and the second for particle updates. As the "*L-PSO" paradigm involves the evaluation procedure to calculate the fitness of the constructed exemplars, our BLPSO does not belong to the "*L-PSO" paradigm.

4 Numeric simulations

We employ the 30 benchmark functions from CEC 2014 on single-objective numerical optimization (Liang et al. 2013) to evaluate the performance of our proposed BLPSO. These benchmark functions fall into four groups: (1) unimodal functions (F1 - F3); (2) simple multimodal functions (F4 - F16); (3) hybrid functions (F17 - F22); and (4) composition function (F23 - F30).

The proposed BLPSO is tested on both 30-*D* and 50*D* functions. The maximal number of function evaluations (*Max N FEs)* is set as 1000*D*. All numeric simulation results are obtained based on 30 independent runs.

Two performance criteria as below are used for comparing the performance of each algorithm.

(1) **Error** (Suganthan et al. 2005): The mean and standard deviation of the error value $f(x) - f(x^*)$ are recorded, where x^* is the global optimum of the test function and x is the best solution found by the algorithm in a single run.
(2) **Statistics by the Wilcoxon and the Friedman tests**: The Wilcoxon rank sum test at 5 % significance level is conducted to show the significant differences between two algorithms on the same problem. The Friedman test conducted by the KEEL software (Alcala-Fdez et al. 2009) is used to obtain the rankings of the algorithms on all the benchmark functions.

4.1 Comparisons over different migration models

There are no parameters to tune in BLPSO, but an appropriate migration model has to be selected for BLPSO on the test functions. Hence, first of all, we compare the performance of BLPSO over six different migration models as presented in Appendix 1. The six BLPSO are denoted as BLPSO-*i* , *i* = 1, 2, 3, 4, 5, and 6, i.e., with the *i* -th migration model.
Table 1 shows the parameter settings for the six BLPSO. Table 2 shows the errors of the six BLPSO on the 30-*D* functions. In general, BLPSO-5 demonstrates the best performance among the six BLPSO, as it attains its best results on 12 out of the 30 test functions, including 1 unimodal function, 6 simple multimodal functions, 2 hybrid functions, and 3 composition functions. BLPSO-1, BLPSO-2, BLPSO-3, BLPSO-4, and BLPSO-6 attain their best results on 5, 1, 2, 4, and 6 functions, respectively.

Table 1 Parameter settings for the six BLPSO

Parameter	Value
Population size N	40
Inertia weight w	0.9 ~0.2, linearly decrease
Acceleration coefficients c	1.496
Maximum immigration and rates I	1
Maximum emigration rates E	1

Table 2 Errors of the six BLPSO on the 30-*D* functions

		BLPSO-1	BLPSO-2	BLPSO-3	BLPSO-4	BLPSO-5	BLPSO-6
F1	Mean	**8.91E+05**	4.11E+06	3.52E+06	1.18E+06	2.99E+06	3.21E+06
	SD	**3.84E+05**	1.62E+06	1.09E+06	9.06E+05	1.10E+06	1.38E+06
F2	Mean	7.50E+03	8.39E+03	6.17E+03	7.57E+03	**5.09E+03**	5.75E+03
	SD	4.46E+03	4.27E+03	4.23E+03	6.36E+03	**4.25E+03**	4.41E+03
F3	Mean	1.90E+02	4.73E+01	1.59E+01	5.52E+01	3.67E+00	**2.12E+00**
	SD	1.31E+02	4.95E+01	3.06E+01	7.29E+01	1.16E+01	**4.38E+00**
F4	Mean	**1.43E+00**	2.72E+01	1.07E+01	3.32E+00	2.68E+01	3.11E+01
	SD	**3.03E−01**	3.13E+01	2.36E+01	1.24E+01	3.47E+01	4.04E+01
F5	Mean	2.10E+01	2.09E+01	2.08E+01	2.09E+01	**2.08E+01**	2.08E+01
	SD	3.84E−02	8.02E−02	6.50E−02	5.53E−02	**7.01E−02**	7.22E−02
F6	Mean	3.28E−02	2.62E−07	**1.17E−07**	6.57E−02	9.37E−06	1.18E−06
	SD	1.80E−01	1.44E−06	**6.42E−07**	2.50E−01	3.20E−05	5.27E−06
F7	Mean	1.10E−13	1.21E−13	9.85E−14	**6.44E−14**	9.47E−14	2.47E−04
	SD	3.64E−14	4.15E−14	3.93E−14	**5.73E−14**	4.31E−14	1.35E−03
F8	Mean	3.12E+00	2.67E+00	1.54E+00	3.58E+00	**2.32E−01**	3.32E−01
	SD	1.99E+00	2.59E+00	1.02E+00	1.60E+00	**5.65E−01**	6.03E−01
F9	Mean	1.54E+02	4.49E+01	3.98E+01	5.67E+01	3.54E+01	**3.31E+01**
	SD	7.33E+00	7.09E+00	8.78E+00	1.25E+01	6.93E+00	**6.59E+00**
F10	Mean	1.39E+02	1.66E+02	1.07E+02	**8.36E+01**	8.83E+01	1.00E+02
	SD	1.02E+02	1.16E+02	1.06E+02	**9.52E+01**	6.48E+01	9.47E+01
F11	Mean	6.12E+03	2.73E+03	2.61E+03	3.38E+03	**2.08E+03**	2.29E+03
	SD	3.12E+02	5.23E+02	4.55E+02	4.10E+02	**3.82E+02**	4.00E+02
F12	Mean	2.41E+00	1.25E+00	1.32E+00	1.73E+00	**8.83E−01**	8.90E−01
	SD	2.51E−01	2.67E−01	2.86E−01	3.59E−01	**1.49E−01**	1.63E−01
F13	Mean	2.36E−01	2.32E−01	2.14E−01	**2.08E−01**	2.21E−01	2.11E−01
	SD	3.96E−02	4.42E−02	3.90E−02	**4.29E−02**	2.85E−02	3.52E−02
F14	Mean	2.45E−01	2.29E−01	2.28E−01	2.19E−01	**2.14E−01**	2.24E−01
	SD	3.09E−02	2.62E−02	3.20E−02	3.39E−02	**2.88E−02**	2.38E−02
F15	Mean	1.50E+01	9.12E+00	8.93E+00	1.06E+01	**7.41E+00**	7.43E+00
	SD	8.57E−01	1.18E+00	1.65E+00	1.60E+00	**8.49E−01**	1.03E+00
F16	Mean	1.10E+01	9.98E+00	9.46E+00	9.85E+00	9.67E+00	**9.43E+00**
	SD	3.85E−01	5.15E−01	8.07E−01	6.46E−01	4.92E−01	**5.50E−01**
F17	Mean	4.35E+05	2.83E+05	2.37E+05	3.26E+05	**1.86E+05**	2.48E+05
	SD	2.26E+05	1.98E+05	8.98E+04	1.30E+05	**1.11E+05**	1.42E+05
F18	Mean	6.14E+02	**3.63E+02**	6.83E+02	7.56E+02	9.05E+02	9.10E+02
	SD	6.85E+02	**4.89E+02**	8.48E+02	1.06E+03	1.20E+03	1.24E+03
F19	Mean	4.55E+00	3.86E+00	**3.60E+00**	3.67E+00	3.74E+00	3.67E+00

	SD	5.30E−01	4.93E−01	**6.32E−01**	5.86E−01	6.12E−01	6.27E−01
F20	Mean	6.06E+02	5.58E+02	2.77E+02	2.53E+02	3.12E+02	**1.77E+02**
	SD	2.88E+02	3.26E+02	2.06E+02	7.50E+01	3.48E+02	**8.05E+01**
F21	Mean	1.45E+05	4.04E+04	5.73E+04	5.86E+04	3.85E+04	**3.79E+04**
	SD	5.85E+04	2.60E+04	3.93E+04	5.45E+04	3.19E+04	**3.01E+04**
F22	Mean	2.08E+02	1.20E+02	1.34E+02	1.46E+02	**1.16E+02**	1.42E+02
	SD	4.41E+01	4.86E+01	7.67E+01	3.95E+01	**6.86E+01**	6.11E+01
F23	Mean	3.15E+02	3.15E+02	3.15E+02	3.15E+02	**3.15E+02**	3.15E+02
	SD	5.24E−12	1.39E−11	2.69E−12	8.64E−12	**6.11E−13**	1.77E−12
F24	Mean	**2.14E+02**	2.21E+02	2.20E+02	2.20E+02	2.22E+02	2.22E+02
	SD	**1.02E+01**	3.94E+00	6.65E+00	5.46E+00	7.39E−01	3.07E+00
F25	Mean	2.05E+02	2.05E+02	2.04E+02	**2.04E+02**	2.05E+02	2.05E+02
	SD	6.05E−01	5.94E−01	5.68E−01	**6.45E−01**	4.41E−01	5.39E−01
F26	Mean	**1.00E+02**	1.04E+02	1.07E+02	1.04E+02	1.04E+02	1.04E+02
	SD	**1.25E+00**	1.82E+01	2.54E+01	1.82E+01	1.82E+01	1.83E+01
F27	Mean	3.03E+02	3.00E+02	3.07E+02	3.07E+02	3.08E+02	**3.00E+02**
	SD	1.87E+01	4.22E−02	2.59E+01	2.59E+01	2.94E+01	**8.82E−03**
F28	Mean	8.12E+02	7.94E+02	8.08E+02	8.15E+02	**7.87E+02**	7.99E+02
	SD	4.62E+01	4.61E+01	3.62E+01	4.40E+01	**5.22E+01**	4.51E+01
F29	Mean	**1.23E+03**	1.52E+03	1.49E+03	1.31E+03	1.39E+03	1.39E+03
	SD	**1.54E+02**	1.32E+02	1.36E+02	1.96E+02	1.39E+02	1.27E+02
F30	Mean	1.40E+03	1.47E+03	1.37E+03	1.34E+03	**1.19E+03**	1.40E+03
	SD	3.36E+02	3.34E+02	3.14E+02	2.74E+02	**2.49E+02**	2.78E+02

The best mean error values are marked in bold

Table 3 shows the ranking of the six BLPSO according to the Friedman test on the 30-*D* functions. BLPSO-5 attains the best rank, BLPSO-6 the second, followed by BLPSO-3, BLPSO-4, BLPSO-2, and BLPSO-1.

Table 3 Ranking of the six BLPSO according to the Friedman test on the 30-*D* functions

	BLPSO-1	BLPSO-2	BLPSO-3	BLPSO-4	BLPSO-5	BLPSO-6
Friedman rank	4.45	4.05	3.17	3.68	2.68	2.97
Final rank	6	5	3	4	1	2

Table 4 shows the errors of the six BLPSO on the 50-*D* functions. BLPSO-5 also demonstrates the best performance among the six BLPSO, as it attains its best results on 13 out of the 30 functions, including 7 simple multimodal functions, 2 hybrid functions, and 4 composition functions. BLPSO-1, BLPSO-2, BLPSO-3, BLPSO-4, and BLPSO-6 attain their best results on 1, 3, 3, 4, and 6 functions, respectively.

Table 5 shows the ranking of the six BLPSO according to the Friedman test on the 50-*D* functions. BLPSO-5 attains the best rank, BLPSO-6 the second, followed by BLPSO-3, BLPSO-4, BLPSO-2, and BLPSO-1.

Overall, BLPSO-5 demonstrates the best performance on both 30-*D*and 50-*D*benchmark functions. Therefore, BLPSO-5 is to be used in the following comparisons with the other algorithms.

