Information Sciences 376 (2017) 71–94

Contents lists available at ScienceDirect

Information Sciences

journal homepage: www.elsevier.com/locate/ins

Learning backtracking search optimisation algorithm and its application

Debao Chen [a,*], Feng Zou [a], Renquan Lu [b,*], Peng Wang [a]

[a] School of Physics and Electronic Information, Huaibei Normal University, Huaibei 235000, China
[b] School of Automation, Guangdong University of Technology, Guangzhou, 510006, China

ARTICLE INFO

Article history:
Received 17 January 2016
Revised 21 September 2016
Accepted 2 October 2016
Available online 5 October 2016

Keywords:
Evolutionary algorithm
backtracking search algorithm
Teaching-learning–based optimisation
Artificial neural network training
Optimisation problems

ABSTRACT

The backtracking search algorithm (BSA) is a recently proposed evolutionary algorithm (EA) that has been used for solving optimisation problems. The structure of the algorithm is simple and has only a single control parameter that should be determined. To improve the convergence performance and extend its application domain, a new algorithm called the learning BSA (LBSA) is proposed in this paper. In this method, the globally best information of the current generation and historical information in the BSA are combined to renew individuals according to a random probability, and the remaining individuals have their positions renewed by learning knowledge from the best individual, the worst individual, and another random individual of the current generation. There are two main advantages of the algorithm. First, some individuals update their positions with the guidance of the best individual (the teacher), which makes the convergence faster, and second, learning from different individuals, especially when avoiding the worst individual, increases the diversity of the population. To test the performance of the LBSA, benchmark functions in CEC2005 and CEC2014 were tested, and the algorithm was also used to train artificial neural networks for chaotic time series prediction and nonlinear system modelling problems. To evaluate the performance of LBSA with some other EAs, several comparisons between LBSA and other classical algorithms were conducted. The results indicate that LBSA performs well with respect to other algorithms and improves the performance of BSA.

1. Introduction

For some science and engineering problems that can be converted to optimisation problems, the search for and design of better optimisation algorithms has never ceased. To solve complex problems such as nonlinear, non-differentiable, and non-convex objective function problems, researchers have focused on designing novel evolutionary algorithms (EAs) or improving the performance of existing algorithms. Of these algorithms, swarm intelligence optimisation algorithms and genetic evolution algorithms play very important roles in the solution of complex optimisation problems. The particle swarm optimisation (PSO) algorithm [12], which mimics the foraging law of birds, has been successfully used in function optimisation [37], sequence classification problems [36], and power output systems [25]. Some variants of PSO, such as the fully informed particle swarm (PSOFIPS) [20], fitness-distance-ratio-based PSO (PSOFDR) [23], and comprehensive learning particle swarm (CLPSO) [15], have also been proposed to improve the performance of PSO and solve optimisation problems. Moreover, some

* Corresponding authors.
E-mail address: chendb_8@163.com (D. Chen).

http://dx.doi.org/10.1016/j.ins.2016.10.002

hybrid algorithms are also introduced to improve the performance of PSO [13,21], and some discrete methods are proposed to extend the application domain of PSO [2,8,10]. Simulating the teaching-learning process in a class, the teaching-learning–based optimisation (TLBO) algorithm [31] has also been proposed. There are fewer parameters that must be determined in its updating equations, and the algorithm is easily implemented. To make full use of the advantages of TLBO, two approaches have been taken: one is to improve the efficiency of the algorithm by modifying the updating process or combining it with other EAs, and the other is to extend its application domains. The concept of elitism has been utilised in elitism TLBO (ELTBO) to improve its performance by replacing some worst learners in the class by elitisms [29]. A self-learning TLBO that designs a new self-learning method has been used to train the radial basis function (RBF) neural modelling of batteries [41]. Many teachers, an adaptive teaching factor, tutorial training, and self-motivated learning have all been used to improve the diversity of the population [30]. These modified TLBO algorithms perform well for some optimisation problems. With respect to the application fields, the Lévy mutation operator [7] has been used to help TLBO avoid local optima, and the new algorithm has successfully been used for the IEEE 30-bus test system. An improved TLBO [6] with direction angle has been introduced to increase the searching ability of its learners, and it has been combined with a double differential evolution algorithm to handle the optimal reactive power dispatch (ORPD) problem. Local and self-learning methods [3] have been used to improve the performance of the original TLBO, and these approaches display some good performance for global optimisation problems. The multi-objective teaching-learning optimiser [19] has been used for handling reactive power systems. The optimisation of plate-fin heat exchangers [22] and modern machining processes [28], short-term hydrothermal scheduling problems [32], two-stage thermoelectric coolers [27], and flow-shop rescheduling problems [11,14] have all been solved by TLBO. A detailed review of applications of TLBO can be found in [26]. Simplifying the training process and decreasing the number of undetermined parameters of intelligence optimisation algorithms are very important issues. To address them, a novel algorithm called the backtracking search algorithm (BSA) [5] was proposed in 2013 and used for some engineering optimisation problems. BSA has been used for surface wave analysis [34] and to solve concentric circular antenna array synthesis optimisation problems [9]. BSA and differential evolution algorithm (DE) have also been combined to solve unconstrained optimisation problems [40]. The oppositional BSA [17] was introduced to solve hyper-chaotic system parameter identification optimisation problems. The BSA with three constraint handling methods [43] was used for solving constrained optimisation problems. Compared to some classical EAs, BSA is a very new optimisation algorithm, and the improved variants of BSA are relatively few.

As mentioned above, BSA and TLBO have both shown to be superior at solving some optimisation problems, especially because the algorithms are simple and there are few undetermined parameters in the individual updating equations. As young algorithms, their performance at solving complex problems needs to be improved. Their disadvantages are detailed in Sections 2 and 3.

To improve the global performance of BSA, a new method that combines the ideas of TLBO and BSA is presented in this paper. The main contributions of the new method are as follows. The first contribution is the addition of learning guidance in the BSA updating equations to improve convergence speed of BSA. The second is the integration of three learning methods into one equation to update parts of individuals according to a random probability. Another contribution is the avoidance of the use of the worst individual in order to improve the diversity of the population. In contrast to the two function evaluations (FEs) for an individual in a generation of the TLBO algorithm, there is only one FE per generation in the new method. Hence, the computation cost for a generation of the proposed algorithm is less than that of TLBO.

The rest of the paper is organised as follows: the basic BSA and original TLBO are briefly introduced in Section 2 and Section 3, respectively; LBSA is presented in Section 4; the benchmark functions in CEC2005 and CEC2014 are tested to show the effectiveness of different algorithms in Section 5; the applications for two typical nonlinear modelling problems and chaotic time series prediction problems are displayed in Section 6; and conclusions are given in Section 7.

2. Basic BSA

The BSA is a population-based EA that is developed from a DE algorithm, but it is not the same as DE. It contains five processes: initialisation, selection-I, mutation, crossover, and selection-II. There are two populations in BSA: one is the evolution population and the other is the trial population. The trial population is composed of some historical information regarding the evolution population, and a search-direction matrix is built by the two populations to update the positions of the individuals. There is only one control parameter, namely the *mix rate*, which controls the number of individual elements that will be mutated in a trial. The five steps of BSA are simply described as follows, and further details can be found in [5].

(1) **Initialisation.** The initial population and history population of BSA are initialised according to Eq. 1 and 2, where U is the uniform distribution, and low_j and up_j are the low and up boundaries of variables.

$$P_{i,j} \sim U(low_j, up_j) \tag{1}$$

$$oldP_{i,j} \sim U(low_j, up_j) \tag{2}$$

where i is the ith individual of the population, $i=1, 2, \cdots, N$, N is the population size, j is the jth bit of the ith individual, $j=1, 2, \cdots, D$, and D is the dimension size of variables.

(2) **Selection-I.** In the beginning of each iteration, the history population *oldP* is introduced in BSA according to Eq. 3 and 4:

$$oldP = \begin{cases} P & if \quad (a < b | a, b \sim U(0, 1)) \\ oldP & otherwise \end{cases} \tag{3}$$

$$oldP := permuting(oldP) \tag{4}$$

(3) **Mutation.** The initial form of the trial population of BSA is generated by mutation operation according to Eq. (5). As shown in Eq. (5), BSA takes advantage of its experiences from previous generations to generate a trial population; the historical information of the population penetrates the whole evolutionary process. F controls the amplitude of the search-direction matrix. The common value of F is equal to $3 \bullet randn$, where $randn \sim N(0, 1)$.

$$M = P + F \cdot (oldP - P) \tag{5}$$

(4) **Crossover.** In BSA, the initial value of the trial population comes from the mutation process, and the trial individuals with better fitness are used to evolve the target population individuals. A binary integer-valued matrix (*map*) of size $N \bullet D$ guides the crossover directions of BSA algorithms. The crossover strategy of BSA is expressed in Eq. 6:

$$V_{i,j} = \begin{cases} P_{i,j} & if \quad map_{i,j} = 1 \\ M_{i,j} & otherwise \end{cases} \tag{6}$$

(5) **Selection-II**. At this step, the population of the next generation is generated according to a greedy selection mechanism. In the proposed method, for minimal problems, if the fitness of V_i is smaller than that of P_i, then P_i is replaced by V_i. This process is shown in Eq. 7:

$$P_i^{next} = \begin{cases} V_i & if \quad f(V_i) \le f(P_i) \\ P_i & otherwise \end{cases} \tag{7}$$

As mentioned in the introduction, BSA is an algorithm with a simpler structure than some EAs, such as PSOs, genetic algorithms, and DEs. The crossover and mutation mechanisms are different from those of DE and its variants. The historical information is used in the updating process to improve the performance of the algorithm. However, as a young algorithm, BSA also has some disadvantages. The first is that there is no guidance as to the best individual in the updating process, which can slow the convergence speed of BSA. The second is that individuals cannot be renewed when the diversity of the population is lost, because the trial population may not be changed when the population is almost the same in a later generation. This decreases the ability of BSA to avoid local optima.

3. Main steps of TLBO

TLBO is also a population-based EA that mimics the philosophy of teaching and learning in a class. There are two main phases in TLBO: the teacher phase and the learner phase. Because only the learning ideas of TLBO are used in the proposed learning BSA (LBSA), only the two main phases of the TLBO algorithm are described in this section.

3.1. Teacher phase

In the teacher phase, the teacher distributes its knowledge for all learners in the class, and the learners update their positions according to the teacher and the mean position of the current class. The learner with the best fitness is chosen to be the teacher. For a D-dimensional optimisation problem, let the position of the ith learner be $X_i = \{x_{i,1}, x_{i,2}, \ldots, x_{i,D}\}$, where X_{mean} denotes the mean solution of the current class, and $X_{teacher}$ denotes the position of the teacher. The ith learner updates its position according to the difference between the teacher and mean position as follows:

$$X_{new,i} = X_{old,i} + rand(.) * (X_{teacher} - T_F X_{mean}) \tag{8}$$

where $X_{new,i}$ and $X_{old,i}$ are the new and old positions, respectively, of the ith learner, and $rand(.)$ is a random number in the range [0,1]. The greedy selection mechanism is used to accept the better of $X_{new,i}$ and $X_{old,i}$. The accepted position then flows to the learner phase. Teaching factor T_F determines the value of the mean to be changed. The value of T_F is heuristically set to either 1 or 2. It is decided randomly with equal probability as follows:

$$T_F = round[1 + rand(0, 1)\{2 - 1\}] \tag{9}$$

3.2. Learner phase

At any iteration, the kth learner is randomly selected as the learning object of the ith learner. The method for the learning of the ith learner can be mathematically expressed as follows:

$$X_{new,i} = \begin{cases} X_{old,i} + rand(.)(X_{old,i} - X_{old,k}) & if \ f(X_{old,i}) < f(X_{old,k}) \\ X_{old,i} + rand(.)(X_{old,k} - X_{old,i}) & otherwise \end{cases} \tag{10}$$

where $X_{new,\, i}$ is the new position of the *i*th individual, and $X_{old,\, i}$ and $X_{old,\, k}$ are the old positions of the *i*th and *k*th learners, respectively. In addition, $X_{new,\, i}$ is accepted if the fitness of $X_{new,\, i}$ is better than that of $X_{old,\, i}$.

The structure of TLBO is simple, and there are no parameters that should be predetermined in the updating equations. It is hence widely used for solving certain optimisations. However, it also has two disadvantages. The first is that there are two FEs for each individual in each generation, and hence the computation cost of a single generation is larger than that of an algorithm with one FE per individual per generation. The second is that learning knowledge from the better learners and the two greedy selections of an individual might reduce the diversity of the population too quickly.

To improve the performance of BSA, the learning mechanism of TLBO and a repulsion operator to improve the diversity of the population are proposed in this study. Furthermore, there is only one FE of each individual in a generation, as detailed in the next section.