Table 4 Error values of the six BLPSO on the 50-*D* functions

		BLPSO-1	BLPSO-2	BLPSO-3	BLPSO-4	BLPSO-5	BLPSO-6
F1	Mean	**1.01E+06**	7.22E+06	6.31E+06	2.66E+06	5.10E+06	5.38E+06
	SD	**2.46E+05**	2.41E+06	1.85E+06	2.54E+06	1.28E+06	1.63E+06
F2	Mean	3.49E+03	**2.57E+03**	3.25E+03	3.32E+03	3.44E+03	3.74E+03
	SD	3.08E+03	**2.52E+03**	3.04E+03	2.46E+03	2.35E+03	2.65E+03
F3	Mean	7.41E+02	5.61E+02	9.70E+01	1.70E+02	4.23E+01	**3.21E+01**
	SD	2.70E+02	2.75E+02	7.00E+01	1.27E+02	8.97E+01	**3.01E+01**
F4	Mean	8.99E+01	9.00E+01	7.22E+01	7.87E+01	8.64E+01	**6.06E+01**
	SD	5.21E+00	5.28E+00	2.92E+01	2.28E+01	5.04E+00	**3.17E+01**
F5	Mean	2.11E+01	2.10E+01	2.10E+01	2.11E+01	**2.09E+01**	2.09E+01
	SD	3.89E−02	6.25E−02	6.17E−02	4.22E−02	**5.07E−02**	4.62E−02
F6	Mean	5.02E−02	1.08E−01	8.79E−02	**1.28E−05**	9.22E−02	8.04E−02
	SD	2.74E−01	3.51E−01	3.09E−01	**2.68E−05**	3.21E−01	3.83E−01
F7	Mean	4.74E−13	5.42E−13	3.49E−13	3.87E−13	**2.92E−13**	2.47E−04

	SD	1.30E−13	7.72E−14	7.86E−14	1.02E−13	**6.46E−14**	1.35E−03
F8	Mean	9.75E+00	1.32E+01	5.49E+00	9.75E+00	**4.97E−01**	1.16E+00
	SD	2.81E+00	5.49E+00	3.48E+00	3.22E+00	**8.16E−01**	1.06E+00
F9	Mean	3.12E+02	1.05E+02	1.00E+02	1.50E+02	**7.10E+01**	7.52E+01
	SD	1.29E+01	2.14E+01	1.78E+01	2.46E+01	**9.02E+00**	1.03E+01
F10	Mean	3.88E+02	4.27E+02	**3.45E+02**	3.61E+02	3.63E+02	4.31E+02
	SD	1.72E+02	2.09E+02	**1.75E+02**	1.95E+02	1.81E+02	2.21E+02
F11	Mean	1.21E+04	6.39E+03	6.18E+03	7.69E+03	**4.46E+03**	4.65E+03
	SD	4.56E+02	7.88E+02	8.13E+02	7.23E+02	**4.77E+02**	4.67E+02
F12	Mean	3.31E+00	1.61E+00	1.55E+00	2.21E+00	**8.77E−01**	9.03E−01
	SD	3.55E−01	2.36E−01	3.62E−01	3.67E−01	**1.18E−01**	1.47E−01
F13	Mean	3.14E−01	3.13E−01	2.97E−01	**2.83E−01**	2.86E−01	2.88E−01
	SD	3.89E−02	3.57E−02	3.79E−02	**3.62E−02**	3.72E−02	3.30E−02
F14	Mean	3.07E−01	2.73E−01	2.95E−01	2.77E−01	2.65E−01	**2.62E−01**
	SD	1.40E−01	3.48E−02	1.24E−01	8.72E−02	2.42E−02	**2.40E−02**
F15	Mean	3.00E+01	1.93E+01	1.87E+01	2.33E+01	**1.48E+01**	1.55E+01
	SD	1.65E+00	2.32E+00	2.23E+00	3.12E+00	**1.33E+00**	1.74E+00
F16	Mean	2.04E+01	1.89E+01	1.87E+01	1.92E+01	1.82E+01	**1.81E+01**
	SD	3.77E−01	7.41E−01	5.45E−01	5.64E−01	4.73E−01	**6.34E−01**
F17	Mean	9.19E+05	6.69E+05	6.85E+05	6.90E+05	**5.97E+05**	6.34E+05
	SD	6.21E+05	2.87E+05	3.69E+05	3.52E+05	**2.10E+05**	3.09E+05
F18	Mean	2.32E+02	**1.36E+02**	1.67E+02	1.92E+02	3.73E+02	4.41E+02
	SD	1.98E+02	**1.75E+02**	1.71E+02	2.05E+02	3.68E+02	4.44E+02
F19	Mean	1.41E+01	2.00E+01	1.93E+01	**1.22E+01**	2.16E+01	1.96E+01
	SD	3.45E+00	8.74E+00	9.13E+00	**1.03E+00**	9.78E+00	8.73E+00
F20	Mean	5.54E+02	4.87E+02	2.83E+02	3.02E+02	2.57E+02	**2.19E+02**
	SD	1.02E+02	1.51E+02	9.59E+01	5.18E+01	1.41E+02	**9.69E+01**
F21	Mean	7.87E+05	6.43E+05	4.56E+05	4.98E+05	**3.80E+05**	4.56E+05
	SD	3.27E+05	2.87E+05	2.41E+05	2.97E+05	**1.44E+05**	2.28E+05
F22	Mean	8.02E+02	2.09E+02	**2.04E+02**	2.39E+02	2.64E+02	2.27E+02
	SD	2.43E+02	1.21E+02	**1.09E+02**	1.63E+02	1.30E+02	1.06E+02
F23	Mean	3.44E+02	3.44E+02	**3.44E+02**	3.44E+02	3.44E+02	3.44E+02
	SD	2.23E−13	2.67E−13	**1.93E−13**	2.60E−13	2.56E−13	2.63E−13
F24	Mean	2.59E+02	2.58E+02	2.60E+02	2.59E+02	2.58E+02	**2.57E+02**
	SD	4.64E+00	3.54E+00	4.70E+00	4.17E+00	4.07E+00	**3.49E+00**
F25	Mean	2.10E+02	2.11E+02	2.09E+02	**2.09E+02**	2.10E+02	2.11E+02
	SD	1.23E+00	1.10E+00	1.16E+00	**9.90E−01**	7.36E−01	1.11E+00
F26	Mean	1.70E+02	1.61E+02	1.67E+02	1.71E+02	**1.47E+02**	1.70E+02
	SD	4.36E+01	5.00E+01	4.81E+01	4.68E+01	**5.08E+01**	4.66E+01
F27	Mean	3.25E+02	**3.15E+02**	3.27E+02	3.21E+02	3.24E+02	3.27E+02
	SD	2.65E+01	**2.40E+01**	3.00E+01	2.72E+01	2.85E+01	3.24E+01
F28	Mean	1.15E+03	1.15E+03	1.14E+03	1.14E+03	**1.14E+03**	1.17E+03
	SD	6.49E+01	1.06E+02	4.89E+01	4.13E+01	**4.20E+01**	5.50E+01
F29	Mean	1.78E+03	2.11E+03	1.87E+03	1.71E+03	**1.36E+03**	1.48E+03
	SD	2.72E+02	3.84E+02	2.87E+02	2.96E+02	**1.82E+02**	2.33E+02
F30	Mean	9.44E+03	9.64E+03	9.64E+03	9.57E+03	**9.09E+03**	9.44E+03
	SD	4.47E+02	4.14E+02	4.14E+02	4.61E+02	**3.05E+02**	3.54E+02

The best mean error values are marked in bold

Table 5 Ranking of the six BLPSO according to the Friedman test on the 50-*D* functions

	BLPSO-1	BLPSO-2	BLPSO-3	BLPSO-4	BLPSO-5	BLPSO-6
Friedman rank	4.65	3.88	3.15	3.52	2.63	3.17
Final rank	6	5	3	4	1	2

4.2 Comparison with the other PSO variants

In order to study the performance of BLPSO against PSO variants, we compare BLPSO-5 with five well-established

PSO variants:[2]

- Linearly decreasing inertia weight PSO (LDWPSO) (Shi and Eberhart 1998)
- Dynamic multi-swarm PSO (DMSPSO) (Liang and Suganthan 2005)
- Fully informed PSO (FIPS) (Mendes et al. 2004)
- Social learning PSO (SL-PSO) (Cheng and Jin 2015b)
- Comprehensive learning PSO (CLPSO) (Liang et al. 2006)

Table 6 shows the parameter settings for the other PSO variants.

Table 7 shows the errors of BLPSO-5 and the other PSO variants on the 30-*D* functions. BLPSO-5, LDWPSO, DMSPSO, FIPS, SL-PSO, and CLPSO attain their best results on 12, 1, 4, 1, 6, and 6 functions, respectively. It can also be observed that BLPSO-5 performs particularly well on hybrid functions and composition functions. The three rows at the bottom section present the results of the Wilcoxon rank sum test between BLPSO-5 and the other PSO variants. BLPSO-5 performs significantly better than LDWPSO, DMSPSO, FIPS, SL-PSO, and CLPSO on 26, 18, 28, 22, and 21 functions, respectively, though it is significantly worse than LDWPSO, DMSPSO, FIPS, SL-PSO, and CLPSO on 2, 8, 1, 6, and 8 functions, while similar to them on 2, 3, 1, 2, and 1 functions, respectively.

Table 6 Parameter settings for the other PSO variants

Algorithm	Parameter settings
LDWPSO	Population size N = 40, inertia weight w linearly decreasing from 0.9 to 0.4, acceleration coefficients $c_1 = c_2 = 2$, Global topology
DMSPSO	Population size N = 40, inertia weight w = 0.729, acceleration coefficients $c_1 = c_2$ = 1.496, population size of sub-swarm m = 5, regrouping period R = 5
FIPS	Population size N = 40, constriction coefficient χ = 0.729, sum of acceleration coefficients ϕ = 4.1, URing topology
SL-PSO	Population size $N = 100 + floor(D/10)$, social influence factor $\varepsilon = 0.0001D$
CLPSO	Population size N = 40, inertia weight w linearly decreasing from 0.9 to 0.2, acceleration coefficients c = 1.496, refreshing gap m = 5, parameters for calculating learning probability a = 0, b = 0.5

Table 7 Error results of BLPSO-5 and the other PSO variants on the 30-*D* functions

		LDWPSO		DMSPSO		FIPSO		SL-PSO		CLPSO		BLPSO-5
F1	Mean	3.41E+06	=	**2.42E+05**	−	1.01E+07	+	3.81E+05	−	8.74E+06	+	2.99E+06
	SD	3.69E+06		**1.71E+05**		3.81E+06		3.38E+05		3.09E+06		1.10E+06
F2	Mean	**3.02E+01**	−	1.23E+02	−	3.08E+03	=	1.06E+04	+	2.18E+02	−	5.09E+03
	SD	**5.14E+01**		3.32E+02		2.97E+03		9.46E+03		9.42E+02		4.25E+03
F3	Mean	9.31E+01	+	1.25E+02	+	1.85E+03	+	5.74E+03	+	1.29E+02	+	**3.67E+00**
	SD	1.04E+02		1.49E+02		9.64E+02		5.42E+03		1.36E+02		**1.16E+01**
F4	Mean	1.39E+02	+	**1.67E+01**	−	1.99E+02	+	4.16E+01	+	6.06E+01	+	2.68E+01
	SD	3.72E+01		**2.99E+01**		2.36E+01		2.95E+01		2.35E+01		3.47E+01
F5	Mean	2.09E+01	+	2.04E+01	−	2.09E+01	+	2.10E+01	+	**2.03E+01**	−	2.08E+01
	SD	8.97E−02		8.57E−02		5.66E−02		4.56E−02		**3.55E−02**		7.01E−02
F6	Mean	1.08E+01	+	5.51E+00	+	6.25E+00	+	1.05E+00	+	1.34E+01	+	**9.37E−06**
	SD	2.50E+00		2.80E+00		2.56E+00		1.45E+00		1.50E+00		**3.20E−05**
F7	Mean	1.50E−02	+	8.69E−03	+	1.25E−04	+	1.31E−03	+	3.47E−05	+	**9.47E−14**
	SD	1.33E−02		1.21E−02		6.15E−04		3.47E−03		3.94E−05		**4.31E−14**
F8	Mean	1.93E+01	+	4.33E+01	+	4.97E+01	+	1.65E+01	+	**1.14E−13**	−	2.32E−01
	SD	4.58E+00		1.22E+01		1.04E+01		3.79E+00		**0.00E+00**		5.65E−01
F9	Mean	5.75E+01	+	4.51E+01	+	1.45E+02	+	**2.18E+01**	−	5.27E+01	+	3.54E+01
	SD	1.73E+01		1.50E+01		1.16E+01		**9.99E+00**		6.59E+00		6.93E+00
F10	Mean	4.59E+02	+	8.50E+02	+	2.13E+03	+	4.38E+02	+	**1.55E−01**	−	8.83E+01
	SD	2.16E+02		3.62E+02		5.09E+02		2.50E+02		**3.72E−02**		6.48E+01
F11	Mean	2.93E+03	+	2.57E+03	+	6.11E+03	+	**8.93E+02**	−	2.20E+03	=	2.08E+03
	SD	8.68E+02		5.61E+02		3.63E+02		**4.72E+02**		2.78E+02		3.82E+02
F12	Mean	1.75E+00	+	8.36E−01	=	2.48E+00	+	2.28E+00	+	**3.34E−01**	−	8.83E−01

[2] The source codes of DMSPSO, FIPS, and CLPSO are provided by Dr. P.N. Suganthan, and the source code of SL-PSO is downloaded from Dr. Y. Jin’s homepage http://www.surrey.ac.uk/cs/research/nice/ people/yaochu_jin/.

	SD	5.16E−01		1.99E−01		2.73E−01		5.99E−01		**5.34E−02**		1.49E−01
F13	Mean	4.81E−01	+	1.83E−01	−	2.96E−01	+	**1.62E−01**	−	2.94E−01	+	2.21E−01
	SD	1.32E−01		2.66E−02		3.33E−02		**3.23E−02**		3.88E−02		2.85E−02
F14	Mean	3.01E−01	+	**2.04E−01**	=	2.85E−01	+	3.98E−01	+	2.60E−01	+	2.14E−01
	SD	4.65E−02		**4.10E−02**		3.16E−02		7.73E−02		3.15E−02		2.88E−02
F15	Mean	7.60E+00	=	8.27E+00	+	1.50E+01	+	**5.97E+00**	−	8.27E+00	+	7.41E+00
	SD	2.56E+00		1.83E+00		9.32E−01		**4.27E+00**		1.01E+00		8.49E−01
F16	Mean	1.08E+01	+	1.05E+01	+	1.18E+01	+	1.20E+01	+	1.01E+01	+	**9.67E+00**
	SD	8.30E−01		4.90E−01		3.17E−01		2.62E−01		3.57E−01		**4.92E−01**
F17	Mean	3.07E+05	+	1.11E+05	−	4.20E+05	+	**8.28E+04**	−	9.45E+05	+	1.86E+05
	SD	1.75E+05		7.38E+04		2.19E+05		**7.24E+04**		5.13E+05		1.11E+05
F18	Mean	2.31E+03	+	1.12E+03	=	2.94E+03	+	1.40E+03	=	**9.40E+01**	−	9.05E+02
	SD	2.30E+03		1.07E+03		3.37E+03		2.43E+03		**4.91E+01**		1.20E+03
F19	Mean	6.98E+00	+	1.05E+01	+	5.72E+00	+	6.90E+00	+	7.75E+00	+	**3.74E+00**
	SD	2.55E+00		1.48E+01		1.45E+00		1.03E+00		5.96E−01		**6.12E−01**
F20	Mean	4.87E+02	+	3.54E+02	+	2.23E+03	+	2.19E+04	+	2.70E+03	+	**3.12E+02**
	SD	2.91E+02		1.38E+02		1.28E+03		1.36E+04		1.41E+03		**3.48E+02**
F21	Mean	8.77E+04	+	4.61E+04	=	9.08E+04	+	7.12E+04	+	8.94E+04	+	**3.85E+04**
	SD	6.75E+04		2.99E+04		5.86E+04		5.95E+04		4.96E+04		**3.19E+04**
F22	Mean	1.99E+02	+	2.36E+02	+	1.71E+02	+	1.85E+02	+	1.98E+02	+	**1.16E+02**
	SD	1.34E+02		7.31E+01		9.30E+01		1.17E+02		7.60E+01		**6.86E+01**
F23	Mean	3.15E+02	+	**3.15E+02**	−	3.16E+02	+	3.15E+02	+	3.15E+02	+	3.15E+02
	SD	1.46E−01		**2.19E−13**		3.33E−01		1.20E−12		1.33E−05		6.11E−13
F24	Mean	2.31E+02	+	2.23E+02	+	2.24E+02	+	2.32E+02	+	2.25E+02	+	**2.22E+02**
	SD	7.12E+00		5.01E+00		5.95E−01		6.59E+00		2.46E+00		**7.39E−01**
F25	Mean	2.08E+02	+	2.09E+02	+	2.08E+02	+	**2.04E+02**	=	2.08E+02	+	2.05E+02
	SD	1.75E+00		2.81E+00		1.08E+00		**1.25E+00**		1.17E+00		4.41E−01
F26	Mean	1.17E+02	+	1.67E+02	+	**1.00E+02**	−	1.07E+02	+	1.00E+02	−	1.04E+02
	SD	3.79E+01		4.79E+01		**4.48E−02**		2.53E+01		9.22E−02		1.82E+01
F27	Mean	5.46E+02	+	4.15E+02	+	3.33E+02	+	3.81E+02	+	4.14E+02	+	**3.08E+02**
	SD	1.06E+02		4.25E+01		4.96E+01		6.46E+01		5.28E+00		**2.94E+01**
F28	Mean	1.20E+03	+	9.93E+02	+	1.30E+03	+	9.00E+02	+	8.92E+02	+	**7.87E+02**
	SD	3.30E+02		1.85E+02		7.72E+01		7.59E+01		4.98E+01		**5.22E+01**
F29	Mean	1.26E+03	−	1.09E+03	−	4.01E+03	+	2.02E+03	+	**9.79E+02**	−	1.39E+03
	SD	8.50E+02		3.05E+02		1.85E+03		5.58E+02		**1.13E+02**		1.39E+02
F30	Mean	3.59E+03	+	1.90E+03	+	4.40E+03	+	3.26E+03	+	4.13E+03	+	**1.19E+03**
	SD	1.68E+03		4.78E+02		1.60E+03		1.01E+03		1.28E+03		**2.49E+02**
+			26		18		28		22		21	
=			2		4		1		2		1	
−			2		8		1		6		8	

"+", "−", and "=" symbolize a relation that BLPSO-5 is better than, worse than, or similar to its competitor, respectively, according to the Wilcoxon rank sum test at $\alpha = 0.05$

The best mean error values are marked in bold

Table 8 shows the ranking of BLPSO-5 and the other PSO variants according to the Friedman test on the 30-*D* functions. BLPSO-5 attains the best rank, DMSPSO the second, followed by CLPSO, SL-PSO, LDWPSO, and FIPS.

Table 9 shows the errors of BLPSO-5 and the other PSO variants on the 50-*D* functions. BLPSO-5, LDWPSO, DMSPSO, FIPS, SL-PSO, and CLPSO attain their best results on 14, 0, 3, 1, 4, and 8 functions, respectively. It can be observed that BLPSO-5 performs well on simple multimodal functions, hybrid functions, as well as composition functions. The three rows at the bottom section present the results of the Wilcoxon rank sum test between BLPSO-5 and the other PSO variants. BLPSO-5 performs significantly better than LDWPSO, DMSPSO, FIPS, SL-PSO, and CLPSO on 24, 19, 28, 21, and 22 functions, respectively, though it is significantly worse than LDWPSO, DMSPSO, FIPS, SLPSO, and CLPSO on 5, 6, 2, 5, and 2 functions, and it is similar to them on 1, 5, 0, 4, and 6 functions, respectively.

Table 10 shows the ranking of BLPSO-5 and the other PSO variants according to the Friedman test on the 50-*D* functions. BLPSO-5 attains the best rank, CLPSO the second, followed by DMSPSO, SL-PSO, LDWPSO, and FIPS.

Based on the comparisons with the other PSO variants on both 30-*D* and 50-*D* benchmark functions, it is fair to say that the performance of BLPSO-5 is the best among the PSO variants. Also, BLPSO-5 performs particularly well on the composition functions.

Table 8 Ranking of BLPSO-5 and the other PSO variants according to the Friedman test on the 30-*D* functions

	LDWPSO	DMSPSO	FIPSO	SL-PSO	CLPSO	BLPSO-5
Friedman rank	4.25	3.25	4.73	3.47	3.30	2.00
Final rank	5	2	6	4	3	1