4. LBSA

4.1. Motivations

The main motivation behind LBSA is to fuse the learning ideas of TLBO into basic BSA to improve its global performance. In the basic BSA algorithm, when the diversity of the population becomes poor in the anaphase of evolution, the evolution ability of the individuals becomes weak. This phenomenon restricts the individual from finding the global optimum. Moreover, practice has shown that learning operators can improve the adaptability of the population, and learning from the best individual can improve the convergence speed of the algorithm and increase the exploitation capability of the algorithm. However, the diversity of the population and the exploration capability of the algorithm will be decreased with the rapid convergence speed. How to increase the exploration ability of the algorithm through improving the diversity of the population by modifying the updating process of individuals is very important for improving the overall properties of the optimisation algorithm. Hence, the proposed LBSA improves the global performance of basic BSA by modifying its mutation processes.

4.2. LBSA mutation

In basic BSA, only the historical information and the current position of an individual are used to generate a new position. The individuals cannot learn knowledge from the teacher of the whole class. Some EA methods have shown that tracking the best individual may increase the convergence speed of the algorithm. To improve the learning ability of BSA, the learning guidance of the best individual is introduced in the mutation process of the algorithm as follows:

$$M = P + F \cdot (0.5^{*}(oldP - P) + 0.5^{*}rand(.)^{*}(Teacher - P)) \tag{11}$$

where *Teacher* is the best individual of the current generation, $F{=}3{*}rand$, and the meanings of the other parameters are the same as those in basic BSA.

The learning methods of TLBO were modified before they were applied in LBSA. The new method contains three parts. First, each individual learns knowledge from a random individual from the current population, which is the same as it is in TLBO. Second, each individual learns knowledge from the best individual, which is different from TLBO because the mean position of the current population is not used. Moreover, to increase the diversity of the generated individuals, avoidance of the worst individual of the current generation is designed into the updating equation. In contrast to TLBO, the three methods in LBSA are integrated into one equation, and each individual only executes one FE per generation. The detailed process is as follows:

$$\begin{cases} M_{i,j} = P_{i,j} + rand^{*}(P_{i,j} - P_{k,j}) + rand^{*}(Teacher_{1,j} - P_{i,j}) - rand^{*}(Worst_{1,j} - P_{i,j}) \\ \qquad\qquad if \quad ith\ individual\ is\ better\ than\ kth\ individual \\ M_{i,j} = P_{i,j} + rand^{*}(P_{k,j} - P_{i,j}) + rand^{*}(Teacher_{1,j} - P_{i,j}) - rand^{*}(Worst_{1,j} - P_{i,j}) \\ \qquad\qquad otherwise \end{cases} \tag{12}$$

where $P_{i,\,j}$ is the *j*th bit of the *i*th individual, $i{=}1, 2, \cdots, N, j{=}1, 2, \cdots, D$, and $P_{k,\,j}$ is the value of the *j*th bit of the *k*th individual, which is randomly chosen from the population and is different from the *i*th individual. $M_{i,\,j}$ is the value of the *j*th bit of the *i*th individual after the mutation operator is applied. In addition, *Teacher* and *Worst*are the best and worst positions of the current generation, respectively. We note that Eqs. 11 and 12 are randomly chosen in the LBSA mutation process. The pseudo code for LBSA is shown in Algorithm 1.

4.3. Diversity analysis of BSA and LBSA

To verify that the diversity of basic BSA is improved with the modified operators, functions F5, F6, F9, and F15 in CEC2005 were used to test the change of diversities as the generations increased. There are many measurements of diversity, such as SPD, HPD [38], the methods based on the average Hamming distance of all individuals, the distance between the positions of all individuals and the centre position of the population, and entropy [4]. To simply and directly analyse the diversity of BSA and LBSA, the simple method based on average Hamming distance of all individuals was chosen in this study. To fairly

Algorithm 1 . Pseudo code of LBSA ().

```
Begin
 Initialize N (number of population), D (dimension), maxFEs, maxgen, mixrate, low_{1:D}, up_{1:D};
% Initialize population;
 for i = 1:N
  for j = 1:D
  P_{i,j} = low_j + rand*(up_j - low_j);
  old P_{i,j} = low_j + rand*(up_j - low_j);
  endfor
    endfor
   Evaluate all individuals;
    memorize the best (Teacher) and the worst individual (worst);
    For gen = 1:maxgen
%   Selection-I
      if (a < b | a, b ~ U(0, 1)) then oldP=P end;
        oldP= randperm (oldP); % 'permuting' arbitrary changes in positions of two individuals in oldP
%   Generate trial population
%   Mutation
    for i = 1:N
      if (a < b | a, b ~ U(0, 1))
           the individual is mutated by Eq. 11, generate partly individuals of trial population T;
      else
           the individual is mutated by Eq. 12, generate the remainder individuals of trial population T;
      end
   endfor
%  Crossover
   map_{1:n,1:d} = 1; %Initial map matrix
   if (a < b | a, b ~ U(0, 1))
      for i = 1:N
      u = randperm(D);
     map_{i,u(1: ceil(mixrate*rand*D))} = 0;
  endfor
  else
      for i = 1:N
         map_{i,randi(D)} = 0;
   endfor
  end
  for i = 1:N
   for j = 1:D
      if map_{i,j} = 1
        T_{i,j} = P_{i,j};
   end
  endfor
    endfor
%   Boundary control mechanism
   for i = 1:N
   for j = 1:D
         if (T_{i,j} < low_j) or (T_{i,j} > up_j)
           T_{i,j} = low_j + rand*(up_j - low_j);
           end
           endfor
    endfor
%            Selection-II
    Evaluate trial population T;
    for i = 1:N
     if T_i is better than P_i
           P_i = T_i;
               fitness(P_i)=fitness(T_i);
      end
    endfor
      renew Teacher and worst
      output the best solution min(fitness(P))
   endFor
```

compare the diversities of BSA and LBSA algorithms, the distance between two individuals is normalised in the range [0,1]. The definition of diversity of the population is as follows.

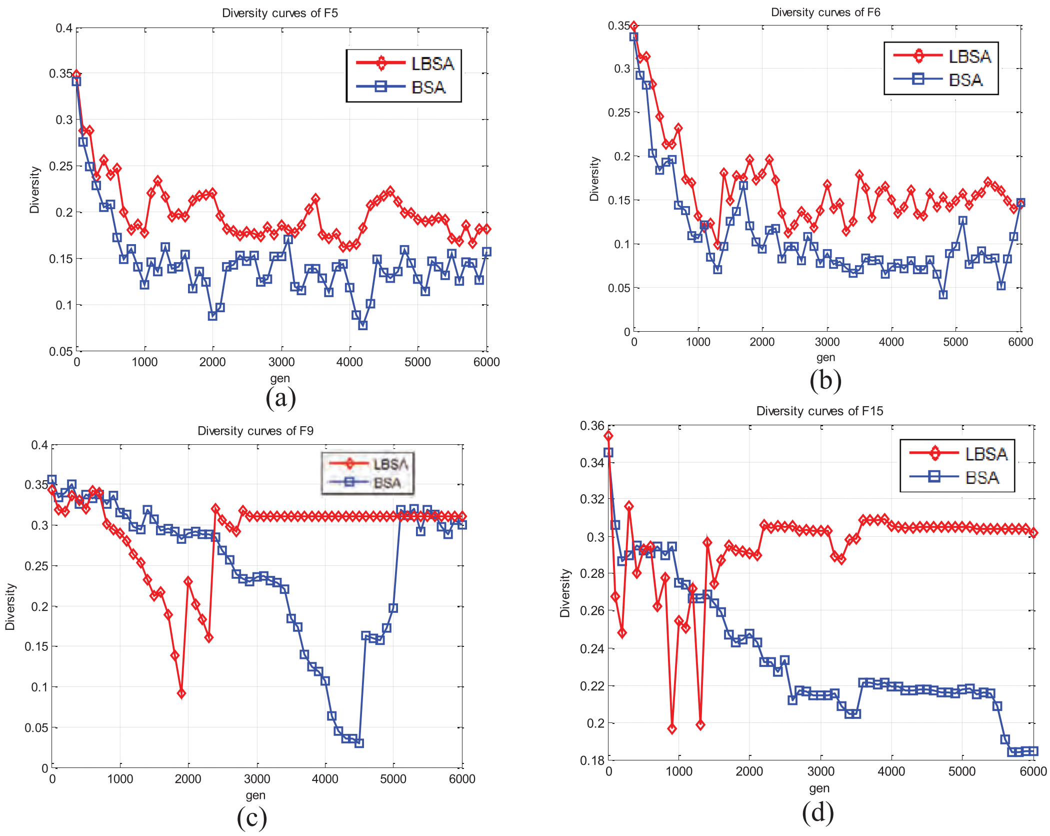

Fig. 1. The diversties curve of LBSA and BSA.

Definition 1. The Hamming distance between the *i*th individual (p_i) and *l*th individual (p_l) is expressed in Eq. 13:

$$H(p_i, p_l) = \sum_{j=1}^{D} \left| p_{i,j} - p_{l,j} \right| \tag{13}$$

Definition 2. The normalised distance between the *i*th individual (p_i) and *l*th individual (p_l) is expressed in Eq. 14:

$$H^*(p_i, p_l) = H(p_i, p_l) / \max\{H(p_m, p_n), m = 1, 2, \cdots, N-1; n = m+1, m+2, \cdots, N\} \tag{14}$$

Definition 3. The average Hamming distance of the population at each iteration is shown in Eq. 15:

$$\bar{H}(T) = \frac{2\sum_{i=1}^{N-1}\sum_{j=i+1}^{N} H^*(p_i, p_l)}{N*(N-1)} \tag{15}$$

The training parameters were as follows: the population size was 50, the termination condition of the algorithm was the maximum number of iterations, the value of iteration was 6000, the dimension of the functions was 30, and the two algorithms were tested over 30 independent runs. The change of average diversity is shown in Fig. 1, which indicates that the average diversity of LBSA is better than that of basic BSA for most of the iterations. This helps LBSA easily escape from local optima.

Fig. 1(a) and (b) display that the diversities of LBSA are almost better than those of BSA, and the diversities of the two algorithms are vibrant in all evolutions. Fig. 1(c) and (d) indicate that the diversity of LBSA is worse than that of BSA in the beginning phase of evolution, but the opposite results are derived in the anaphase of evolution. Fig. 1(c) and (d) also display that the diversities of the algorithms are almost unchanged in the end of the evolution for the algorithms that have arrived at the local or global optima.

5. Simulation experiments

To test the performance of the LBSA, three benchmark tests were conducted. To illustrate the effectiveness of the proposed algorithm, three PSO variants (PSOFIPS [20], PSOFDR [23], and CLPSO [15]), two DE variants (jDE [1] and SaDE [24]), three recently proposed TLBO variants (TLBO [31], ETLBO [29], and DGSTLBO [45]), and basic BSA [5] were used as comparison algorithms. The included algorithms are more or less related to LBSA, such as the learning ideas in PSO and TLBO, historical information in basic BSA, and crossover operator in DE. Hence, these algorithms or their variants were chosen for comparison. To indicate the effectiveness of the proposed algorithm, CEC2005 [35] and CEC2014 [16,18] were used as the benchmark function problems. The training of artificial neural networks (ANNs) for time series prediction and nonlinear system modelling was used as the application problems.

5.1. Experimental setup

The detailed information regarding 25 benchmark problems used in CEC2005 is provided in [35]. The 30 benchmark problems in CEC2014 are detailed in [16]. Nonlinear single input–single output (SISO) and multiple input–single output (MISO) systems [39] and two classical time series prediction problems (the Mackey-Glass chaotic time series and Box-Jenkins chaotic time series [33,44]) were predicted in the experiments. The prediction model was multi-layered perceptron (MLP), and its parameters were trained by different EAs.