Table 9 Errors of BLPSO-5 and the other PSO variants on the 50-*D* functions

		LDWPSO		DMSPSO		FIPSO		SL-PSO		CLPSO		BLPSO-5
F1	Mean	6.54E+06	=	8.93E+05	−	3.43E+07	+	**8.81E+05**	−	1.62E+07	+	5.10E+06
	SD	5.07E+06		3.06E+05		9.04E+06		**2.92E+05**		4.01E+06		1.28E+06
F2	Mean	3.32E+03	=	4.66E+03	=	2.97E+04	+	8.40E+03	=	**4.68E+01**	−	3.44E+03
	SD	4.67E+03		4.72E+03		3.10E+04		8.19E+03		**4.82E+01**		2.35E+03
F3	Mean	2.93E+03	+	1.37E+03	+	1.15E+04	+	1.93E+04	+	1.76E+03	+	**4.23E+01**
	SD	2.16E+03		6.57E+02		2.39E+03		9.39E+03		8.01E+02		**8.97E+01**
F4	Mean	1.89E+02	+	**5.40E+01**	−	2.68E+02	+	9.40E+01	+	1.01E+02	+	8.64E+01
	SD	5.58E+01		**3.36E+01**		3.70E+01		5.00E+00		8.95E+00		5.04E+00
F5	Mean	2.11E+01	+	2.05E+01	−	2.11E+01	+	2.11E+01	+	**2.05E+01**	−	2.09E+01
	SD	7.13E−02		8.75E−02		3.65E−02		3.28E−02		**3.40E−02**		5.07E−02
F6	Mean	2.60E+01	+	1.51E+01	+	2.02E+01	+	2.52E+00	+	2.93E+01	+	**9.22E−02**
	SD	4.37E+00		4.44E+00		5.56E+00		1.68E+00		2.31E+00		**3.21E−01**
F7	Mean	9.11E−03	+	3.45E−03	=	6.07E−05	+	1.40E−03	+	1.09E−03	+	**2.92E−13**
	SD	9.46E−03		6.12E−03		1.72E−04		3.76E−03		1.08E−03		**6.46E−14**
F8	Mean	4.79E+01	+	1.02E+02	+	1.60E+02	+	3.72E+01	+	**1.17E−13**	−	4.97E−01
	SD	1.17E+01		2.18E+01		1.70E+01		8.29E+00		**2.08E−14**		8.16E−01
F9	Mean	1.38E+02	+	1.06E+02	+	3.25E+02	+	1.67E+02	+	1.27E+02	+	7.10E+01
	SD	2.28E+01		2.84E+01		1.80E+01		9.57E+01		1.33E+01		9.02E+00
F10	Mean	1.09E+03	+	2.94E+03	+	6.63E+03	+	1.19E+03	+	**4.53E−01**	−	3.63E+02
	SD	4.06E+02		6.84E+02		5.84E+02		4.69E+02		**3.02E−01**		1.81E+02
F11	Mean	6.12E+03	+	5.97E+03	+	1.26E+04	+	**2.06E+03**	−	4.93E+03	+	4.46E+03
	SD	2.42E+03		8.40E+02		3.03E+02		**6.46E+02**		4.11E+02		4.77E+02
F12	Mean	2.65E+00	+	1.10E+00	+	3.36E+00	+	3.16E+00	+	**3.68E−01**	−	8.77E−01
	SD	5.62E−01		2.01E−01		2.69E−01		5.24E−01		**5.99E−02**		1.18E−01
F13	Mean	5.83E−01	+	3.01E−01	=	4.41E−01	+	3.15E−01	+	3.95E−01	+	**2.86E−01**
	SD	1.19E−01		5.12E−02		4.78E−02		4.27E−02		3.42E−02		**3.72E−02**
F14	Mean	4.07E−01	+	**2.50E−01**	=	3.48E−01	+	4.50E−01	+	3.06E−01	+	2.65E−01
	SD	1.91E−01		**3.74E−02**		3.81E−02		1.33E−01		2.17E−02		2.42E−02
F15	Mean	1.88E+01	+	1.92E+01	+	3.22E+01	+	2.75E+01	+	1.83E+01	+	**1.48E+01**
	SD	6.07E+00		3.09E+00		1.41E+00		4.47E+00		2.22E+00		**1.33E+00**
F16	Mean	2.08E+01	+	1.94E+01	+	2.15E+01	+	2.18E+01	+	1.86E+01	+	**1.82E+01**
	SD	8.41E−01		7.28E−01		3.10E−01		2.59E−01		4.20E−01		**4.73E−01**
F17	Mean	5.52E+05	−	1.97E+05	−	1.78E+06	+	**9.95E+04**	−	2.78E+06	+	5.97E+05
	SD	4.48E+05		2.66E+05		7.18E+05		**6.30E+04**		9.94E+05		2.10E+05
F18	Mean	5.38E+02	=	1.19E+03	+	1.18E+03	+	1.33E+03	+	**1.69E+02**	=	3.73E+02
	SD	4.59E+02		1.03E+03		9.53E+02		1.43E+03		**6.47E+01**		3.68E+02
F19	Mean	4.37E+01	+	3.34E+01	+	6.51E+01	+	1.83E+01	=	**1.81E+01**	=	2.16E+01
	SD	2.82E+01		1.94E+01		1.91E+01		5.25E+00		**3.06E+00**		9.78E+00
F20	Mean	1.52E+03	+	5.40E+02	+	1.86E+03	+	2.74E+04	+	4.92E+03	+	**2.57E+02**
	SD	7.99E+02		1.29E+02		6.23E+02		1.31E+04		1.87E+03		**1.41E+02**
F21	Mean	3.51E+05	=	1.39E+05	−	1.19E+06	+	**9.91E+04**	−	1.56E+06	+	3.80E+05
	SD	2.56E+05		7.26E+04		4.72E+05		**5.86E+04**		7.61E+05		1.44E+05
F22	Mean	8.20E+02	+	4.00E+02	+	1.02E+03	+	3.82E+02	=	7.18E+02	+	**2.64E+02**
	SD	2.65E+02		1.78E+02		2.84E+02		3.14E+02		1.65E+02		**1.30E+02**
F23	Mean	3.45E+02	+	**3.44E+02**	+	3.51E+02	+	3.44E+02	+	3.44E+02	+	3.44E+02
	SD	7.79E−01		**1.73E−13**		1.26E+00		3.60E−13		7.48E−05		2.56E−13
F24	Mean	2.76E+02	+	2.71E+02	+	**2.58E+02**	=	2.73E+02	+	2.60E+02	+	2.58E+02
	SD	3.71E+00		4.64E+00		**4.02E+00**		3.03E+00		3.62E+00		4.07E+00
F25	Mean	2.19E+02	+	2.24E+02	+	2.21E+02	+	2.12E+02	=	2.16E+02	+	**2.10E+02**
	SD	3.29E+00		4.19E+00		2.16E+00		3.43E+00		1.68E+00		**7.36E−01**

F26	Mean	1.37E+02	=	1.70E+02	=	1.39E+02	=	1.47E+02	=	**1.01E+02**	−	1.47E+02
	SD	4.91E+01		4.65E+01		5.11E+01		5.04E+01		**1.05E−01**		5.08E+01
F27	Mean	1.01E+03	+	7.69E+02	+	6.99E+02	+	4.63E+02	+	8.74E+02	+	**3.24E+02**
	SD	8.97E+01		9.17E+01		1.80E+02		6.57E+01		2.92E+02		**2.85E+01**
F28	Mean	2.26E+03	+	2.22E+03	+	2.88E+03	+	1.36E+03	+	1.48E+03	+	**1.14E+03**
	SD	6.77E+02		6.94E+02		2.18E+02		2.29E+02		1.30E+02		**4.20E+01**
F29	Mean	1.36E+07	+	1.48E+03	=	2.62E+04	+	2.53E+03	+	1.54E+03	+	**1.36E+03**
	SD	4.24E+07		4.27E+02		1.31E+04		5.95E+02		2.51E+02		**1.82E+02**
F30	Mean	3.17E+04	+	1.15E+04	+	5.20E+04	+	1.28E+04	+	1.04E+04	+	**9.09E+03**
	SD	1.04E+04		1.06E+03		1.10E+04		2.43E+03		9.29E+02		**3.05E+02**
+			24		19		28		21		22	
=			5		6		2		5		2	
−			1		5		0		4		6	

"+", "−", and "=" symbolize a relation that BLPSO-5 is better than, worse than, or similar to its competitor, respectively, according to the Wilcoxon rank sum test at $\alpha = 0.05$
The best mean error values are marked in bold

Table 10 Ranking of BLPSO-5 and the other PSO variants according to the Friedman test on the 50-*D* functions

	LDWPSO	DMSPSO	FIPSO	SL-PSO	CLPSO	BLPSO-5
Friedman rank	4.33	3.23	4.98	3.60	2.97	1.88
Final rank	5	3	6	4	2	1

4.3 Comparisons with BBO algorithms

BLPSO can be viewed as a hybrid algorithm of CLPSO and BBO. Therefore, it makes sense to compare BLPSO with the major BBO algorithms. In the following, four representative BBO are selected for comparisons:

- Real code BBO (RCBBO) (Gong et al. 2010b)
- Real code BBO with Gaussian mutation (RCBBOG) (Gong et al. 2010b)
- Perturb BBO (PBBO) (Li et al. 2011)
- Hybrid differential evolution with BBO (DE/BBO) (Gong et al. 2010a)

Table 11 shows the parameter settings for the BBO algorithms.

Table 12 shows the errors of BLPSO-5 and the BBO algorithms on the 30-*D* functions. BLPSO-5, RCBBO, RCBBOG, PBBO, and DE/BBO attain their best results on 17, 2, 0, 3, and 8 functions, respectively. Based on the results of the Wilcoxon rank sum test, it can be observed that BLPSO5 performs significantly better than RCBBO, RCBBOG, PBBO, and DE/BBO on 25, 23, 23, and 17 functions, respectively, though BLPSO-5 performs significantly worse than RCBBO, RCBBOG, PBBO, and DE/BBO on 4, 3, 5, and 10 functions, and it is similar to them on 1, 4, 2, and 3 functions, respectively.

Table 13 shows the ranking of BLPSO-5 and the BBO algorithms according to the Friedman test on the 30-*D* functions. BLPSO-5 attains the best rank, DE/BBO the second, followed by PBBO, RCBBO, and RCBBOG.

Table 11 Parameter settings for the compared BBO

Algorithm	Parameter settings
RCBBO	Population size N = 100, maximum habitat probability $mmax$ = 0.005, maximum possible immigration and emigration rates $I = E = 1$, linear migration model
RCBBOG	Population size N = 100, maximum habitat probability $mmax$ = 0.005, maximum possible immigration and emigration rates $I = E = 1$, linear migration model
PBBO	Population size N = 100, maximum habitat probability $mmax$ = 0.005, maximum possible immigration and emigration rates $I = E = 1$, sinusoidal migration model
DE/BBO	Population size N = 100, maximum habitat probability $mmax$ = 0.005, maximum possible immigration and emigration rates $I = E = 1$, scaling factor $F = rand(0.1, 1)$, crossover probability CR = 0.9, linear migration model