5.2. Parameter settings

All the experiments were carried out on the same machine using MATLAB 2012a. Each algorithm was independently simulated for 30 runs. For all algorithms, the population size was set to 50. The maximum FE was used as the stopping criterion for all algorithms. For CEC2005 and CEC2014, the maximum FE was set to 5000D (D is the dimension variable). In addition, the 10D and 30D functions of CEC2005 were simulated, and the 30D functions of CEC2014 were tested. Some parameters of other comparison algorithms are listed as follows:

- PSOFIPS [20]: $w = 0.7298$;
- PSOFDR [23]: $wmin = 0.4$, $wmax = 0.9$, $\psi 1 = 1$, $\psi 2 = 1$, $\psi 3 = 2$;
- CLPSO [15]: $cc = [1.49445\ 1.49445]$; $iwt = 0.9 - (1{:}\ maxgen) * (0.7\ /\ maxgen)$

$$P_{c_i} = 0.05 + 0.45^* \frac{\exp(\frac{10(i-1)}{N-1}) - 1}{\exp(10) - 1} \qquad (N \text{ is the population size})$$

- jDE [1]: $F = 0.5$, $CR = 0.9$;
- SaDE [24]: $F \sim N(0.5, 0.3)$, $CR0 = 0.5$, $CR \sim N(CRm, 0.1)$, $LP = 50$;
- ETLBO [29]: elite size = 2
- DGSTLBO[45]: m (group size) = 5, p (regroup period) = 5, Pc (learning probability) = 5;

5.3. CEC2005 experiments

5.3.1. Comparison of 10D function solution accuracy in CEC2005

The simulation results for the 10D functions in CEC2005 are shown in Table 1. The results show that with respect to the mean, LBSA outperformed the other nine algorithms for functions F9, F10, F15, F22, and F23. jDE outperformed the other nine algorithms for functions F3, F6, F7, and F17. For function F3, the mean of jDE was closer to the theoretical optimal solution, and the standard deviation (Std.) of jDE was also the smallest among all algorithms. LBSA could not converge to the theoretical optimal solution in all 30 runs, and its mean and standard deviation were worse than those of jDE and SaDE. The rank of LBSA in terms of mean was 3. SaDE outperformed the other nine algorithms in terms of mean for the two functions F12 and F16. The means of PSOFIPS were better than those of the other algorithms for functions F21 and F25. PSOFDR outperformed other algorithms in terms of mean for functions F8, F11, and F14. BSA was better than the other algorithms for functions F13, F18, and F20. For function F20, the best solutions of PSOFIPS,TLBO, BSA, and LBSA were the same as the theoretical optimal solution; that is, there existed global optima in 30 runs of these three algorithms. For function F1, all algorithms converged to the global optimum, except for PSOFDR. All ten algorithms nearly found the global optimum of function F2. For function F4, PSOFIPS and CLPSO could not converge to the global optimum. For function F5, three algorithms could not find the global optimum. The table also shows that LBSA ranked in first place for 10 functions in terms of mean for 25 functions, which is higher than that of the other nine algorithms. The average rank of LBSA was also the best of the ten algorithms for CEC2005. Moreover, there was a phenomenon that the mean best solution and the standard deviation of an algorithm have opposite character. For example, the standard deviation of LBSA was the smallest but the mean value was not the best for function F16, and the mean best solution of LBSA was relatively better than those of the other eight algorithms but the standard deviation was not the best for function F20. The reason for this phenomenon is explained as follows. For a multimodal problem with five extremum points in Fig. 2, algorithms A and B are used to find the global optima with m independent runs. If algorithm A converges to point 2 in m runs, algorithm B converges to point 4 with m-1 runs and converges to point 5 with one run. The mean best solution of algorithm B will be better than that of algorithm A, but the standard deviation of algorithm A might be smaller than that of algorithm B. The performance in terms of mean best solution of algorithm B is generally better than that of algorithm A. The bold values in Table 3 are the

Table 1
Comparative results of ten algorithms for 10D problems from CEC2005.

F	AL.	PSOFIPS	PSOFDR	CLPSO	jDE	SaDE	TLBO	ETLBO	DGSTLBO	BSA	LBSA
F1	mean	**−450.000**	−439.674	**−450.000**	**−450.000**	**−450.000**	**−450.000**	**−450.000**	**−450.000**	**−450.000**	**−450.000**
	std.	**0.000**	32.653	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
	best	**−450.000**	**−450.000**	**−450.000**	**−450.000**	**−450.000**	**−450.000**	**−450.000**	**−450.000**	**−450.000**	**−450.000**
	Rank	1	2	1	1	1	1	1	1	1	1
F2	mean	**−450.000**	**−450.000**	−449.676	**−450.000**	**−450.000**	**−450.000**	**−450.000**	**−450.000**	−449.997	**−450.000**
	std.	**0.000**	**0.000**	0.177	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	0.005	**0.000**
	best	**−450.000**	**−450.000**	−449.906	**−450.000**	**−450.000**	**−450.000**	**−450.000**	**−450.000**	**−450.000**	**−450.000**
	Rank	1	1	3	1	1	1	1	1	2	1
F3	mean	114,431.841	143,065.100	695,430.984	**−237.160**	546.821	70,188.530	132,295.917	55,887.005	21,518.124	5652.945
	std.	54,898.334	156,627.508	263,113.140	**363.050**	2552.898	47,419.217	98,983.396	82,010.530	18,993.641	7686.985
	best	48,691.653	6576.186	333,184.289	**−450.000**	−449.996	22,503.152	23,199.940	7846.953	3316.003	251.786
	Rank	7	9	10	1	2	6	8	5	4	3
F4	mean	**−450.000**	−438.829	−445.693	**−450.000**	**−450.000**	**−450.000**	**−450.000**	**−450.000**	−449.800	**−450.000**
	std.	**0.000**	23.550	2.683	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	0.306	**0.000**
	best	**−450.000**	**−450.000**	−449.361	**−450.000**	**−450.000**	**−450.000**	**−450.000**	**−450.000**	−449.994	**−450.000**
	Rank	1	4	3	1	1	1	1	1	2	1
F5	mean	−233.009	**−310.000**	**−310.000**	−299.093	**−310.000**	**−310.000**	**−310.000**	−152.429	**−310.000**	**−310.000**
	std.	33.727	**0.000**	**0.000**	15.577	**0.000**	**0.000**	**0.000**	261.668	**0.000**	**0.000**
	best	−289.340	**−310.000**	**−310.000**	**−310.000**	**−310.000**	**−310.000**	**−310.000**	**−310.000**	**−310.000**	**−310.000**
	Rank	3	1	1	2	1	1	1	4	1	1
F6	mean	394.658	399.415	391.599	**390.399**	390.848	392.592	392.167	395.307	392.105	390.548
	std.	**0.405**	18.471	1.610	1.261	1.529	3.041	2.313	2.735	2.741	1.112
	best	393.841	390.002	390.348	**390.000**	**390.000**	390.000	390.000	390.179	390.091	390.001
	Rank	8	10	4	1	3	7	6	9	5	2
F7	mean	−179.810	1087.288	1087.208	**−179.965**	1087.046	1087.046	1087.046	−167.902	1087.082	1087.046
	std.	0.098	0.197	0.057	0.029	0.000	0.000	0.000	4.011	0.076	**0.000**
	best	−179.949	1087.226	1087.046	**−180.000**	1087.046	1087.046	1087.046	−172.669	1087.046	1087.046
	Rank	2	7	6	1	4	4	4	3	5	4
F8	mean	−119.640	**−119.714**	−119.624	−119.661	−119.666	−119.628	−119.648	−119.653	−119.681	−119.643
	std.	0.090	0.075	0.081	0.074	0.082	0.087	0.066	0.089	0.085	**0.053**
	best	**−119.851**	−119.800	−119.792	−119.791	−119.813	−119.831	−119.718	−119.777	−119.812	−119.728
	Rank	8	1	10	4	3	9	6	5	2	7
F9	mean	−329.333	−327.314	**−330.000**	**−330.000**	**−330.000**	−322.342	−322.737	−322.409	**−330.000**	**−330.000**
	std.	0.723	1.487	**0.000**	**0.000**	**0.000**	2.928	2.696	2.528	**0.000**	**0.000**
	best	**−330.000**	−329.005	**−330.000**	**−330.000**	**−330.000**	−326.020	−327.015	−325.025	**−330.000**	**−330.000**
	Rank	2	3	1	1	1	5	4	6	1	1
F10	mean	−317.409	−319.259	−316.698	−322.417	−322.339	−319.885	−321.513	−318.339	−320.479	**−323.493**
	std.	5.428	6.162	3.125	2.704	3.604	4.132	3.593	4.213	3.542	**1.987**
	best	−324.188	−328.010	−322.465	−326.395	**−329.005**	−325.025	−327.015	−324.012	−324.562	−325.724
	Rank	9	7	10	2	3	6	4	8	5	1
F11	mean	93.633	**91.962**	95.319	96.078	92.093	94.002	94.632	93.193	94.889	95.188
	std.	**0.536**	1.023	0.572	0.698	2.355	1.652	0.997	1.049	1.196	0.746
	best	92.922	90.702	94.425	94.895	**90.000**	91.059	92.619	91.772	92.408	93.737
	Rank	4	1	9	10	2	5	6	3	7	8
F12	mean	−422.531	−203.253	−363.666	−186.398	**−456.999**	1445.385	794.693	998.556	−438.290	−444.398
	std.										
td.	47.468	507.838	65.389	84.882	**4.832**	5027.194	2860.501	3344.475	16.021	16.545	
	best	−455.719	−459.496	−431.128	−323.775	**−460.000**	**−460.000**	**−460.000**	−449.679	−455.477	−459.985
	Rank	4	6	5	7	1	10	8	9	3	2
F13	mean	−128.810	−129.352	−129.564	−129.675	−129.486	−129.196	−129.044	−129.307	**−129.697**	−129.666
	std.	0.146	0.271	0.118	**0.058**	0.088	0.165	0.256	0.160	0.118	0.073
	best	−129.109	−129.745	−129.769	−129.756	−129.689	−129.447	−129.384	−129.481	**−129.932**	−129.830
	Rank	10	6	4	2	5	8	9	7	1	3
F14	mean	−297.170	**−297.241**	−296.591	−296.562	−296.912	−297.215	−297.081	−297.230	−296.694	−296.902
	std.	0.306	0.685	**0.151**	0.252	0.217	0.249	0.459	0.353	0.205	0.207
	best	−297.642	**−298.570**	−296.948	−297.118	−297.246	−297.566	−297.701	−297.724	−297.108	−297.288
	Rank	4	1	9	10	6	3	5	2	8	7
F15	mean	209.829	387.275	140.582	177.355	136.613	450.644	401.382	397.863	124.232	**120.030**
	std.	99.415	203.967	28.900	64.825	21.448	141.887	156.564	166.285	12.918	**0.089**
	best	120.460	176.891	120.002	**120.000**	**120.000**	195.411	197.129	180.374	**120.000**	**120.000**
	Rank	6	7	4	5	3	10	9	8	2	1
F16	mean	232.105	254.799	257.700	220.475	**219.202**	228.617	230.096	242.547	231.537	223.754
	std.	10.777	29.896	14.508	4.893	4.653	9.831	9.792	13.648	10.701	**3.450**
	best	221.355	222.865	223.542	214.433	215.972	219.015	217.668	223.611	**209.614**	218.507
	Rank	7	9	10	2	1	4	5	8	6	3
F17	mean	251.164	247.905	269.953	**222.366**	224.970	246.701	228.562	236.482	249.588	241.073
	std.	19.976	21.479	16.875	19.527	7.003	19.773	**5.540**	11.421	10.448	11.036
	best	216.798	219.166	252.680	**172.389**	218.009	218.722	222.809	217.019	237.089	227.764
	Rank	9	7	10	1	2	6	3	4	8	5

(*continued on next page*)

Table 1 (*continued*)

F	AL.	PSOFIPS	PSOFDR	CLPSO	jDE	SaDE	TLBO	ETLBO	DGSTLBO	BSA	LBSA
F18	mean	580.649	864.345	702.404	738.946	760.000	709.009	715.153	858.448	**393.335**	603.702
	std.	277.920	96.241	**86.522**	235.263	158.114	237.366	240.232	148.634	155.192	253.504
	best	**310.000**	700.558	559.128	**310.000**	**310.000**	**310.000**	393.632	597.895	**310.000**	**310.000**
	Rank	2	10	4	7	8	5	6	9	1	3
F19	mean	442.020	864.003	752.056	622.746	740.949	756.348	716.366	912.207	**404.878**	572.111
	std.	214.463	131.102	**46.800**	269.169	162.827	201.804	227.717	115.327	162.270	278.437
	best	**310.000**	698.725	656.476	**310.000**	**310.000**	417.269	366.021	634.105	**310.000**	**310.000**
	Rank	2	9	7	4	6	8	5	10	1	3
F20	mean	666.374	896.631	711.274	843.889	810.000	748.635	791.997	956.972	**410.210**	442.654
	std.	251.178	117.945	75.428	40.267	**0.000**	186.662	164.780	80.840	177.868	211.159
	best	**310.000**	693.240	566.636	830.917	810.000	**310.000**	467.410	810.000	**310.000**	**310.000**
	Rank	3	9	4	8	7	5	6	10	1	2
F21	mean	**830.619**	1212.545	846.708	1093.998	930.000	1110.454	931.239	1035.818	832.592	910.002
	std.	222.507	400.425	**28.555**	267.013	170.294	367.780	360.290	366.993	86.672	171.591
	best	660.000	660.000	790.540	860.000	660.000	660.000	660.000	660.000	**585.919**	660.019
	Rank	1	10	3	8	5	9	6	7	2	4
F22	mean	1131.082	1177.217	1141.724	1110.531	1117.446	1150.209	1135.155	1155.322	1079.861	**1074.270**
	std.	3.327	79.315	7.787	83.245	5.557	29.625	24.736	61.423	148.046	145.718
	best	1125.586	1109.829	1131.022	886.118	1108.033	1116.253	1110.458	1094.723	**660.000**	**660.000**
	Rank	5	10	7	3	4	8	6	9	2	1
F23	mean	1069.522	1431.577	906.039	1282.261	1035.216	1178.115	1237.380	1351.502	918.677	**892.609**
	std.	179.445	215.479	42.468	261.521	178.858	232.518	257.118	254.423	**1.797**	56.624
	best	919.468	919.468	**785.173**	919.468	785.173	919.468	919.468	919.468	914.095	785.173
	Rank	5	10	2	8	4	6	7	9	3	1
F24	mean	588.505	1126.804	696.247	636.172	**460.000**	**460.000**	**460.000**	490.000	**460.000**	**460.000**
	std.	86.471	239.476	295.248	3.399	**0.000**	**0.000**	**0.000**	94.868	**0.000**	**0.000**
	best	460.001	760.000	460.211	628.653	**460.000**	**460.000**	**460.000**	**460.000**	**460.000**	**460.000**
	Rank	3	6	5	4	1	1	1	2	1	1
F25	mean	**612.785**	1339.389	1079.655	634.054	1082.075	1078.221	1076.067	1122.095	1080.375	1079.394
	std.	80.691	217.394	2.151	7.695	**1.638**	2.460	2.650	484.200	2.634	2.258
	best	**460.000**	1071.868	1077.376	613.127	1079.380	1074.200	1071.563	**460.000**	1074.953	1075.267
	Rank	1	10	6	2	8	4	3	9	7	5
Average rank		4.32	6.24	5.52	3.84	3.32	5.32	4.84	5.96	3.24	**2.84**

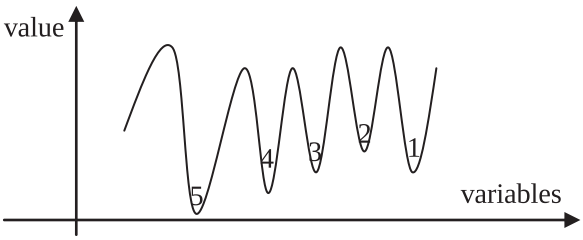

Fig. 2. Multimodal problems with five extremums.

best solutions. The results for CEC2005 show that LBSA achieved comparable performance, although it was not always better than others in all aspects.

5.3.2. Comparison of 30D function solution accuracy in CEC2005

In these experiments, the fitness function was chosen to be the difference between the real solution of the algorithms and the theoretical optimum of the functions. If the real solution of the *i*th algorithm is f_i, then the fitness function can be expressed as $|f_i - F_i^*|$. The dimension of functions in CEC2005 in this experiment was 30. The best, mean, and standard deviation of the solutions over 30 runs for the ten algorithms are shown in Table 2. Moreover, the ranks and the average ranks of the ten algorithms for the 25 functions are also given. Table 2 shows that the rank for LBSA in terms of the mean best solution was first for functions F1, F12, F13, F16, F17, F21, F23, and F24. jDE was ranked first for functions F1, F4, F5, F6, F9, F10, F18, F19, F20, and F22. The first ranks of the other functions were shared by the other eight algorithms. In terms of the total number of first ranks, jDE outperformed the other nine algorithms and LBSA was in second place. The best results are shown in bold in Table 2. The standard deviations of LBSA were smaller than those of the other algorithms for functions, F12, F16, F19, F24, and F25, and those of jDE outperformed the other algorithms for functions, F3, F4, F9, F10, F13, and F20. The smallest solutions in terms of standard deviation for the other functions were achieved by the other eight algorithms, and the sum of these for each algorithm was less than six. The statistical results in Table 2 generally indicate that the global performances of jDE and LBSA were better than those of the other algorithms, although jDE was slightly better than LBSA in this experiment. To display the convergence process of the ten algorithms for some of the functions in CEC2005, Fig. 3

Table 2
Comparative results of ten algorithms for 30D functions from CEC2005(the bold is the best solution).

F	AL.	PSOFIPS	PSOFDR	CLPSO	jDE	SaDE	TLBO	ETLBO	DGSTLBO	BSA	LBSA
F1	mean	1.057E−06	9.933E+02	4.839E−10	**0.000E+00**	**0.000E+00**	5.836E−26	3.138E−27	2.252E−06	3.995E−16	**0.000E+00**
	std.	3.274E−07	1.632E+03	1.409E−10	**0.000E+00**	**0.000E+00**	2.769E−26	1.661E−27	3.901E−06	2.267E−16	**0.000E+00**
	best	7.480E−07	0.000E+00	3.244E−10	**0.000E+00**	**0.000E+00**	2.640E−26	1.379E−27	2.565E−13	1.713E−16	**0.000E+00**
	Rank	7	9	6	1	1	4	3	8	5	1
F2	mean	4.560E+02	9.374E+01	6.355E+03	1.945E−03	8.734E−03	7.480E−05	**6.632E−05**	5.228E+02	2.644E+03	4.995E−02
	std.	6.497E+01	8.638E+01	7.050E+02	3.259E−03	4.026E−03	**3.423E−05**	7.342E−05	1.901E+02	2.038E+02	4.852E−02
	best	3.818E+02	8.972E−03	5.676E+03	5.487E−05	4.213E−03	3.527E−05	**2.493E−06**	3.241E+02	2.450E+03	1.714E−02
	Rank	7	6	10	3	4	2	1	8	9	5
F3	mean	1.265E+07	1.757E+06	4.814E+07	6.223E+05	**4.385E+05**	1.344E+06	1.331E+06	5.770E+06	3.069E+06	9.240E+05
	std.	1.170E+06	6.131E+05	5.129E+06	**5.257E+04**	1.529E+05	6.881E+05	2.882E+05	3.854E+06	2.791E+05	3.486E+05
	best	1.138E+07	1.084E+06	4.296E+07	5.870E+05	**3.380E+05**	6.838E+05	1.128E+06	2.335E+06	2.784E+06	7.127E+05
	Rank	9	5	10	2	1	5	4	8	7	3
F4	mean	3.019E+03	7.308E+02	1.613E+04	**1.621E+01**	4.769E+02	2.128E+03	8.611E+02	2.102E+03	9.404E+03	7.097E+02
	std.	8.707E+02	2.258E+02	1.800E+03	**1.004E+01**	7.790E+02	2.915E+03	7.453E+02	2.238E+03	7.848E+02	4.298E+02
	best	2.016E+03	5.912E+02	1.505E+04	**9.789E+00**	2.337E+01	1.736E+02	4.067E+02	6.345E+02	8.510E+03	2.659E+02
	Rank	8	4	10	1	2	7	5	6	9	3
F5	mean	3.104E+03	6.156E+03	4.315E+03	**2.067E+03**	2.427E+03	4.546E+03	4.161E+03	9.835E+03	3.334E+03	3.029E+03
	std.	**1.055E+02**	1.750E+03	9.405E+02	9.514E+02	2.732E+02	3.140E+02	4.601E+02	1.524E+03	1.767E+02	7.939E+02
	best	3.009E+03	5.077E+03	3.523E+03	**1.098E+03**	2.141E+03	4.197E+03	3.678E+03	8.119E+03	3.130E+03	2.501E+03
	Rank	4	9	7	1	2	8	6	10	5	3
F6	mean	2.891E+01	1.638E+07	3.974E+01	5.364E+01	7.807E+01	**2.297E+01**	3.895E+01	2.164E+04	7.803E+01	2.958E+01
	std.	2.546E+00	2.646E+07	2.123E+01	3.630E+01	**2.212E+00**	2.780E+00	3.169E+01	3.596E+04	3.027E+01	4.229E+01
	best	2.714E+01	2.290E+02	1.529E+01	1.174E+01	7.557E+01	2.109E+01	2.027E+01	1.766E+02	4.494E+01	**3.069E+00**
	Rank	2	10	5	6	8	1	4	9	7	3
F7	mean	3.846E-01	4.918E+03	4.696E+03	**9.037E-03**	4.696E+03	4.696E+03	4.696E+03	9.207E+02	4.696E+03	4.696E+03
	std.	9.646E-02	3.844E+02	**0.000E+00**	1.421E-03	3.750E-12	1.575E-12	9.095E-13	1.570E+02	1.720E-08	1.575E-12
	best	2.859E-01	4.696E+03	4.696E+03	**7.396E-03**	4.696E+03	4.696E+03	4.696E+03	7.393E+02	4.696E+03	4.696E+03
	Rank	2	5	4	1	4	4	4	3	4	4
F8	mean	2.093E+01	2.092E+01	2.093E+01	2.098E+01	2.097E+01	2.098E+01	**2.089E+01**	2.095E+01	2.098E+01	2.096E+01
	std.	**1.087E−02**	2.802E−02	8.812E−02	3.299E−02	2.665E−02	1.689E−02	6.181E−02	6.464E−02	4.595E−02	5.689E−02
	best	2.092E+01	2.089E+01	2.085E+01	2.095E+01	2.095E+01	2.097E+01	**2.084E+01**	2.088E+01	2.093E+01	2.091E+01
	Rank	3	2	3	7	6	7	1	4	7	5
F9	mean	5.904E+01	5.164E+01	5.082E−04	**0.000E+00**	3.317E−01	9.618E+01	6.567E+01	9.220E+01	2.277E+00	1.184E−15
	std.	1.441E+01	5.605E+00	2.078E−04	**0.000E+00**	5.744E−01	2.185E+01	3.011E+01	1.580E+01	1.295E+00	2.051E−15
	best	4.240E+01	4.754E+01	2.719E−04	**0.000E+00**	**0.000E+00**	8.258E+01	4.179E+01	8.258E+01	1.134E+00	**0.000E+00**
	Rank	7	6	3	1	4	10	8	9	5	2
F10	mean	1.822E+02	1.141E+02	1.647E+02	**5.881E+01**	7.975E+01	9.228E+01	9.913E+01	8.652E+01	8.281E+01	7.969E+01
	std.	5.261E+00	1.892E+01	5.872E+00	**3.873E+00**	4.349E+01	4.150E+01	1.259E+01	7.217E+00	1.541E+01	7.575E+00
	best	1.788E+02	9.451E+01	1.608E+02	5.441E+01	**3.022E+01**	5.771E+01	8.743E+01	8.107E+01	6.963E+01	7.131E+01
	Rank	10	8	9	1	3	6	7	5	4	2
F11	mean	2.879E+01	**1.803E+01**	2.798E+01	3.317E+01	2.997E+01	3.467E+01	3.714E+01	2.548E+01	3.091E+01	2.833E+01
	std.	2.252E+00	2.803E+00	**1.205E+00**	2.996E+00	1.669E+00	4.680E+00	1.629E+00	1.237E+00	2.377E+00	1.680E+00
	best	2.625E+01	**1.551E+01**	2.685E+01	3.104E+01	2.840E+01	2.930E+01	3.526E+01	2.413E+01	2.843E+01	2.661E+01
	Rank	5	1	3	8	6	9	10	2	7	4
F12	mean	2.555E+04	2.882E+04	4.411E+04	3.067E+04	6.056E+03	9.523E+03	1.550E+04	3.196E+04	2.801E+04	**2.976E+03**
	std.	9.809E+03	4.351E+04	3.443E+03	2.969E+03	7.626E+03	3.881E+03	1.420E+04	1.637E+04	1.173E+04	**1.281E+03**
	best	1.962E+04	3.259E+03	4.022E+04	2.755E+04	**1.008E+03**	6.257E+03	3.727E+03	1.320E+04	1.447E+04	1.698E+03
	Rank	5	7	10	8	2	3	4	9	6	1
F13	mean	1.358E+01	2.870E+00	3.409E+00	1.911E+00	4.792E+00	4.284E+00	4.063E+00	5.512E+00	2.524E+00	**1.883E+00**
	std.	1.025E−01	7.919E−01	2.156E−01	**1.650E−02**	6.707E−01	5.438E−01	1.202E+00	1.189E−01	1.192E−01	1.203E−01
	best	1.347E+01	2.293E+00	3.162E+00	1.900E+00	4.113E+00	3.912E+00	2.682E+00	5.410E+00	2.408E+00	**1.771E+00**
	Rank	10	4	5	2	8	7	6	9	3	1
F14	mean	1.280E+01	**1.185E+01**	1.312E+01	1.362E+01	1.309E+01	1.289E+01	1.309E+01	1.280E+01	1.309E+01	1.286E+01
	std.	1.223E−01	7.922E−01	1.258E−01	1.847E−01	1.846E−01	1.513E−01	**7.225E−02**	5.319E−01	3.739E−01	1.284E−01
	best	1.266E+01	**1.112E+01**	1.298E+01	1.343E+01	1.288E+01	1.278E+01	1.303E+01	1.234E+01	1.266E+01	1.276E+01
	Rank	2	1	6	7	5	4	5	2	5	3
F15	mean	3.544E+02	5.285E+02	1.214E+02	1.682E+02	2.354E+02	4.713E+02	4.746E+02	4.104E+02	**7.111E+01**	4.015E+02
	std.	6.430E+01	5.548E+01	2.545E+01	7.807E+01	2.047E+02	9.623E+01	4.142E+01	4.289E+01	**1.159E+01**	1.013E+02
	best	3.007E+02	4.831E+02	9.610E+01	7.919E+01	**6.213E+00**	3.977E+02	4.268E+02	3.628E+02	6.421E+01	3.000E+02
	Rank	5	10	2	3	4	8	9	6	1	7
F16	mean	2.507E+02	2.413E+02	2.018E+02	1.077E+02	1.051E+02	2.400E+02	2.652E+02	2.160E+02	1.484E+02	**8.450E+01**
	std.	5.874E+01	1.017E+02	2.174E+01	5.049E+01	3.010E+01	2.269E+02	1.175E+02	1.178E+01	3.611E+01	**5.658E+00**
	best	2.087E+02	1.245E+02	1.775E+02	7.827E+01	**7.058E+01**	1.078E+02	1.912E+02	2.027E+02	1.240E+02	7.797E+01
	Rank	9	8	5	3	2	7	10	6	4	1
F17	mean	2.812E+02	1.932E+02	3.001E+02	2.417E+02	1.663E+02	2.639E+02	2.404E+02	2.334E+02	2.117E+02	**1.511E+02**
	std.	6.842E+01	1.022E+02	4.652E+01	1.212E+02	1.345E+01	1.708E+02	1.690E+02	**8.678E+00**	1.541E+01	4.484E+01
	best	2.388E+02	1.267E+02	2.574E+02	1.413E+02	1.508E+02	1.399E+02	1.323E+02	2.262E+02	1.964E+02	**1.198E+02**
	Rank	9	3	10	7	2	8	6	5	4	1