Table 12 Errors of BLPSO-5 and BBO on the 30-*D* functions

		RCBBO		RCBBOG		PBBO		DE/BBO		BLPSO-5
F1	Mean	2.75E+07	+	3.77E+06	=	**1.55E+06**	−	1.01E+07	+	2.99E+06
	SD	2.26E+07		2.16E+06		**1.08E+06**		2.93E+06		1.10E+06
F2	Mean	1.35E+06	+	9.36E+03	=	1.09E+04	+	**2.72E+01**	−	5.09E+03
	SD	5.76E+05		1.08E+04		9.93E+03		**1.42E+02**		4.25E+03
F3	Mean	1.51E+04	+	1.53E+04	+	1.37E+04	+	**5.31E−01**	−	3.67E+00
	SD	1.20E+04		1.19E+04		1.25E+04		**6.87E−01**		1.16E+01
F4	Mean	1.19E+02	+	7.70E+01	+	5.92E+01	+	6.25E+01	+	**2.68E+01**
	SD	3.54E+01		3.37E+01		3.62E+01		1.67E+01		**3.47E+01**
F5	Mean	2.02E+01	−	2.00E+01	−	**2.00E+01**	−	2.04E+01	−	2.08E+01
	SD	4.02E−02		4.44E−03		**4.45E−06**		4.28E−02		7.01E−02
F6	Mean	1.47E+01	+	2.09E+01	+	1.39E+01	+	4.53E+00	+	**9.37E−06**
	SD	2.72E+00		2.94E+00		3.81E+00		5.68E+00		**3.20E−05**
F7	Mean	8.38E−01	+	1.98E−02	+	4.97E−03	+	**4.17E−14**	−	9.47E−14
	SD	1.27E−01		2.75E−02		7.34E−03		**5.57E−14**		4.31E−14
F8	Mean	2.79E−01	+	2.83E+01	+	1.52E+01	+	**1.89E−14**	=	2.32E−01
	SD	1.15E−01		7.33E+00		5.12E+00		**4.31E−14**		5.65E−01
F9	Mean	5.05E+01	+	5.54E+01	+	5.12E+01	+	6.07E+01	+	**3.54E+01**
	SD	1.47E+01		1.51E+01		1.46E+01		5.90E+00		**6.93E+00**
F10	Mean	**1.68E+00**	−	1.61E+02	=	1.39E+02	=	1.38E+01	−	8.83E+01
	SD	**4.72E−01**		1.22E+02		1.23E+02		3.18E+00		6.48E+01
F11	Mean	2.39E+03	=	3.22E+03	+	2.96E+03	+	3.15E+03	+	**2.08E+03**
	SD	7.67E+02		5.42E+02		5.49E+02		2.51E+02		**3.82E+02**
F12	Mean	**2.06E−01**	−	2.29E−01	−	2.32E−01	−	5.83E−01	−	8.83E−01
	SD	**5.63E−02**		1.17E−01		1.07E−01		7.89E−02		1.49E−01
F13	Mean	4.54E−01	+	3.85E−01	+	3.21E−01	+	3.14E−01	+	**2.21E−01**
	SD	1.12E−01		9.87E−02		8.20E−02		3.02E−02		**2.85E−02**
F14	Mean	4.05E−01	+	4.77E−01	+	3.62E−01	+	2.63E−01	+	**2.14E−01**
	SD	1.65E−01		2.24E−01		1.17E−01		2.15E−02		**2.88E−02**
F15	Mean	1.42E+01	+	4.37E+01	+	**6.10E+00**	−	7.70E+00	=	7.41E+00
	SD	4.88E+00		1.49E+01		**1.77E+00**		8.34E−01		8.49E−01
F16	Mean	1.01E+01	+	1.18E+01	+	1.18E+01	+	1.02E+01	+	**9.67E+00**
	SD	8.45E−01		7.54E−01		6.17E−01		3.09E−01		**4.92E−01**
F17	Mean	3.99E+06	+	1.13E+06	+	6.42E+05	+	6.94E+05	+	**1.86E+05**
	SD	1.97E+06		7.61E+05		4.45E+05		2.95E+05		**1.11E+05**
F18	Mean	9.69E+04	+	3.67E+03	+	9.52E+02	=	1.80E+03	=	**9.05E+02**
	SD	6.42E+04		4.95E+03		1.03E+03		1.75E+03		**1.20E+03**
F19	Mean	2.58E+01	+	1.49E+01	+	8.07E+00	+	4.83E+00	+	**3.74E+00**
	SD	2.82E+01		1.11E+01		1.08E+01		3.72E−01		**6.12E−01**
F20	Mean	4.62E+04	+	3.83E+04	+	3.07E+04	+	1.19E+03	+	**3.12E+02**
	SD	2.32E+04		1.81E+04		1.68E+04		6.63E+02		**3.48E+02**
F21	Mean	1.09E+06	+	4.35E+05	+	3.70E+05	+	7.46E+04	+	**3.85E+04**
	SD	8.69E+05		3.50E+05		2.47E+05		2.85E+04		**3.19E+04**
F22	Mean	5.24E+02	+	4.90E+02	+	5.30E+02	+	**5.03E+01**	−	1.16E+02
	SD	1.97E+02		1.97E+02		2.32E+02		**3.02E+01**		6.86E+01
F23	Mean	3.17E+02	+	3.15E+02	+	3.15E+02	+	**3.15E+02**	−	3.15E+02
	SD	1.86E+00		1.68E−02		5.33E−04		**5.78E−14**		6.11E−13
F24	Mean	2.35E+02	+	2.47E+02	+	2.35E+02	+	2.23E+02	+	**2.22E+02**
	SD	3.56E+00		4.37E+00		5.59E+00		8.32E−01		**7.39E−01**
F25	Mean	2.11E+02	+	2.15E+02	+	2.11E+02	+	2.07E+02	+	**2.05E+02**
	SD	2.75E+00		6.88E+00		4.16E+00		9.81E−01		**4.41E−01**
F26	Mean	1.00E+02	−	1.01E+02	−	1.00E+02	−	**1.00E+02**	−	1.04E+02
	SD	8.45E−02		1.48E−01		8.78E−02		**3.61E−02**		1.82E+01
F27	Mean	6.11E+02	+	6.89E+02	+	5.30E+02	+	3.19E+02	+	**3.08E+02**
	SD	1.19E+02		2.20E+02		1.18E+02		3.68E+01		**2.94E+01**
F28	Mean	9.74E+02	+	1.14E+03	+	9.31E+02	+	**7.80E+02**	−	7.87E+02
	SD	6.99E+01		2.44E+02		5.33E+01		**2.04E+01**		5.22E+01
F29	Mean	2.28E+03	+	1.50E+03	=	1.63E+03	+	1.80E+03	+	**1.39E+03**
	SD	4.82E+02		3.29E+02		3.87E+02		3.04E+02		**1.39E+02**

F30	Mean	5.84E+03	+	4.20E+03	+	3.62E+03	+	1.88E+03	+	**1.19E+03**
	SD	2.45E+03		4.67E+03		9.00E+02		6.57E+02		**2.49E+02**
+			25		23		23		17	
=			1		4		2		3	
−			4		3		5		10	

"+", "−", and "=" symbolize a relation that BLPSO-5 is better than, worse than, or similar to its competitor, respectively, according to the Wilcoxon rank sum test at $\alpha = 0.05$

The best mean error values are marked in bold

Table 13 Ranking of BLPSO-5 and BBO according to the Friedman test on the 30-*D* functions

	RCBBO	RCBBOG	PBBO	DE/BBO	BLPSO-5
Friedman rank	3.87	4.02	2.92	2.43	1.77
Final rank	4	5	3	2	1

Table 14 shows the errors of BLPSO-5 and the BBO algorithms on the 50-*D* functions. BLPSO-5, RCBBO, RCBBOG, PBBO, and DE/BBO attain their best results on 17, 3, 1, 2, and 7 functions, respectively. Based on the results of the Wilcoxon rank sum test, it can be observed that BLPSO5 performs significantly better than RCBBO, RCBBOG, PBBO, and DE/BBO on 24, 24, 22, and 18 functions, respectively, though BLPSO-5 performs significantly worse than RCBBO, RCBBOG, PBBO, and DE/BBO on 4, 3, 5, and 8 functions, and it is similar to them on 2, 3, 3, and 4 functions, respectively.

Table 15 shows the ranking of BLPSO-5 and the BBO algorithms according to the Friedman test on the 50-*D* functions. BLPSO-5 attains the best rank, DE/BBO the second, followed by PBBO, RCBBO, and RCBBOG.

Based on the comparisons with the BBO algorithms on both 30-*D* and 50-*D* benchmark functions, it can be observed that BLPSO-5 performs significantly better than all the four BBO algorithms.

Table 14 Errors of BLPSO-5 and the BBO algorithms on the 50-*D* functions

		RCBBO		RCBBOG		PBBO		DE/BBO		BLPSO-5
F1	Mean	1.55E+07	+	3.35E+06	−	**1.94E+06**	−	2.07E+07	+	5.10E+06
	SD	6.54E+06		1.39E+06		**6.00E+05**		5.56E+06		1.28E+06
F2	Mean	8.69E+05	+	3.52E+03	=	4.91E+03	=	3.44E+03	=	**3.44E+03**
	SD	3.52E+05		4.81E+03		4.26E+03		2.81E+03		**2.35E+03**
F3	Mean	3.14E+04	+	3.41E+04	+	3.15E+04	+	1.08E+03	+	**4.23E+01**
	SD	1.38E+04		1.66E+04		1.45E+04		4.25E+02		**8.97E+01**
F4	Mean	1.15E+02	+	9.76E+01	+	8.81E+01	+	9.68E+01	+	**8.64E+01**
	SD	3.16E+01		3.40E+01		2.66E+01		2.23E+00		**5.04E+00**
F5	Mean	2.01E+01	−	**2.00E+01**	−	2.00E+01	−	2.06E+01	−	2.09E+01
	SD	3.12E−02		**5.33E−05**		3.43E−03		4.35E−02		5.07E−02
F6	Mean	2.70E+01	+	4.13E+01	+	2.40E+01	+	1.41E+01	+	**9.22E−02**
	SD	4.45E+00		5.60E+00		6.07E+00		1.17E+01		**3.21E−01**
F7	Mean	7.89E−01	+	1.42E−02	+	6.33E−03	+	**2.31E−13**	−	2.92E−13
	SD	9.33E−02		7.10E−03		6.42E−03		**5.57E−14**		6.46E−14
F8	Mean	1.93E−01	−	5.78E+01	+	3.50E+01	+	**6.99E−12**	=	4.97E−01
	SD	6.56E−02		1.07E+01		6.16E+00		**3.69E−11**		8.16E−01
F9	Mean	9.85E+01	+	1.06E+02	+	1.03E+02	+	1.32E+02	+	**7.10E+01**
	SD	1.55E+01		2.48E+01		2.88E+01		1.32E+01		**9.02E+00**
F10	Mean	**1.31E+00**	−	5.54E+02	+	4.31E+02	=	3.98E+01	−	3.63E+02
	SD	**3.40E−01**		2.92E+02		2.13E+02		6.56E+00		1.81E+02
F11	Mean	4.48E+03	=	5.79E+03	+	5.68E+03	+	6.43E+03	+	**4.46E+03**
	SD	7.89E+02		8.05E+02		7.22E+02		4.62E+02		**4.77E+02**
F12	Mean	**1.78E−01**	−	2.49E−01	−	2.07E−01	−	6.44E−01	−	8.77E−01
	SD	**5.09E−02**		8.51E−02		6.84E−02		7.41E−02		1.18E−01
F13	Mean	5.44E−01	+	5.26E−01	+	4.13E−01	+	4.27E−01	+	**2.86E−01**
	SD	1.17E−01		1.20E−01		9.04E−02		4.57E−02		**3.72E−02**
F14	Mean	3.99E−01	+	5.47E−01	+	4.14E−01	+	3.34E−01	=	**2.65E−01**
	SD	1.59E−01		2.50E−01		1.40E−01		1.76E−01		**2.42E−02**
F15	Mean	2.85E+01	+	9.56E+01	+	1.88E+01	+	1.65E+01	+	**1.48E+01**
	SD	7.27E+00		1.96E+01		5.44E+00		1.32E+00		**1.33E+00**