(*continued on next page*)

Table 2 (*continued*)

F	AL.	PSOFIPS	PSOFDR	CLPSO	jDE	SaDE	TLBO	ETLBO	DGSTLBO	BSA	LBSA
F18	mean	8.323E+02	9.388E+02	9.098E+02	**8.178E+02**	9.116E+02	9.148E+02	8.771E+02	9.938E+02	9.153E+02	9.186E+02
	std.	1.686E+00	2.396E+01	7.681E-01	1.053E+00	**4.342E-01**	4.459E+00	6.679E+01	1.333E+01	3.761E+00	6.806E+00
	best	8.310E+02	9.187E+02	9.090E+02	8.167E+02	9.111E+02	9.105E+02	**8.000E+02**	9.830E+02	9.119E+02	9.108E+02
	Rank	2	9	4	1	5	6	3	10	7	8
F19	mean	8.323E+02	9.449E+02	9.100E+02	**8.180E+02**	8.741E+02	9.594E+02	9.289E+02	9.803E+02	9.161E+02	9.137E+02
	std.	4.396E-01	2.678E+01	4.197E-01	4.957E-01	6.415E+01	5.067E+01	2.414E+01	3.604E+01	2.038E+00	**1.897E−01**
	best	8.318E+02	9.172E+02	9.096E+02	8.175E+02	**8.000E+02**	9.201E+02	9.093E+02	9.551E+02	9.147E+02	9.136E+02
	Rank	2	8	4	1	3	9	7	10	6	5
F20	mean	8.321E+02	9.501E+02	9.088E+02	**8.172E+02**	8.737E+02	9.475E+02	9.216E+02	1.016E+03	9.149E+02	8.779E+02
	std.	9.107E−01	2.351E+01	9.501E-01	**4.755E-01**	6.383E+01	3.759E+01	8.330E+00	3.126E+01	2.841E+00	6.752E+01
	best	8.310E+02	9.327E+02	9.078E+02	8.168E+02	8.000E+02	9.141E+02	9.136E+02	9.804E+02	9.128E+02	**8.000E+02**
	Rank	2	9	5	1	3	8	7	10	6	4
F21	mean	5.000E+02	9.728E+02	6.508E+02	8.604E+02	**5.000E+02**	9.551E+02	**5.000E+02**	9.694E+02	**5.000E+02**	**5.000E+02**
	std.	3.922E−06	1.750E+02	1.376E+02	1.503E+00	**0.000E+00**	3.941E+02	8.247E−13	4.062E+02	1.842E−13	1.969E−13
	best	5.000E+02	8.549E+02	5.427E+02	8.588E+02	**5.000E+02**	**5.000E+02**	**5.000E+02**	5.005E+02	**5.000E+02**	**5.000E+02**
	Rank	1	6	2	3	1	4	1	5	1	1
F22	mean	5.290E+02	8.860E+02	9.189E+02	**5.004E+02**	9.275E+02	9.220E+02	9.094E+02	1.055E+03	9.709E+02	9.372E+02
	std.	**2.608E−01**	2.439E+01	2.153E+01	4.134E-01	4.853E+00	3.630E+01	3.054E+01	8.101E+01	8.997E+00	1.765E+01
	best	5.287E+02	8.648E+02	8.984E+02	**5.000E+02**	9.230E+02	8.831E+02	8.747E+02	9.689E+02	9.617E+02	9.266E+02
	Rank	2	3	5	1	7	6	4	10	9	8
F23	mean	**5.342E+02**	1.030E+03	6.655E+02	8.668E+02	**5.342E+02**	1.052E+03	1.226E+03	8.292E+02	**5.342E+02**	**5.342E+02**
	std.	2.583E−04	1.329E+02	5.669E+01	1.596E−01	1.186E−04	2.102E+02	3.965E+01	1.592E+02	**1.987E−05**	4.661E−04
	best	**5.342E+02**	8.767E+02	6.255E+02	8.667E+02	5.342E+02	8.094E+02	1.184E+03	6.727E+02	5.342E+02	5.342E+02
	Rank	1	5	2	4	1	6	7	3	1	1
F24	mean	2.183E+02	9.579E+02	9.489E+02	2.126E+02	**2.000E+02**	**2.000E+02**	**2.000E+02**	8.476E+02	**2.000E+02**	**2.000E+02**
	std.	9.720E−01	2.510E+01	1.994E+00	1.237E+00	3.481E−14	4.702E−11	3.212E−12	5.342E+02	1.879E−12	**0.000E+00**
	best	2.176E+02	9.399E+02	9.471E+02	2.113E+02	**2.000E+02**	**2.000E+02**	**2.000E+02**	2.482E+02	**2.000E+02**	**2.000E+02**
	Rank	3	6	5	2	1	1	1	4	1	1
F25	mean	**2.070E+02**	1.064E+03	1.008E+03	2.120E+02	9.951E+02	1.006E+03	1.052E+03	1.211E+03	1.016E+03	9.938E+02
	std.	1.213E+01	2.582E+01	1.268E+01	1.878E+00	4.148E+00	2.774E+01	4.316E+01	1.059E+02	8.040E+00	**1.593E+00**
	best	**2.000E+02**	1.036E+03	1.000E+03	2.103E+02	9.924E+02	9.817E+02	1.009E+03	1.089E+03	1.007E+03	9.922E+02
	Rank	1	9	6	2	4	5	8	10	7	3
Average rank		4.72	6.12	5.64	**3.08**	3.56	5.80	5.24	6.84	5.20	3.20

contains eight representative convergence graphs for the 30D functions F1, F2 (unimodal functions), F12, F13 (multimodal functions), F16, F17, F23, and F24 (hybrid composition function). This figure shows that the convergence speed of SaDE was faster than those of the other algorithms for F1, and ETLBO outperformed other algorithms for function F2. For F12, F13, F16, and F17, LBSA achieved the fastest performance. For F23, the convergence speed of SaDE, BSA, and LBSA were almost the same. For F24, SaDE, TLBO, ETLBO, BSA, and LBSA had almost the same convergence speed. Table 2 and Fig. 3 also indicate that LBSA achieved a performance that is comparable to that of the other algorithms, but it was not always the best for all statistical measures. Comparing the performance between LBSA and jDE, we can find that the mean best solutions of jDE were the smallest for ten functions, but the LBSA provided the smallest mean best solutions for eight functions. The average rank in terms of mean best solution of jDE was 3.08, which is better than the value (3.20) of LBSA. The comparisons show that jDE outperformed LBSA for these 25 functions.

5.3.3. T-test comparisons for 30D functions in CEC2005

The differences between LBSA and the other algorithms were tested using a *t*-test [42]. Before the *t*-test, an F-test was used to test the equality of the variances between LBSA and the other algorithms. The F-values between LBSA and the other algorithms are shown in Table 3. Table 3 indicates that the differences of variances between LBSA and the other algorithms were not significant. In *t*-test experiments, a two-tailed test with a significance level of 0.05 was adopted in this study, and the t-values and p-values for every function pair are listed in Table 4. The better results between LBSA and the other algorithms for different functions are shown in bold. 'B', 'W', and 'S' indicate that LBSA performed significantly better than, almost the same as, or significantly worse than the compared algorithm, respectively. Table 4 shows that the average excellent and good rate between LBSA and the other algorithms for the 30D functions in CEC2005 in terms of the *t*-test was 72.22% ($\sum_{i=1}^{9}(B_i+S_i)/30^*9$), and the average performance of LBSA was good, although it was not always the best for all functions with respect to the ten algorithms. The table also shows that the performance of basic BSA was improved by LBSA.

5.4. Experiments for functions in CEC2014

CEC2014 includes 30 benchmark functions. Because their global optima are shifted, finding them is relatively difficult, and the acceptable solutions are hard to obtain. In this part, only the mean, standard deviation, best solutions, and *t*-test results of different algorithms are given. In this part, the fitness function is the error between the real optimal solution and

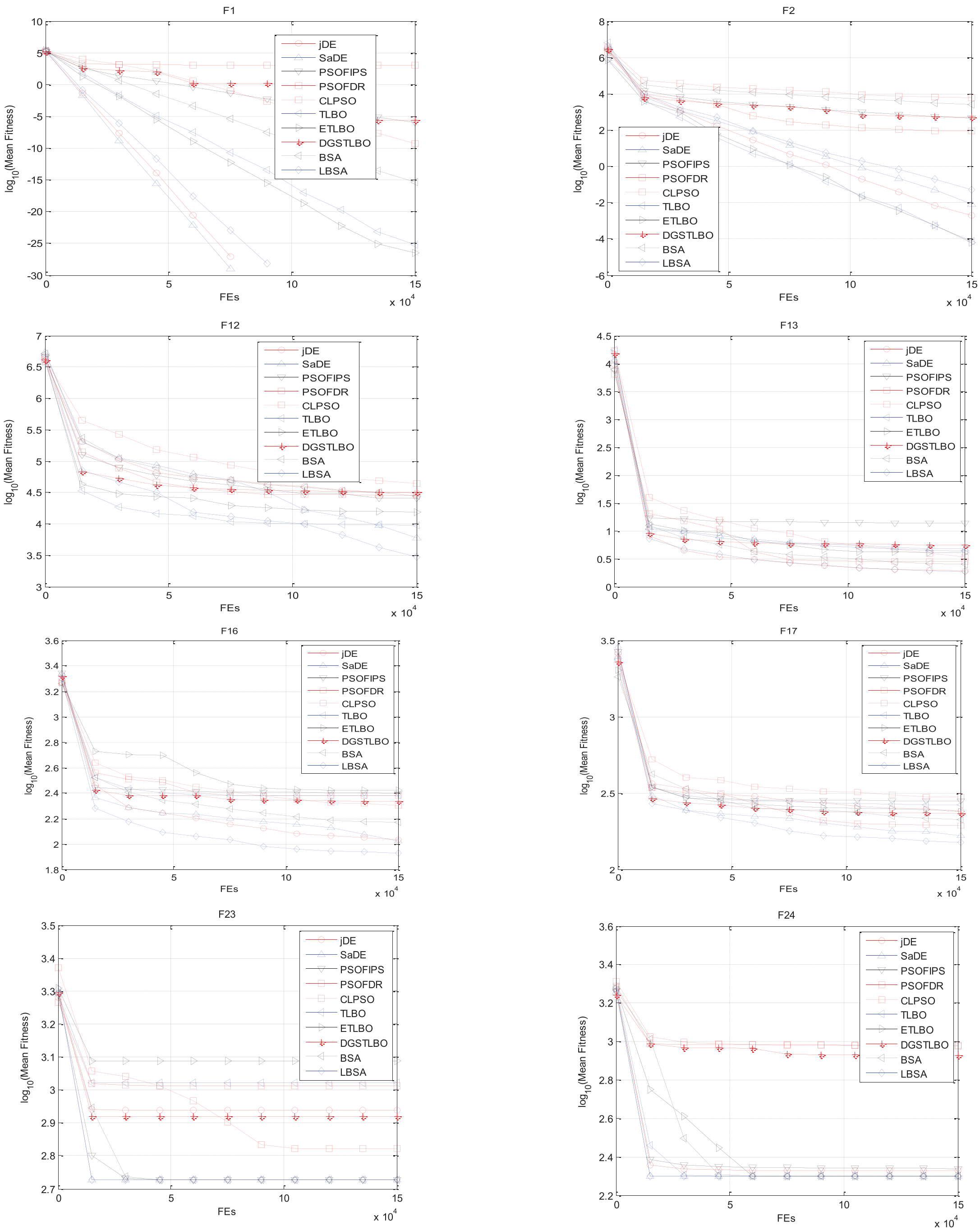

Fig. 3. The convergence graphs of eight respective functions.

the solution of the optimisation algorithm. Assume that the real global optimal solution is $x*$, and the best solution obtained by the optimisation algorithm is $y*$. Then, $|f(y*)\text{-}f(x*)|$ is chosen to be the fitness function.

5.4.1. Comparison of 10D function solution accuracy in CEC2014

The statistical results in terms of mean, standard deviation, best solution, rank according to the mean, and average rank of different algorithms for the 30D functions in CEC2014 are listed in Table 5. The best results are shown in bold. Table 5 shows that LBSA performed best in terms of mean for functions F26, F30, F34, F36, F37, F39, F41, and F47. jDE performed best in terms of mean for functions F28, F29, F33, F35, F45, F48, F50, F53, and F55. The performance in terms of mean with SaDE was better than the others for functions F27, F31, F32, F42, F43, F44, F46 and F52. For function F38, BSA had the

Table 3
Comparisons between LBSA and other algorithms using F-test for 30D functions of CEC2005.

F	jDE	SaDE	PSOwFIPS	PSOFDR	CLPSO	TLBO	ETLBO	DGSTLBO	BSA
F1	0.00E+00	0.00E+00	6.22E−48	2.50E−67	3.36E-41	8.70E−10	2.42E−07	4.38E−50	1.30E−29
F2	8.98E−03	1.37E−02	1.12E−06	6.31E−07	9.47E−09	9.96E−07	4.58E−06	1.30E−07	1.13E−07
F3	4.45E−02	3.23E−01	1.63E−01	4.89E−01	9.20E−03	4.09E−01	8.12E−01	1.62E−02	7.81E−01
F4	1.09E−03	4.67E−01	3.92E−01	4.33E−01	1.08E−01	4.25E−02	4.99E−01	7.11E−02	4.61E−01
F5	8.21E−01	2.12E−01	3.47E−02	3.41E−01	8.32E−01	2.71E−01	5.03E−01	4.27E−01	9.44E−02
F6	8.48E−01	5.46E−03	7.22E−03	5.11E−12	4.02E−01	8.61E−03	7.19E−01	2.77E−06	6.77E−01
F7	3.32E−17	1.00E+00	7.20E−21	4.54E−28	0.00E+00	0.00E+00	0.00E+00	2.72E−27	2.26E−07
F8	5.03E−01	3.60E−01	7.05E−02	3.90E−01	5.88E−01	1.62E−01	9.17E−01	8.73E−01	7.90E−01
F9	0.00E+00	2.55E−29	4.05E−32	2.68E−31	1.95E−22	1.76E−32	9.28E−33	3.37E−32	5.01E−30
F10	4.15E−01	5.89E−02	6.51E−01	2.76E−01	7.51E−01	6.45E−02	5.31E−01	9.52E−01	3.89E−01
F11	4.78E−01	9.94E−01	7.15E−01	5.29E−01	6.80E−01	2.28E−01	9.69E−01	7.03E−01	6.66E−01
F12	3.14E−01	5.49E−02	3.36E−02	1.73E−03	2.43E−01	1.97E−01	1.61E−02	1.22E−02	2.36E−02
F13	3.69E−02	6.23E−02	8.41E−01	4.51E−02	4.75E−01	9.33E−02	1.98E−02	9.89E−01	9.91E−01
F14	6.52E−01	6.52E−01	9.51E−01	5.12E−02	9.79E−01	8.38E−01	4.81E−01	1.10E−01	2.11E−01
F15	7.45E−01	3.93E−01	5.74E−01	4.62E−01	1.19E−01	9.49E−01	2.87E−01	3.04E−01	2.59E−02
F16	2.48E−02	6.83E−02	1.84E−02	6.18E−03	1.27E−01	1.24E−03	4.62E−03	3.75E−01	4.79E−02
F17	2.41E−01	1.65E−01	6.01E−01	3.23E−01	9.63E−01	1.29E−01	1.32E−01	7.22E−02	2.11E−01
F18	4.68E−02	8.11E−03	1.16E−01	1.49E−01	2.52E−02	6.01E−01	2.06E−02	4.14E−01	4.68E−01
F19	2.55E−01	1.75E−05	3.14E−01	1.00E−04	3.39E−01	2.80E−05	1.24E−04	5.54E−05	1.72E−02
F20	9.92E−05	9.44E−01	3.64E−04	2.16E−01	3.96E−04	4.73E−01	3.00E−02	3.53E−01	3.53E−03
F21	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
F22	1.10E−03	1.41E−01	4.37E−04	6.87E−01	8.04E−01	3.82E−01	5.01E−01	9.06E−02	4.12E−01
F23	1.71E−05	1.22E−01	4.70E−01	2.46E−11	1.35E−10	9.83E−12	2.76E−10	1.72E−11	3.63E−03
F24	1.58E−27	0.00E+00	2.57E−27	3.85E−30	6.09E−28	1.10E−06	1.97E−04	8.49E−33	8.16E−04
F25	8.37E−01	2.57E−01	3.39E−02	7.59E−03	3.11E−02	6.58E−03	2.72E−03	4.53E−04	7.56E−02

smallest mean among all algorithms. For function F40, the mean of FDRPSO was the smallest. The three TLBOs had the same mean for function F49, which is the smallest of the ten algorithms. For function F54, PSOFIPS had the best performance in terms of mean. Considering the best solutions, LBSA outperformed the other algorithms for functions F26, F29, F30, F34, F37, F41, and F42. jDE performed best for 11 functions (F27, F28, F33, F35, F43, F45, F48, F50, F51, F53, and F55), and the performance in terms of the best solution of SaDE was better than the other algorithms for functions F27, F28, F31, F32, F42, F46, and F51. PSOFIPS is better than other algorithms in terms of the best solutions for functions F48, F51, and F54. PSOFDR outperformed the other algorithms for functions F26, F41, F42, F44, and F51. The table also shows the rank according to the mean of the ten algorithms for 30 functions. LBSA ranked in first place for nine functions, jDE for ten functions, and SaDE for nine functions. The average rank of jDE and LBSA were the same. The ranks of the other algorithms were relatively lower than these three algorithms.

5.4.2. T-test comparisons for functions in CEC2014

The *t*-test results for CEC2014 between LBSO and the other algorithms are listed in Table 6, which shows that the average excellent and good rate between LBSA and the other algorithms for the 30D functions of CEC2014 in terms of the *t*-test was 82.96% ($\sum_{i=1}^{9} (B_i + S_i)/30*9$). The table also shows that the average performance of LBSA was good, and the performance of basic BSA was improved.

6. ANN training

System identification is a method of identifying or measuring the mathematical model of a system from measurements of the system inputs and outputs. The accurate mathematical model of a nonlinear system is sometimes difficult to obtain, and hence building an intelligent model is very important. The L-step-ahead prediction approach uses the earlier states $x(t - d+1)$, $x(t - d+2)$, ..., $x(t)$ to predict the state of $\hat{x}(t+L)$. It can be expressed as follows:

$$\hat{x}(t+L) = f(x(t-d+1), x(t-d+2), \cdots, x(t-d+k), \cdots, x(t)) \quad (16)$$

where d is the number of input variables or embedding dimension, and L is the number of steps ahead of the multi-step ahead direct prediction model. An ANN is a flexible mathematical structure that can be used to identify complex nonlinear systems and predict chaotic time series. A powerful optimisation technique may help ANN find optimal parameters for high dimensional and multimodal problems. In turn, training the parameters of ANN is usually done to test the effectiveness of an optimisation algorithm. To deeply test the effectiveness of the LBSA, three feed-forward neural networks for nonlinear system identification and chaotic time series prediction were chosen as learning objectives. The three-layer ANN ($n \times p \times q$) is shown in Fig. 4.

In Fig. 4, a, b, and c are the bias of the units; W_{ij} is the weight between the input layer and hidden layer, i = 1, 2, 3, ..., n; and W_{jk} is the weight between the hidden layer and output layer, j = 1, 2, 3, ..., p, k = 1, 2, 3, ..., q.

Table 4
Comparison between LBSA and other algorithms using *t*-test for 30D functions in CEC2005.

F	AL.	jDE	SaDE	PSOwFIPS	PSOFDR	CLPSO	TLBO	ETLBO	DGSTLBO	BSA
F1	T	0.00000	0.00000	**17.68471**	**3.33398**	**18.80463**	**11.54607**	**10.34528**	**3.16228**	**9.65364**
	P	0.00000	0.00000	**0.00000**	**0.00150**	**0.00000**	**0.00000**	**0.00000**	**0.00249**	**0.00000**
F2	T	−5.40719	−4.63710	**38.44279**	**5.94088**	**49.37527**	−5.63054	−5.63149	**15.05782**	**71.08136**
	P	0.00000	0.00002	**0.00000**	**0.00000**	**0.00000**	0.00000	0.00000	**0.00000**	**0.00000**
F3	T	−4.68677	−6.98513	**52.63073**	**6.46872**	**50.30775**	**2.98004**	**4.93468**	**6.85918**	**26.30395**
	P	0.00002	0.00000	**0.00000**	**0.00000**	**0.00000**	**0.00421**	**0.00001**	**0.00000**	**0.00000**
F4	T	−8.83578	−1.43331	**13.02394**	0.23750	**45.62547**	**2.63607**	0.96386	**3.34643**	**53.21922**
	P	0.00000	0.15714	**0.00000**	0.81311	**0.00000**	**0.01074**	0.33912	**0.00144**	**0.00000**
F5	T	−4.24931	−3.92747	0.51187	**8.91373**	**5.72179**	**9.73227**	**6.75718**	**21.69149**	**2.05247**
	P	0.00008	0.00023	0.61069	**0.00000**	**0.00000**	**0.00000**	**0.00000**	**0.00000**	**0.04465**
F6	T	**2.36387**	**6.27131**	−0.08681	**3.38988**	1.17613	−0.85451	0.97110	**3.29211**	**5.10191**
	P	**0.02146**	**0.00000**	0.93112	**0.00126**	0.24435	0.39634	0.33553	**0.00170**	**0.00000**
F7	T	−18,101,746.42079	**23.27015**	−266,644.09865	**3.16228**	−12.64911	−2.23607	−5.47723	−131.68793	**12.63842**
	P	0.00000	**0.00000**	0.00000	**0.00249**	0.00000	0.02921	0.00000	0.00000	**0.00000**
F8	T	1.64038	0.53111	−3.44622	−3.90129	−1.76397	1.59020	−4.98949	−0.57250	1.47252
	P	0.10634	0.59737	0.00106	0.00025	0.08300	0.11723	0.00001	0.56920	0.14629
F9	T	−3.16228	**3.16228**	**22.43949**	**50.46114**	**13.39723**	**24.10822**	**11.94422**	**31.95200**	**9.62749**
	P	0.00249	**0.00249**	**0.00000**	**0.00000**	**0.00000**	**0.00000**	**0.00000**	**0.00000**	**0.00000**
F10	T	−13.44492	0.00782	**60.89409**	**9.24016**	**48.58806**	1.63538	**7.24567**	**3.57657**	0.99441
	P	0.00000	0.99379	**0.00000**	**0.00000**	**0.00000**	0.10739	**0.00000**	**0.00071**	0.32415
F11	T	**7.71623**	**3.79701**	0.89863	−17.27548	−0.93886	**6.98202**	**20.60879**	−7.48811	**4.85430**
	P	**0.00000**	**0.00035**	0.37257	0.00000	0.35170	**0.00000**	**0.00000**	0.00000	**0.00001**
F12	T	**46.90837**	**2.18136**	**12.50062**	**3.25139**	**61.33180**	**8.77446**	**4.81014**	**9.66598**	**11.62215**
	P	**0.00000**	**0.03322**	**0.00000**	**0.00192**	**0.00000**	**0.00000**	**0.00001**	**0.00000**	**0.00000**
F13	T	1.24288	**23.38608**	**405.50792**	**6.75056**	**33.84927**	**23.61371**	**9.87963**	**117.49133**	**20.72818**
	P	0.21891	**0.00000**	**0.00000**	**0.00000**	**0.00000**	**0.00000**	**0.00000**	**0.00000**	**0.00000**
F14	T	**18.49545**	**5.69450**	−1.89436	−6.87122	**8.15495**	1.03281	**8.61714**	−0.56327	**3.17343**
	P	**0.00000**	**0.00000**	0.06317	0.00000	**0.00000**	0.30598	**0.00000**	0.57542	**0.00241**
F15	T	−9.99534	−3.98428	−2.14955	**6.02115**	−14.69182	**2.73337**	**3.65687**	0.44053	−17.75187
	P	0.00000	0.00019	0.03577	**0.00000**	0.00000	**0.00830**	**0.00055**	0.66119	0.00000
F16	T	**2.50162**	**3.67786**	**15.42945**	**8.43688**	**28.59267**	**3.75187**	**8.40948**	**55.11609**	**9.57010**
	P	**0.01521**	**0.00052**	**0.00000**	**0.00000**	**0.00000**	**0.00041**	**0.00000**	**0.00000**	**0.00000**
F17	T	**3.83912**	1.77984	**8.70942**	**2.06396**	**12.63384**	**3.49691**	**2.79639**	**9.86821**	**6.99704**
	P	**0.00031**	0.08034	**0.00000**	**0.04350**	**0.00000**	**0.00091**	**0.00700**	**0.00000**	**0.00000**
F18	T	−80.19478	−5.63544	−67.42664	**4.42834**	−7.01082	−2.56462	−3.38845	**27.51971**	−2.36599
	P	0.00000	0.00000	0.00000	**0.00004**	0.00000	0.01294	0.00127	**0.00000**	0.02134
F19	T	−987.60043	−3.38841	−932.07451	**6.37335**	−44.16027	**4.92961**	**3.43036**	**10.11533**	**6.28060**
	P	0.00000	0.00127	0.00000	**0.00000**	0.00000	**0.00001**	**0.00112**	**0.00000**	**0.00000**
F20	T	−4.91949	−0.24609	−3.71294	**5.53261**	**2.51308**	**4.93922**	**3.52358**	**10.19886**	**3.00469**
	P	0.00001	0.80648	0.00046	**0.00000**	**0.01477**	**0.00001**	**0.00084**	**0.00000**	**0.00392**
F21	T	**1313.62799**	−4.74342	**3.56127**	**14.79979**	**6.00513**	**6.32393**	**10.28143**	**6.32969**	**17.32051**
	P	**0.00000**	0.00001	**0.00075**	**0.00000**	**0.00000**	**0.00000**	**0.00000**	**0.00000**	**0.00000**
F22	T	−135.50194	−2.90207	−126.65053	−9.31134	−3.59045	−2.06451	−4.31748	**7.79198**	**9.33451**
	P	0.00000	0.00523	0.00000	0.00000	0.00068	0.04345	0.00006	**0.00000**	**0.00000**
F23	T	**11,414.66588**	−2.88386	−1.14256	**20.43957**	**12.69206**	**13.49374**	**95.50230**	**10.15204**	−3.55380
	P	**0.00000**	0.00550	0.25792	**0.00000**	**0.00000**	**0.00000**	**0.00000**	**0.00000**	0.00076
F24	T	**55.94285**	−482.99068	**103.24512**	**165.41374**	**2056.93066**	**9.99764**	**10.95402**	**6.63985**	**3.31402**
	P	**0.00000**	0.00000	**0.00000**	**0.00000**	**0.00000**	**0.00000**	**0.00000**	**0.00000**	**0.00159**
F25	T	−1738.64382	1.69241	−352.23753	**14.86674**	**6.04790**	**2.31739**	**7.38330**	**11.25121**	**15.09965**
	P	0.00000	0.09593	0.00000	**0.00000**	**0.00000**	**0.02403**	**0.00000**	**0.00000**	**0.00000**
B		9	8	12	20	17	17	18	20	20
W		13	10	8	4	5	4	5	2	3
S		3	7	5	1	3	4	2	3	2