F16	Mean	**1.81E+01**	=	2.13E+01	+	2.12E+01	+	1.89E+01	+	1.82E+01
	SD	**7.66E−01**		6.47E−01		9.03E−01		3.39E−01		4.73E−01
F17	Mean	6.02E+06	+	9.61E+05	+	**5.07E+05**	−	3.42E+06	+	5.97E+05
	SD	3.01E+06		5.13E+05		**2.89E+05**		8.87E+05		2.10E+05
F18	Mean	7.40E+04	+	1.33E+03	+	1.00E+03	+	3.76E+02	=	**3.73E+02**
	SD	4.93E+04		1.28E+03		1.02E+03		4.51E+02		**3.68E+02**
F19	Mean	5.26E+01	+	2.27E+01	=	1.38E+01	−	**1.14E+01**	−	2.16E+01
	SD	1.98E+01		5.28E+00		2.09E+00		**5.36E−01**		9.78E+00
F20	Mean	6.53E+04	+	7.82E+04	+	7.00E+04	+	3.49E+03	+	**2.57E+02**
	SD	2.47E+04		3.66E+04		2.81E+04		1.18E+03		**1.41E+02**
F21	Mean	6.86E+06	+	1.42E+06	+	7.36E+05	+	1.38E+06	+	**3.80E+05**
	SD	4.87E+06		7.65E+05		4.76E+05		5.64E+05		**1.44E+05**
F22	Mean	1.09E+03	+	1.11E+03	+	1.22E+03	+	5.05E+02	+	**2.64E+02**
	SD	2.99E+02		3.82E+02		2.68E+02		1.07E+02		**1.30E+02**
F23	Mean	3.46E+02	+	3.44E+02	+	3.44E+02	+	**3.44E+02**	+	3.44E+02
	SD	1.95E+00		2.60E−04		9.18E−07		**1.79E−13**		2.56E−13
F24	Mean	2.67E+02	+	2.98E+02	+	2.76E+02	+	2.61E+02	+	**2.58E+02**
	SD	5.86E+00		7.16E+00		3.54E+00		4.23E+00		**4.07E+00**
F25	Mean	2.19E+02	+	2.27E+02	+	2.21E+02	+	2.16E+02	+	**2.10E+02**
	SD	3.96E+00		9.51E+00		7.64E+00		1.70E+00		**7.36E−01**
F26	Mean	1.53E+02	+	1.40E+02	=	1.40E+02	=	**1.00E+02**	−	1.47E+02
	SD	6.30E+01		4.96E+01		4.98E+01		**4.48E−02**		5.08E+01
F27	Mean	1.01E+03	+	1.33E+03	+	8.43E+02	+	3.48E+02	+	**3.24E+02**
	SD	8.68E+01		9.56E+01		1.15E+02		1.72E+01		**2.85E+01**
F28	Mean	1.63E+03	+	1.84E+03	+	1.53E+03	+	**1.04E+03**	−	1.14E+03
	SD	2.05E+02		2.47E+02		3.51E+02		**2.03E+01**		4.20E+01
F29	Mean	4.29E+03	+	2.39E+03	+	2.49E+03	+	2.30E+03	+	**1.36E+03**
	SD	9.91E+02		7.93E+02		5.99E+02		3.08E+02		**1.82E+02**
F30	Mean	1.49E+04	+	1.33E+04	+	1.34E+04	+	**8.15E+03**	−	9.09E+03
	SD	2.69E+03		2.03E+03		1.95E+03		**2.53E+02**		3.05E+02
+			24		24		22		18	
=			2		3		3		4	
−			4		3		5		8	

"+", "−", and "=" symbolize a relation that BLPSO-5 is better than, worse than, or similar to its competitor, respectively, according to the Wilcoxon rank sum test at $\alpha = 0.05$
The best mean error values are marked in bold

Table 15 Ranking of BLPSO-5 and the BBO algorithms according to the Friedman test on the 30-*D* functions

	RCBBO	RCBBOG	PBBO	DE/BBO	BLPSO-5
Friedman rank	3.67	3.98	3.05	2.45	1.85
Final rank	4	5	3	2	1

4.4 Comparisons with the other EAs

We further compare BLPSO with four non-PSO and non-BBO algorithms. The four EAs are:[3]

- Covariance matrix adaptation evolution strategy (CMAES) (Hansen and Ostermeier 2001)
- Global and local real-coded genetic algorithms based on parent-centric crossover operators (GL-25)(GarciaMartinez et al. 2008)
- Gaussian bare-bones artificial bee colony (GBABC) (Zhou et al. 2014)
- Adaptive differential evolution with Optional External Archive (JADE) (Zhang and Sanderson 2009)

Table 16 shows the parameter settings for these EAs.

Table 17 shows the errors of BLPSO-5 and the other EAs on the 30-*D* functions. BLPSO-5, CMAES, GL-25, GBABC, and JADE attain their best results on 6, 6, 2, 4, and 12 functions, respectively. Based on the results of the Wilcoxon rank sum test, it can be observed that BLPSO-5 performs significantly better than CMAES, GL-25, GBABC,

[3] The source codes of CMAES, GL-25, and JADE are downloaded from Dr. Y. Wang's homepage http://ist.csu.edu.cn/YongWang.htm.

and JADE on 16, 18, 14, and 8 functions, respectively, though BLPSO-5 performs significantly worse than CMAES, GL-25, GBABC, and JADE on 12, 7, 12, and 19 functions, and it is similar to them on 2, 5, 4, and 3 functions, respectively.

Table 18 shows the ranking of BLPSO-5 and the other EAs according to the Friedman test on the 30-*D* functions. JADE attains the best rank, BLPSO-5 the second, followed by GBABC, CMAES, and GL-25.

Table 16 Parameter settings for the EAs

Algorithm	Parameter settings
CMAES	Number of parent individuals $\mu = floor(\lambda/2)$, number of offspring individuals $\lambda = 4 + floor(3\log(D))$
GL-25	Population size N = 400,
GBABC	Population size N = 60, limit = 100, crossover probability CR = 0.3,
JADE	Population size NP = 100, c = 0.1, p = 0.05

Table 17 Errors of BLPSO-5 and the other EAs on the 30-*D* functions

		CMAES		GL-25		GBABC		JADE		BLPSO-5
F1	Mean	**2.70E−14**	−	8.52E+05	−	9.61E+06	+	4.43E+02	−	2.99E+06
	SD	**9.40E−15**		9.84E+05		2.90E+06		8.28E+02		1.10E+06
F2	Mean	5.59E−14	−	3.37E+03	−	2.03E+01	−	**1.33E−14**	−	5.09E+03
	SD	2.30E−14		5.27E+03		5.97E+01		**1.44E−14**		4.25E+03
F3	Mean	**1.08E−13**	−	2.18E−01	=	9.70E+01	+	3.10E−04	−	3.67E+00
	SD	**3.76E−14**		6.56E−01		1.42E+02		1.49E−03		1.16E+01
F4	Mean	**1.33E−01**	−	9.17E+01	+	3.52E+01	=	1.80E+01	−	2.68E+01
	SD	**7.28E−01**		1.02E+01		3.32E+01		3.70E+01		3.47E+01
F5	Mean	**2.00E+01**	−	2.10E+01	+	2.04E+01	−	2.03E+01	−	2.08E+01
	SD	**5.32E−06**		3.45E−02		3.44E−02		3.07E−02		7.01E−02
F6	Mean	4.18E+01	+	7.25E+00	+	5.74E+00	+	1.26E+01	+	**9.37E−06**
	SD	1.05E+01		3.61E+00		2.07E+00		1.33E+00		**3.20E−05**
F7	Mean	1.23E−03	+	1.70E−11	+	2.77E−13	+	1.54E−08	+	**9.47E−14**
	SD	3.28E−03		4.41E−11		9.76E−14		8.38E−08		**4.31E−14**
F8	Mean	4.13E+02	+	2.38E+01	+	**1.78E−13**	−	0.00E+00	−	2.32E−01
	SD	6.71E+01		6.23E+00		**6.46E−14**		0.00E+00		5.65E−01
F9	Mean	6.43E+02	+	5.39E+01	=	4.86E+01	+	**2.44E+01**	−	3.54E+01
	SD	1.28E+02		5.34E+01		1.03E+01		**3.71E+00**		6.93E+00
F10	Mean	4.87E+03	+	1.11E+03	+	1.01E+00	−	**8.33E−03**	−	8.83E+01
	SD	5.75E+02		4.57E+02		7.56E−01		**1.51E−02**		6.48E+01
F11	Mean	5.15E+03	+	5.55E+03	+	2.65E+03	+	**1.82E+03**	−	2.08E+03
	SD	7.76E+02		2.13E+03		2.73E+02		**3.03E+02**		3.82E+02
F12	Mean	**3.17E−01**	−	2.96E+00	+	5.81E−01	−	3.59E−01	−	8.83E−01
	SD	**3.79E−01**		1.93E−01		7.65E−02		5.21E−02		1.49E−01
F13	Mean	2.48E−01	=	3.22E−01	+	**1.70E−01**	−	2.12E−01	=	2.21E−01
	SD	6.17E−02		4.93E−02		**2.59E−02**		2.69E−02		2.85E−02
F14	Mean	3.77E−01	+	2.95E−01	+	9.57E−02	−	3.09E−01	+	**2.14E−01**
	SD	1.08E−01		2.80E−02		2.05E−02		1.13E−01		**2.88E−02**
F15	Mean	3.48E+00	−	1.27E+01	+	5.43E+00	−	**3.19E+00**	−	7.41E+00
	SD	8.60E−01		5.13E+00		1.17E+00		**3.39E−01**		8.49E−01
F16	Mean	1.43E+01	+	1.21E+01	+	1.08E+01	+	9.88E+00	=	**9.67E+00**
	SD	4.91E−01		3.10E−01		3.42E−01		3.66E−01		**4.92E−01**
F17	Mean	1.75E+03	−	1.68E+05	=	9.94E+05	+	**1.33E+03**	−	1.86E+05
	SD	4.54E+02		8.63E+04		4.81E+05		**4.37E+02**		1.11E+05
F18	Mean	1.42E+02	−	2.34E+02	−	2.24E+03	=	**9.71E+01**	−	9.05E+02
	SD	4.03E+01		2.28E+02		2.79E+03		**5.29E+01**		1.20E+03
F19	Mean	9.74E+00	+	5.01E+00	+	4.90E+00	+	5.49E+00	+	**3.74E+00**
	SD	1.57E+00		5.63E−01		7.26E−01		8.32E−01		**6.12E−01**
F20	Mean	2.72E+02	=	**1.62E+02**	−	1.36E+03	+	3.49E+03	+	3.12E+02
	SD	1.08E+02		**6.32E+01**		7.13E+02		2.87E+03		3.48E+02
F21	Mean	**1.00E+03**	−	6.21E+04	+	1.11E+05	+	5.44E+03	−	3.85E+04
	SD	**3.46E+02**		2.82E+04		5.50E+04		2.80E+04		3.19E+04

F22	Mean	2.30E+02	+	1.53E+02	+	1.29E+02	=	1.71E+02	+	**1.16E+02**
	SD	1.28E+02		4.95E+01		7.45E+01		7.22E+01		**6.86E+01**
F23	Mean	3.15E+02	+	3.15E+02	+	**3.12E+02**	−	3.15E+02	−	3.15E+02
	SD	3.88E−12		2.68E−09		**1.79E+01**		7.77E−02		6.11E−13
F24	Mean	2.41E+02	+	2.22E+02	=	**2.03E+02**	−	2.25E+02	+	2.22E+02
	SD	4.83E+01		4.75E−01		**5.39E+00**		2.98E+00		7.39E−01
F25	Mean	2.04E+02	−	2.07E+02	+	2.08E+02	+	**2.03E+02**	−	2.05E+02
	SD	3.03E+00		1.31E+00		9.34E−01		**6.27E−01**		4.41E−01
F26	Mean	1.57E+02	+	1.00E+02	−	1.00E+02	−	**1.00E+02**	=	1.04E+02
	SD	1.80E+02		4.36E−02		4.85E−02		**3.81E−02**		1.82E+01
F27	Mean	3.68E+02	+	**3.02E+02**	−	4.03E+02	+	3.99E+02	+	3.08E+02
	SD	3.62E+01		**6.65E−01**		1.01E+00		7.01E+01		2.94E+01
F28	Mean	4.32E+03	+	8.92E+02	+	7.96E+02	=	**5.98E+02**	−	7.87E+02
	SD	3.35E+03		2.65E+01		4.57E+01		**5.59E+01**		5.22E+01
F29	Mean	8.17E+02	−	1.02E+03	−	1.12E+03	−	**2.17E+02**	−	1.39E+03
	SD	8.24E+01		1.06E+02		1.99E+02		**7.96E+00**		1.39E+02
F30	Mean	2.37E+03	+	1.26E+03	=	2.02E+03	+	**7.32E+02**	−	1.19E+03
	SD	5.94E+02		2.66E+02		6.95E+02		**1.71E+02**		2.49E+02
+			16		18		14		8	
=			2		5		4		3	
−			12		7		12		19	

"+", "−", and "=" symbolize a relation that BLPSO-5 is better than, worse than, or similar to its competitor, respectively, according to the Wilcoxon rank sum test at $\alpha = 0.05$
The best mean error values are marked in bold

Table 18 Ranking of BLPSO-5 and the other EAs according to the Friedman test on the 30-*D* functions

	CMAES	GL-25	GBABC	JADE	BLPSO-5
Friedman rank	3.30	3.47	3.08	2.25	2.90
Final rank	4	5	3	1	2

Table 19 shows the errors of BLPSO-5 and the other EAs on the 50-*D* functions. BLPSO-5, CMAES, GL-25, GBABC, and JADE attain their best results on 6, 8, 0, 5, and 11 functions, respectively. Based on the results of the Wilcoxon rank sum test, it can be observed that BLPSO-5 performs significantly better than CMAES, GL-25, GBABC, and JADE on 16, 21, 17, and 8 functions, respectively, though BLPSO-5 performs significantly worse than CMAES, GL-25, GBABC, and JADE on 11, 2, 10, and 18 functions, and it is similar to them on 3, 7, 3, and 4 functions, respectively.