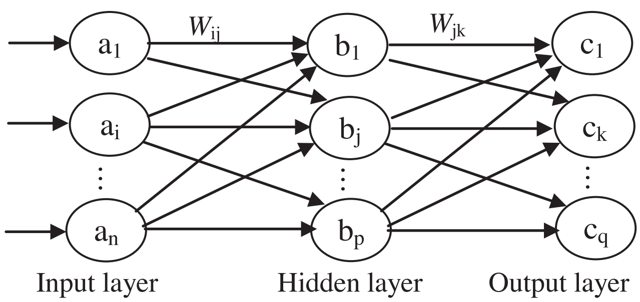

Fig. 4. Structure of a three layers feed forward neural network.

Table 5
Comparative results of ten algorithms for 30D functions in CEC2014.

F	AL.	PSOFIPS	PSOFDR	CLPSO	jDE	SaDE	TLBO	ETLBO	DGSTLBO	BSA	LBSA
F26	mean	1.02E+07	2.78E+06	2.92E+07	3.37E+05	4.41E+05	8.28E+05	3.33E+05	1.04E+07	5.12E+06	**2.79E+05**
	std.	2.88E+06	1.98E+06	4.33E+06	2.98E+05	3.04E+05	7.89E+05	2.02E+05	8.61E+06	4.10E+06	**1.65E+05**
	best	5.60E+06	1.32E+05	2.14E+07	1.06E+05	1.23E+05	5.77E+04	1.37E+05	3.15E+06	4.46E+05	**7.85E+04**
	Rank	8	6	10	3	4	5	2	9	7	1
F27	mean	1.13E+04	1.53E+08	1.26E+03	1.99E−14	**2.84E−15**	1.44E+02	1.19E+02	4.59E+06	8.34E−01	4.83E−14
	std.	5.27E+03	3.61E+08	7.40E+02	1.37E−14	**8.99E−15**	1.72E+02	1.56E+02	1.11E+07	1.48E+00	1.92E−14
	best	3.63E+03	4.60E+01	2.76E+02	**0.00E+00**	**0.00E+00**	2.82E+01	3.29E−01	1.48E+03	5.38E−02	2.84E−14
	Rank	8	10	7	2	**1**	6	5	9	4	3
F28	mean	6.94E+03	5.40E+02	7.67E+02	**4.55E−14**	4.30E−12	2.39E+03	2.14E+03	1.44E+01	5.74E−04	2.27E−13
	std.	5.03E+03	6.05E+02	1.20E+03	**2.40E−14**	1.25E−11	1.77E+03	1.02E+03	1.68E+01	9.44E−04	1.00E−13
	best	1.32E+03	1.36E+01	5.76E+01	**0.00E+00**	0.00E+00	2.90E+02	5.59E+02	6.71E−01	3.13E−05	1.14E−13
	Rank	10	6	7	1	3	9	8	5	4	2
F29	mean	2.67E+01	9.50E+01	1.16E+02	**2.49E+01**	8.29E+01	9.70E+01	7.53E+01	1.46E+02	9.83E+01	4.23E+01
	std.	**6.37E−01**	3.24E+01	1.33E+01	1.74E+01	2.70E+01	3.36E+01	5.88E+00	3.78E+01	2.96E+01	3.57E+01
	best	2.53E+01	6.66E+01	9.56E+01	1.81E+01	6.73E+01	6.74E+01	6.79E+01	8.79E+01	6.81E+01	**1.73E−03**
	Rank	2	6	9	1	5	7	4	10	8	3
F30	mean	2.10E+01	2.09E+01	2.05E+01	2.04E+01	2.06E+01	2.10E+01	2.09E+01	2.10E+01	2.04E+01	**2.03E+01**
	std.	5.71E−02	1.22E−01	4.58E−02	3.56E−02	5.89E−02	5.23E−02	7.24E−02	4.34E−02	**1.56E−02**	3.42E−02
	best	2.09E+01	2.06E+01	2.04E+01	**2.03E+01**	2.05E+01	2.09E+01	2.08E+01	2.09E+01	**2.03E+01**	**2.03E+01**
	Rank	6	5	3	2	4	6	5	6	2	1
F31	mean	6.19E+00	7.67E+00	1.70E+01	1.48E+01	**2.61E−01**	1.54E+01	1.59E+01	1.67E+01	1.62E+01	8.37E+00
	std.	2.22E+00	1.98E+00	6.10E−01	3.97E+00	**4.92E−01**	2.14E+00	2.75E+00	3.45E+00	9.69E−01	3.22E+00
	best	2.75E+00	4.63E+00	1.60E+01	4.01E+00	**0.00E+00**	1.17E+01	1.01E+01	1.23E+01	1.47E+01	5.25E+00
	Rank	2	3	10	5	1	6	7	9	8	4
F32	mean	2.56E−03	1.33E+01	3.92E−03	1.71E−13	**0.00E+00**	6.74E−02	2.75E−02	1.01E+00	4.19E−03	5.41E−03
	std.	6.75E−03	1.31E+01	1.61E−03	8.04E−14	**0.00E+00**	8.69E−02	2.23E−02	1.50E+00	1.32E−02	8.26E−03
	best	7.08E−05	1.48E−02	1.37E−03	1.14E−13	**0.00E+00**	7.84E−12	1.02E−12	1.20E−01	6.62E−07	1.14E−13
	Rank	3	10	4	2	1	8	7	9	5	6
F33	mean	6.60E+01	3.48E+01	5.86E−01	**1.14E−14**	1.99E−01	7.37E+01	8.00E+01	7.67E+01	2.93E+00	1.14E−13
	std.	1.22E+01	5.70E+00	6.20E−01	3.60E−14	4.20E−01	2.11E+01	1.88E+01	2.45E+01	1.46E+00	**0.00E+00**
	best	4.32E+01	2.19E+01	2.65E−04	**0.00E+00**	0.00E+00	5.07E+01	4.48E+01	3.48E+01	1.14E+00	1.14E−13
	Rank	7	6	4	1	3	8	10	9	5	2
F34	mean	1.53E+02	5.53E+01	7.79E+01	5.82E+01	7.70E+01	7.68E+01	8.00E+01	9.84E+01	5.95E+01	**4.32E+01**
	std.	1.61E+01	1.54E+01	1.49E+01	7.79E+00	1.68E+01	1.08E+01	1.42E+01	3.08E+01	7.94E+00	**7.61E+00**
	best	1.21E+02	3.51E+01	5.52E+01	4.78E+01	4.47E+01	5.87E+01	5.87E+01	6.41E+01	4.56E+01	**2.70E+01**
	Rank	10	2	7	3	6	5	8	9	4	1
F35	mean	1.87E+03	9.70E+02	2.50E+01	**1.69E−01**	2.56E+00	1.72E+03	1.57E+03	2.39E+03	3.22E+01	1.08E+01
	std.	4.44E+02	3.47E+02	5.21E+00	**5.12E−01**	4.09E+00	7.41E+02	4.26E+02	4.71E+02	6.57E+00	5.07E+00
	best	1.35E+03	3.70E+02	1.83E+01	**0.00E+00**	3.39E−02	4.64E+02	9.52E+02	1.59E+03	2.15E+01	5.12E+00
	Rank	9	6	4	1	2	8	7	10	5	3
F36	mean	5.76E+03	3.08E+03	3.24E+03	2.88E+03	4.11E+03	6.71E+03	6.46E+03	3.39E+03	2.56E+03	**2.31E+03**
	std.	3.43E+02	6.19E+02	2.18E+02	**3.06E+02**	4.80E+02	3.92E+02	4.46E+02	5.45E+02	2.56E+02	3.17E+02
	best	5.23E+03	**1.77E+03**	2.98E+03	2.51E+03	3.18E+03	6.01E+03	5.52E+03	2.42E+03	2.12E+03	1.86E+03
	Rank	8	4	5	3	7	10	9	6	2	1
F37	mean	2.62E+00	8.12E−01	5.42E−01	4.96E−01	1.05E+00	2.64E+00	2.60E+00	2.75E+00	4.37E−01	**4.18E−01**
	std.	2.82E−01	6.51E−01	8.53E−02	5.71E−02	1.24E−01	2.47E−01	3.32E−01	2.62E−01	7.85E−02	**5.15E−02**
	best	1.97E+00	3.12E−01	4.12E−01	3.72E−01	7.98E−01	2.13E+00	1.93E+00	2.48E+00	**3.05E−01**	3.62E−01
	Rank	8	5	4	3	6	9	7	10	2	1
F38	mean	3.46E−01	4.35E−01	4.18E−01	3.10E−01	3.09E−01	4.88E−01	4.11E−01	4.71E−01	**2.84E−01**	2.90E−01
	std.	**3.30E−02**	1.07E−01	7.14E−02	4.77E−02	3.58E−02	1.14E−01	6.86E−02	1.13E−01	4.68E−02	4.34E−02
	best	2.89E−01	2.55E−01	2.77E−01	2.58E−01	2.62E−01	3.50E−01	2.97E−01	3.09E−01	**2.06E−01**	2.10E−01
	Rank	5	8	7	4	3	10	6	9	1	2
F39	mean	3.07E−01	8.91E−01	3.44E−01	2.95E−01	2.81E−01	2.88E−01	2.78E−01	2.88E−01	2.45E−01	**2.30E−01**
	std.	3.63E−02	3.59E−01	3.38E−02	2.67E−02	**2.64E−02**	4.70E−02	4.42E−02	4.92E−02	4.02E−02	2.69E−02
	best	2.54E−01	1.88E−01	2.97E−01	2.56E−01	2.25E−01	2.11E−01	**1.84E−01**	2.26E−01	2.03E−01	1.87E−01
	Rank	7	9	8	6	4	5	3	5	2	1
F40	mean	1.58E+01	**4.49E+00**	1.09E+01	5.81E+00	9.71E+00	1.80E+01	1.89E+01	3.75E+01	7.06E+00	5.93E+00
	std.	9.26E−01	**6.87E−01**	9.63E−01	8.14E−01	1.28E+00	5.93E+00	8.66E+00	2.19E+01	1.07E+00	1.22E+00
	best	1.41E+01	**3.18E+00**	9.09E+00	4.24E+00	7.32E+00	1.03E+01	9.75E+00	1.54E+01	5.16E+00	3.87E+00
	Rank	7	1	6	2	5	8	9	10	4	3
F41	mean	1.18E+01	1.05E+01	1.11E+01	1.06E+01	1.15E+01	1.20E+01	1.19E+01	1.11E+01	1.07E+01	**1.02E+01**
	std.	2.50E−01	8.64E−01	2.31E−01	**1.90E−01**	2.07E−01	4.35E−01	3.94E−01	6.62E−01	2.71E−01	4.55E−01
	best	1.13E+01	**8.55E+00**	1.07E+01	1.03E+01	1.13E+01	1.14E+01	1.14E+01	1.01E+01	1.01E+01	9.37E+00
	Rank	7	2	5	3	6	9	8	5	4	1
F42	mean	3.76E+05	1.36E+05	2.07E+06	3.16E+04	**1.33E+04**	1.91E+05	2.11E+05	1.67E+05	1.54E+05	4.42E+04
	std.	1.12E+05	1.04E+05	6.62E+05	3.34E+04	**6.50E+03**	1.73E+05	9.21E+04	2.13E+05	8.75E+04	3.70E+04
	best	2.44E+05	2.69E+04	1.07E+06	2.49E+03	**1.73E+03**	3.29E+04	8.02E+04	3.54E+04	4.92E+04	8.62E+03
	Rank	9	4	10	2	1	7	8	6	5	3