Table 20 shows the ranking of BLPSO-5 and the other EAs according to the Friedman test on the 50-*D* functions. JADE attains the best rank, BLPSO-5 the second, followed by CMAES, GBABC, and GL-25.

Based on the comparison with other EAs on both 30-D and 50-D benchmark functions, it is fair to say that BLPSO5 is a highly competitive algorithm, as it ranks the second, only after JADE.

Table 19 Errors of BLPSO-5 and the other EAs on the 50-*D* functions

		CMAES		GL-25		GBABC		JADE		BLPSO-5
F1	Mean	**4.69E−14**	−	2.14E+06	−	8.82E+06	+	1.43E+04	−	5.10E+06
	SD	**1.19E−14**		2.03E+06		1.85E+06		8.54E+03		1.28E+06
F2	Mean	9.19E−14	−	2.29E+03	=	5.74E+03	=	**7.86E−14**	−	3.44E+03
	SD	3.71E−14		1.17E+03		7.08E+03		**2.44E−14**		2.35E+03
F3	Mean	**1.74E−13**	−	3.41E+02	+	1.35E+03	+	4.27E+03	+	4.23E+01
	SD	**5.77E−14**		4.32E+02		1.02E+03		3.16E+03		8.97E+01
F4	Mean	**1.64E+01**	−	9.54E+01	+	8.75E+01	+	3.84E+01	−	8.64E+01
	SD	**3.73E+01**		1.55E+00		2.68E+01		4.73E+01		5.04E+00
F5	Mean	**2.00E+01**	−	2.12E+01	+	2.06E+01	−	2.04E+01	−	2.09E+01
	SD	**1.23E−06**		2.01E−02		3.93E−02		2.85E−02		5.07E−02
F6	Mean	7.80E+01	+	3.80E+00	+	1.38E+01	+	2.21E+01	+	**9.22E−02**
	SD	1.08E+01		3.26E+00		3.83E+00		4.44E+00		**3.21E−01**
F7	Mean	1.23E−03	+	1.57E−08	+	8.53E−13	+	2.88E−03	=	**2.92E−13**
	SD	3.28E−03		4.57E−08		5.12E−13		4.25E−03		**6.46E−14**
F8	Mean	7.23E+02	+	5.09E+01	+	4.17E−13	−	**3.79E−15**	−	4.97E−01

	SD	1.18E+02		9.19E+00		6.89E−14		**2.08E−14**		8.16E−01
F9	Mean	1.18E+03	+	1.12E+02	=	1.11E+02	+	**5.22E+01**	−	7.10E+01
	SD	2.16E+02		1.06E+02		1.83E+01		**6.50E+00**		9.02E+00
F10	Mean	8.46E+03	+	2.52E+03	+	3.75E+00	−	**1.62E−02**	−	3.63E+02
	SD	7.45E+02		8.26E+02		2.30E+00		**1.23E−02**		1.81E+02
F11	Mean	8.21E+03	+	1.29E+04	+	5.92E+03	+	**4.13E+03**	−	4.46E+03
	SD	7.64E+02		1.51E+03		3.86E+02		**3.51E+02**		4.77E+02
F12	Mean	**2.58E−01**	−	3.81E+00	+	6.96E−01	−	3.46E−01	−	8.77E−01
	SD	**2.12E−01**		2.06E−01		6.71E−02		3.57E−02		1.18E−01
F13	Mean	3.75E−01	+	4.54E−01	+	**2.21E−01**	−	3.36E−01	+	2.86E−01
	SD	5.52E−02		5.48E−02		**3.42E−02**		4.30E−02		3.72E−02
F14	Mean	5.21E−01	+	3.45E−01	+	**1.12E−01**	−	3.04E−01	+	2.65E−01
	SD	2.84E−01		2.89E−02		**1.55E−02**		7.22E−02		2.42E−02
F15	Mean	**6.15E+00**	−	2.29E+01	=	1.27E+01	−	7.56E+00	−	1.48E+01
	SD	**1.26E+00**		1.22E+01		2.73E+00		9.07E−01		1.33E+00
F16	Mean	2.36E+01	+	2.16E+01	+	1.99E+01	+	1.82E+01	=	**1.82E+01**
	SD	5.29E−01		3.34E−01		3.26E−01		3.97E−01		**4.73E−01**
F17	Mean	2.61E+03	−	4.50E+05	−	2.52E+06	+	**2.56E+03**	−	5.97E+05
	SD	5.38E+02		2.06E+05		1.14E+06		**7.91E+02**		2.10E+05
F18	Mean	**2.36E+02**	=	5.04E+02	+	1.21E+03	+	2.89E+02	=	3.73E+02
	SD	**6.16E+01**		2.72E+02		1.08E+03		4.05E+02		3.68E+02
F19	Mean	1.82E+01	=	3.44E+01	+	**1.36E+01**	−	1.40E+01	−	2.16E+01
	SD	2.34E+00		5.67E+00		**2.93E+00**		5.27E+00		9.78E+00
F20	Mean	4.60E+02	+	4.30E+02	+	5.24E+03	+	5.24E+03	+	**2.57E+02**
	SD	1.13E+02		2.34E+02		1.63E+03		6.91E+03		**1.41E+02**
F21	Mean	1.84E+03	−	3.55E+05	=	1.64E+06	+	**1.33E+03**	−	3.80E+05
	SD	4.68E+02		1.56E+05		5.15E+05		**3.98E+02**		1.44E+05
F22	Mean	3.90E+02	+	6.00E+02	+	4.10E+02	+	6.14E+02	+	**2.64E+02**
	SD	2.54E+02		4.25E+02		1.74E+02		1.45E+02		**1.30E+02**
F23	Mean	3.44E+02	+	3.44E+02	+	**3.39E+02**	−	3.44E+02	−	3.44E+02
	SD	2.59E−05		2.07E−09		**2.63E+01**		1.93E−01		2.56E−13
F24	Mean	3.14E+02	+	2.60E+02	+	**2.49E+02**	=	2.74E+02	+	2.58E+02
	SD	1.96E+02		4.97E+00		**1.64E+01**		2.58E+00		4.07E+00
F25	Mean	**2.05E+02**	−	2.18E+02	+	2.17E+02	+	2.15E+02	=	2.10E+02
	SD	**3.63E−01**		2.86E+00		1.85E+00		6.73E+00		7.36E−01
F26	Mean	1.00E+02	−	1.17E+02	=	1.00E+02	−	**1.00E+02**	−	1.47E+02
	SD	8.25E−02		3.78E+01		8.14E−02		**1.03E−01**		5.08E+01
F27	Mean	4.99E+02	+	3.33E+02	=	6.18E+02	+	6.80E+02	+	**3.24E+02**
	SD	8.44E+01		2.79E+01		1.71E+02		1.80E+02		**2.85E+01**
F28	Mean	7.52E+03	+	1.29E+03	+	1.23E+03	+	**8.53E+02**	−	1.14E+03
	SD	5.54E+03		8.01E+01		9.26E+01		**6.83E+01**		4.20E+01
F29	Mean	1.02E+08	+	1.41E+03	=	1.32E+03	=	**2.27E+02**	−	1.36E+03
	SD	5.57E+08		1.22E+02		2.69E+02		**1.59E+00**		1.82E+02
F30	Mean	9.05E+03	=	9.91E+03	+	9.41E+03	+	**1.99E+03**	−	9.09E+03
	SD	7.04E+02		3.17E+02		5.36E+02		**2.22E+03**		3.05E+02
+			16		21		17		8	
=			3		7		3		4	
−			11		2		10		18	

"+", "−", and "=" symbolize a relation that BLPSO-5 is better than, worse than, or similar to its competitor, respectively, according to the Wilcoxon rank sum test at α = 0.05

The best mean error values are marked in bold

Table 20 Ranking of BLPSO-5 and the other EAs according to the Friedman test on the 50-*D* functions

	CMAES	GL-25	GBABC	JADE	BLPSO-5
Friedman rank	2.97	3.85	3.02	2.45	2.72
Final rank	3	5	4	1	2

5 Discussion and conclusions

Comprehensive learning PSO (CLPSO) is a PSO variant whereby particle can learn from personal best information of different particles in different dimensions. CLPSO has shown excellent performance on multimodal problems. However, how to effectively select exemplars for particles still puzzles the user of CLPSO.

In this research, we have proposed a biogeography-based learning PSO (BLPSO) using the migration operator of BBO to select exemplars for particles.

Given a specific problem, the original CLPSO needs to onerously tune two parameters, i.e., the learning probability *Pc* and the refreshing gap *m*. However, in our proposed BLPSO, no parameters have to be tuned, and what needs to be done is just select an appropriate migration model.

It should be mentioned that, although BLPSO-5 (i.e., BLPSO using quadratic migration model) shows the overall best performance on the benchmark functions from CEC2014, BLPSO using the other migration models, such as linear, quadratic, and sinusoidal, also gives competitive performances on some specific functions. Therefore, a possible future work may be to develop a new BLPSO by using a mix of migration models.

We have compared BLPSO-5[4] with well-established algo*rithms, including five PSO* variants, four BBO algorithms, and four non-PSO and non-BBO EAs. The numeric simulations have shown that BLPSO-5 outperforms all the other PSO and the BBO algorithms. Compared with the other four EAs, BLPSO-5 ranks the second, only after JADE. Therefore, it is fair to say BLPSO is a highly competitive optimization algorithm among the state-of-the-art EAs.

As to the intrinsic reason that BLPSO can achieve highly competitive performance, as we know, a good search process needs to balance both exploration and exploitation. It has been widely recognized that while the updating formulas used by the original CLPSO as in Eqs. 2 and 3 facilitate the global exploration, its exploitation ability is not satisfactory. Contrastingly, in our proposed BLPSO, the biogeography-based learning strategy employs a ranking technique whereby particles can learn more from particles with high-quality personal best positions, and this effectively enhances the exploitation of the original CLPSO.

It should be mentioned that in the canonical PSO a three-term updating formula as in Eq. 1 is used for velocity, and each particle learns from its own personal best and the swarm's previous best positions in all dimensions. Differently, in our proposed BLPSO, a two-term updating formula as in Eq. 3 is used for velocity, and the exemplar in each dimension also can be different. While there is quite some theoretical analysis undertaken to reveal how the canonical PSO works (Clerc and Kennedy 2002; Poli 2009), there is little theoretical work on CLPSO and its variants. Therefore, theoretical analysis would be interesting for future work on the proposed BLPSO.

Moreover, it would also be interesting to extend the proposed BLPSO to other types of problems, such as constrained optimization, large-scale optimization, multi-objective optimization, and real-world problems.

Acknowledgments This work was partly supported by the Research Talents Startup Foundation of Jiangsu University (Grant No. 15JDG139), the China Postdoctoral Science Foundation (Grant No. 2016M591783), and the Natural Science Foundation of Jiangsu Province (Grant No. BK20160540). The authors would like to especially thank Dr. Wenyin Gong for his helpful comments on work of this paper. The authors would appreciate the scientific efforts of Dr. N. Hansen, Dr. C. GarciaMartinez, Dr. J. Zhang, and Dr. Y. Jin in making available the source codes of CMAES, GL-25, JADE, and SL-PSO, and Dr. P. N. Suganthan for providing the source codes of CLPSO, DMSPSO, and SaDE.

Appendix 1. Migration models of BBO

Ma (2010) provided six mathematical migration models for BBO. The six migration models can be used to design the biogeography-based exemplar generation method for BLPSO, and they are described as follows:

Model 1 (constant immigration and linear emigration model):

$$\begin{cases} \lambda_k = \dfrac{1}{2} \cdot I \\ \mu_k = \dfrac{k}{N} \cdot E \end{cases} \tag{10}$$

Model 2 (linear immigration and constant emigration model)

[4] The source code of our proposed BLPSO is available from the first author upon request.

$$\begin{cases} \lambda_k = \left(1 - \dfrac{k}{N}\right) \cdot I \\ \mu_k = \dfrac{1}{2} \cdot E \end{cases} \tag{11}$$

Model 3 (linear migration model):

$$\begin{cases} \lambda_k = \left(1 - \dfrac{k}{N}\right) \cdot I \\ \mu_k = \left(\dfrac{k}{N}\right) \cdot E \end{cases} \tag{12}$$

Model 4 (trapezoidal migration model)

$$\begin{cases} \lambda_k = \begin{cases} I, & k \le i' \\ 2\left(1 - \dfrac{k}{N}\right) \cdot I, & i' < k \le ps \end{cases} \\ \mu_k = \begin{cases} \left(\dfrac{2k}{N}\right) \cdot E, & k \le i' \\ E, & i' < k \le ps \end{cases} \end{cases} \tag{13}$$

where $i' = ceil\left((ps+1)/2\right)$

Model 5 (quadratic migration model):

$$\begin{cases} \lambda_k = \left(1 - \dfrac{k}{N}\right)^2 \cdot I \\ \mu_k = \left(\dfrac{k}{N}\right)^2 \cdot E \end{cases} \tag{14}$$

Model 6 (sinusoidal migration model):

$$\begin{cases} \lambda_k = \dfrac{1}{2}\left(\cos\left(\dfrac{k\pi}{N}\right) + 1\right) \cdot I \\ \mu_k = \dfrac{1}{2}\left(-\cos\left(\dfrac{k\pi}{N}\right) + 1\right) \cdot E \end{cases} \tag{15}$$

In Eqs. (10) - (15), I and E are the maximum possible immigration and emigration rates; N is the population size; k is the index of the individual with rank k, where k = 1 refers to the worst individual and $k = N$ refers to the best individual.

References

Alcala-Fdez J, Sanchez L, Garcia S, del Jesus MJ, Ventura S, Garrell J, Otero J, Romero C, Bacardit J, Rivas VM (2009) KEEL: a software tool to assess evolutionary algorithms for data mining problems. Soft Comput 13: 307 - 318

Chen DB, Zhao CX (2009) Particle swarm optimization with adaptive population size and its application. Appl Soft Comput 9: 39 - 48

Cheng R, Jin Y (2015a) A competitive swarm optimizer for large scale optimization. Cybern IEEE Trans 45: 191 - 204

Cheng R, Jin Y (2015b) A social learning particle swarm optimization algorithm for scalable optimization. Inf Sci 291: 43 - 60

Clerc M, Kennedy J (2002) The particle swarm-explosion, stability, and convergence in a multidimensional complex space. Evol Comput IEEE Trans 6: 58 - 73

Eberchart RC, Kennedy J (1995) Particle swarm optimization. In: Proceedings of IEEE International Conference on Neural Network. Piscataway: IEEE Press, pp. 1942-1948

Eberhart RC, Kennedy J (1995) A new optimizer using particle swarm theory. In: Proceedings of the sixth international symposium on micro machine and human science. New York, NY, pp. 39-43

Epitropakis MG, Plagianakos VP, Vrahatis MN (2012) Evolving cognitive and social experience in particle swarm optimization through differential evolution: a hybrid approach. Inf Sci 216: 50 - 92

Fang W, Sun J, Chen H, Wu X (2016) A decentralized quantum-inspired particle swarm optimization algorithm with cellular structured population. Inf Sci 330: 19 - 48

Garcia-Martinez C, Lozano M, Herrera F, Molina D, Sánchez AM (2008) Global and local real-coded genetic algorithms based on parent-centric crossover operators. Eur J Oper Res 185: 1088 - 1113

Gong W, Cai Z, Ling CX (2010a) DE/BBO: a hybrid differential evolution with biogeography-based optimization for global numerical optimization. Soft Comput 15: 645 - 665

Gong W, Cai Z, Ling CX, Li H (2010b) A real-coded biogeographybased optimization with mutation. Appl Math Comput 216: 2749 - 2758

Gong Y-J, Li J-J, Zhou Y, Li Y, Chung HS-H, Shi Y-H, Zhang J (2015) Genetic learning particle swarm optimization. IEEE Trans Cybern. doi: 10.1109/TCYB.2015.2475174

Gulcu S, Kodaz H (2015) A novel parallel multi-swarm algorithm based on comprehensive learning particle swarm optimization. Eng Appl Artif Intell 45: 33 - 45

Hansen N, Ostermeier A (2001) Completely derandomized self- adaptation in evolution strategies. Evol Comput 9: 159 - 195

Hu M, Wu T-F, Weir JD (2013) An adaptive particle swarm optimization with multiple adaptive methods. Evol Comput IEEE Trans 17: 705 - 720

Huang VL, Suganthan PN, Liang JJ (2006) Comprehensive learning particle swarm optimizer for solving multiobjective optimization problems. Int J Intell Syst 21: 209 - 226

Kennedy J (1999) Small worlds and mega-minds: effects of neighborhood topology on particle swarm performance. Evolutionary Computation, 1999. CEC 99.In: Proceedings of the 1999 Congress on. IEEE

Li C, Yang S, Nguyen TT (2012) A self-learning particle swarm optimizer for global optimization problems. Syst Man Cybern Part B Cybern IEEE Trans 42: 627 - 646

Li X, Wang J, Zhou J, Yin M (2011) A perturb biogeography based optimization with mutation for global numerical optimization. Appl Math Comput 218: 598 - 609

Liang J, Qu B, Suganthan P (2013) Problem definitions and evaluation criteria for the CEC 2014 special session and competition on single objective real-parameter numerical optimization. Zhengzhou University, Zhengzhou China and Technical Report, Nanyang Technological University, Singapore, Computational Intelligence Laboratory

Liang JJ, Qin AK, Suganthan PN, Baskar S (2006) Comprehensive learning particle swarm optimizer for global optimization of multimodal functions. Evol Comput IEEE Trans 10: 281 - 295

Liang JJ, Suganthan PN (2005) Dynamic multi-swarm particle swarm optimizer. Swarm Intelligence Symposium, 2005. SIS 2005.In: Proceedings 2005 IEEE. IEEE, pp. 124 - 129

Lim WH, Isa NAM (2014a) Particle swarm optimization with adaptive time-varying topology connectivity. Appl Soft Comput 24: 623 - 642

Lim WH, Isa NAM (2014b) Teaching and peer-learning particle swarm optimization. Appl Soft Comput 18: 39 - 58

Lynn N, Suganthan PN (2015) Heterogeneous comprehensive learning particle swarm optimization with enhanced exploration and exploitation. Swarm Evol Comput 24: 11 - 24

Ma H (2010) An analysis of the equilibrium of migration models for biogeography-based optimization. Inf Sci 180: 3444 - 3464

Mendes R, Kennedy J, Neves J (2004) The fully informed particle swarm: simpler, maybe better. Evol Comput IEEE Trans 8: 204 - 210

Nickabadi A, Ebadzadeh MM, Safabakhsh R (2011) A novel particle swarm optimization algorithm with adaptive inertia weight. Appl Soft Comput 11: 3658 - 3670

Ouyang HB, Gao LQ, Kong XY, Li S, Zou DX (2016) Hybrid harmony search particle swarm optimization with global dimension selection. Inf Sci 346: 318 - 337

Parsopoulos KE, Vrahatis MN (2004) UPSO: a unified particle swarm optimization scheme. Lect Ser Comput Comput Sci 1: 868 - 873

Poli R (2009) Mean and variance of the sampling distribution of particle swarm optimizers during stagnation. IEEE Trans Evol Comput 13: 712 - 721

Qin Q, Cheng S, Zhang Q, Li L (2015) Particle swarm optimization with interswarm interactive learning strategy. Cybern IEEE Trans. doi: 10.1109/TCYB.2015.2474153

Robinson J, Sinton S, Rahmat-Samii Y (2002) Particle swarm, genetic algorithm, and their hybrids: optimization of a profiled corrugated horn antenna. AP-S Int Symp (Dig) (IEEE Antennas Propag Soc) 1: 314 - 317

Sheng-Ta H, Tsung-Ying S, Chan-Cheng L, Shang-Jeng T (2009) Efficient population utilization strategy for particle swarm optimizer. Syst Man Cybern Part B Cybern IEEE Trans 39: 444 - 456

Shi Y, Eberhart R (1998) A modified particle swarm optimizer, Evolutionary Computation Proceedings, 1998. IEEE World Congress on Computational Intelligence., The 1998 IEEE International Conference on. IEEE, pp. 69 - 73

Shi Y, Eberhart RC (2001) Fuzzy adaptive particle swarm optimization, Evolutionary Computation, (2001). In: Proceedings of the 2001 Congress on. IEEE, pp. 101 - 106

Simon D (2008) Biogeography-based optimization. Evol Comput IEEE Trans 12: 702 - 713

Suganthan PN, Hansen N, Liang JJ, Deb K, Chen Y-P, Auger A, Tiwari S (2005) Problem definitions and evaluation criteria for the CEC 2005 special session on real-parameter optimization. KanGAL report 2005005

Yu X, Zhang X (2014) Enhanced comprehensive learning particle swarm optimization. Appl Math Comput 242: 265 - 276

Zhan Z-H, Zhang J, Li Y, Shi Y-H (2011) Orthogonal learning particle swarm optimization. Evol Comput IEEE Trans 15: 832 - 847

Zhang J, Sanderson AC (2009) JADE: adaptive differential evolution with optional external archive. Evol Comput IEEE Trans 13: 945 - 958

Zhou X, Wu Z, Wang H, Rahnamayan S (2014) Gaussian bare-bones artificial bee colony algorithm. Soft Comput 20: 1 - 18