(*continued on next page*)

Table 5 (*continued*)

F	AL.	PSOFIPS	PSOFDR	CLPSO	jDE	SaDE	TLBO	ETLBO	DGSTLBO	BSA	LBSA
F43	mean	1.59E+03	3.38E+03	4.92E+02	2.63E+02	**6.89E+01**	2.93E+03	3.31E+03	8.71E+02	9.10E+02	1.14E+03
	std.	8.50E+02	4.21E+03	9.30E+01	5.11E+02	**3.43E+01**	2.35E+03	4.11E+03	1.02E+03	1.05E+03	1.29E+03
	best	4.78E+02	3.13E+02	3.44E+02	**1.35E+01**	3.09E+01	1.35E+02	1.64E+02	6.17E+01	4.35E+01	6.23E+01
	Rank	7	10	3	2	1	8	9	4	5	6
F44	mean	1.19E+01	1.57E+01	1.01E+01	1.41E+01	**5.55E+00**	2.12E+01	2.12E+01	2.71E+01	6.91E+00	6.31E+00
	std.	8.29E−01	2.33E+01	1.02E+00	1.13E+00	6.59E−01	2.55E+01	2.54E+01	2.86E+01	**6.25E−01**	1.01E+00
	best	1.01E+01	**4.44E+00**	8.20E+00	1.19E+01	4.74E+00	5.77E+00	6.72E+00	9.65E+00	6.06E+00	4.60E+00
	Rank	5	7	4	6	1	8	8	9	3	2
F45	mean	5.81E+03	7.05E+03	5.95E+03	**3.17E+01**	4.31E+01	1.54E+03	1.44E+03	4.28E+02	1.68E+02	9.02E+01
	std.	2.71E+03	1.27E+04	3.35E+03	3.78E+01	**2.96E+01**	9.69E+02	6.69E+02	1.77E+02	1.91E+02	8.20E+01
	best	2.36E+03	1.37E+03	2.48E+03	**1.08E+01**	1.91E+01	6.68E+02	3.41E+02	2.03E+02	2.75E+01	3.06E+01
	Rank	8	10	9	1	2	7	6	5	4	3
F46	mean	1.49E+05	6.77E+04	3.39E+05	3.66E+03	**2.49E+03**	9.72E+04	1.19E+05	2.20E+04	6.20E+03	9.86E+03
	std.	6.40E+04	4.21E+04	1.22E+05	3.15E+03	**2.80E+03**	9.01E+04	1.00E+05	2.22E+04	3.02E+03	1.32E+04
	best	8.47E+04	8.85E+03	1.25E+05	6.53E+02	**2.21E+02**	1.89E+04	3.16E+04	4.09E+03	2.22E+03	1.22E+03
	Rank	9	6	10	2	1	7	8	5	3	4
F47	mean	2.25E+02	2.27E+02	2.70E+02	2.96E+02	1.36E+02	2.81E+02	3.27E+02	3.14E+02	1.79E+02	**1.14E+02**
	std.	7.27E+01	9.97E+01	**5.98E+01**	8.12E+01	7.92E+01	1.08E+02	1.33E+02	1.41E+02	8.11E+01	7.56E+01
	best	1.67E+02	1.44E+02	1.59E+02	9.35E+01	2.51E+01	1.64E+02	5.82E+01	1.49E+02	5.06E+01	**2.41E+01**
	Rank	4	5	6	8	2	7	10	9	3	1
F48	mean	**3.14E+02**	3.18E+02	3.15E+02	**3.14E+02**	3.15E+02	3.15E+02	3.15E+02	3.15E+02	3.15E+02	3.15E+02
	std.	1.57E−04	3.50E+00	2.29E−01	2.89E−13	**0.00E+00**	1.23E−11	1.78E−10	4.43E−01	2.92E−07	1.44E−13
	best	**3.14E+02**	3.15E+02	3.15E+02	**3.14E+02**	3.15E+02	3.15E+02	3.15E+02	3.15E+02	3.15E+02	3.15E+02
	Rank	1	3	2	1	2	2	2	2	2	2
F49	mean	2.24E+02	2.27E+02	2.27E+02	2.27E+02	2.25E+02	**2.00E+02**	**2.00E+02**	**2.00E+02**	2.27E+02	2.26E+02
	std.	5.46E−01	2.58E+00	1.01E+00	4.59E+00	5.21E−01	2.20E−03	2.19E−03	**9.68E−04**	2.42E+00	1.28E+00
	best	2.23E+02	2.23E+02	2.25E+02	2.24E+02	2.24E+02	**2.00E+02**	**2.00E+02**	**2.00E+02**	2.25E+02	2.24E+02
	Rank	2	5	5	5	3	1	1	1	5	4
F50	mean	2.07E+02	2.06E+02	2.10E+02	**2.00E+02**	2.06E+02	2.01E+02	**2.00E+02**	2.02E+02	2.07E+02	2.09E+02
	std.	2.45E+00	2.09E+00	1.62E+00	1.46E−01	2.49E+00	2.39E+00	**5.33E−06**	3.62E+00	6.48E−01	3.10E+00
	best	2.04E+02	2.04E+02	2.08E+02	**2.00E+02**	2.03E+02	**2.00E+02**	**2.00E+02**	**2.00E+02**	2.06E+02	2.04E+02
	Rank	5	4	7	1	4	2	1	3	5	6
F51	mean	1.70E+02	1.51E+02	**1.00E+02**	**1.00E+02**	**1.00E+02**	**1.10E+02**	**1.00E+02**	1.10E+02	**1.00E+02**	**1.00E+02**
	std.	4.83E+01	5.18E+01	**3.00E−02**	6.12E−02	7.09E−02	3.15E+01	1.57E−01	3.15E+01	5.83E−02	7.52E−02
	best	**1.00E+02**	**1.00E+02**	**1.00E+02**	**1.00E+02**	**1.00E+02**	**1.00E+02**	**1.00E+02**	**1.00E+02**	**1.00E+02**	**1.00E+02**
	Rank	4	3	1	1	1	2	1	2	1	1
F52	mean	4.34E+02	7.12E+02	4.38E+02	5.65E+02	**3.62E+02**	5.47E+02	6.56E+02	7.94E+02	4.09E+02	4.34E+02
	std.	3.50E+01	1.84E+02	**1.33E+01**	1.73E+02	5.04E+01	1.66E+02	2.13E+02	2.15E+02	3.71E+00	6.71E+01
	best	3.76E+02	4.02E+02	4.17E+02	4.01E+02	3.00E+02	4.02E+02	4.02E+02	4.06E+02	4.04E+02	**3.71E+02**
	Rank	3	8	4	6	1	5	7	9	2	3
F53	mean	3.98E+02	1.48E+03	9.21E+02	**3.83E+02**	8.51E+02	1.24E+03	1.18E+03	1.43E+03	8.77E+02	8.34E+02
	std.	1.16E+01	3.70E+02	3.47E+01	**5.59E+00**	2.76E+01	3.51E+02	2.30E+02	4.37E+02	1.66E+01	3.87E+01
	best	3.89E+02	1.02E+03	8.64E+02	**3.76E+02**	8.06E+02	9.20E+02	8.94E+02	9.96E+02	8.54E+02	7.87E+02
	Rank	2	10	6	1	4	8	7	9	5	3
F54	mean	**2.14E+02**	6.65E+06	3.11E+04	2.15E+02	9.67E+02	3.31E+06	3.93E+06	3.08E+06	1.41E+03	1.29E+03
	std.	**1.02E+00**	1.10E+07	1.52E+04	1.39E+00	1.47E+02	5.39E+06	5.14E+06	4.99E+06	1.89E+02	5.91E+02
	best	**2.12E+02**	2.17E+03	7.11E+03	2.13E+02	6.69E+02	1.25E+03	1.44E+03	9.91E+02	1.21E+03	7.75E+02
	Rank	1	10	6	2	3	8	9	7	5	4
F55	mean	7.00E+02	1.82E+04	7.94E+03	**3.87E+02**	1.15E+03	3.35E+03	3.37E+03	6.47E+03	2.55E+03	1.93E+03
	std.	8.54E+01	3.03E+04	2.72E+03	**6.82E+01**	5.64E+02	1.35E+03	1.75E+03	3.43E+03	7.49E+02	5.26E+02
	best	5.98E+02	1.78E+03	5.62E+03	**2.92E+02**	5.90E+02	1.62E+03	1.58E+03	3.37E+03	1.42E+03	1.15E+03
	Rank	2	10	9	1	3	6	7	8	5	4
Av. rank		5.63	6.13	6.07	**2.70**	3.00	6.57	6.30	6.97	4.00	**2.70**

Training an ANN means optimising its weights and bias. For a three-layer ANN ($n \times p \times q$), the number of variables that should be optimised is $(n+1)p+(p+1)q$, and the mean sum of squared errors (MSE) over all training patterns are often chosen as the objective function of optimisation algorithms, calculated as follows:

$$MSE = \frac{1}{mn}\sum_{i=1}^{m}\sum_{j=1}^{n}\frac{1}{2}(d_{ij} - y_{ij})^2 \tag{17}$$

where m is the size of the training data, n is the number of outputs, d_{ij} is the desired output, and y_{ij} is the output of the ANN.

Table 6
Comparison between LBSA and other algorithms using a *t*-test for 30D functions in CEC2014.

F	AL.	jDE	SaDE	PSOwFIPS	PSOFDR	CLPSO	TLBO	ETLBO	DGSTLBO	BSA
F26	T	0.93373	**2.56475**	**18.86069**	**6.88243**	**36.57498**	**3.72942**	1.12409	**6.44030**	**6.45541**
	P	0.35431	**0.01294**	**0.00000**	**0.00000**	**0.00000**	**0.00044**	0.26561	**0.00000**	**0.00000**
F27	T	−6.59912	−11.75755	**11.75633**	**2.32644**	**9.29153**	**4.56562**	**4.17978**	**2.26966**	**3.08124**
	P	0.00000	0.00000	**0.00000**	**0.02351**	**0.00000**	**0.00003**	**0.00010**	**0.02696**	**0.00315**
F28	T	−9.66465	1.77922	**7.55034**	**4.88356**	**3.50760**	**7.39477**	**11.46202**	**4.69504**	**3.32637**
	P	0.00000	0.08044	**0.00000**	**0.00001**	**0.00088**	**0.00000**	**0.00000**	**0.00002**	**0.00153**
F29	T	−2.40107	**4.96405**	−2.39638	**5.99467**	**10.63308**	**6.10876**	**4.99903**	**10.87966**	**6.61401**
	P	0.01957	**0.00001**	0.01980	**0.00000**	**0.00000**	**0.00000**	**0.00001**	**0.00000**	**0.00000**
F30	T	**5.39965**	**21.32020**	**55.64948**	**24.24835**	**14.66970**	**56.32828**	**43.73363**	**65.34210**	**7.15584**
	P	**0.00000**	**0.00000**	**0.00000**	**0.00000**	**0.00000**	**0.00000**	**0.00000**	**0.00000**	**0.00000**
F31	T	**6.84351**	−13.62043	−3.05097	−1.01696	**14.46031**	**9.92347**	**9.67976**	**9.66437**	**12.68247**
	P	**0.00000**	0.00000	0.00344	0.31340	**0.00000**	**0.00000**	**0.00000**	**0.00000**	**0.00000**
F32	T	−3.58973	−3.58973	−1.46667	**5.57656**	−0.97108	**3.88951**	**5.09523**	**3.66324**	−0.43106
	P	0.00068	0.00068	0.14787	**0.00000**	0.33554	**0.00026**	**0.00000**	**0.00054**	0.66802
F33	T	−15.58846	**2.59808**	**29.66300**	**33.38307**	**5.17855**	**19.13046**	**23.26561**	**17.11579**	**10.96053**
	P	0.00000	**0.01186**	**0.00000**	**0.00000**	**0.00000**	**0.00000**	**0.00000**	**0.00000**	**0.00000**
F34	T	**7.53203**	**10.03933**	**33.90337**	**3.84690**	**11.37620**	**13.92505**	**12.49798**	**9.52840**	**8.13697**
	P	**0.00000**	**0.00000**	**0.00000**	**0.00030**	**0.00000**	**0.00000**	**0.00000**	**0.00000**	**0.00000**
F35	T	−11.39133	−6.89722	**22.97610**	**15.14510**	**10.70980**	**12.63964**	**20.04069**	**27.63079**	**14.13708**
	P	0.00000	0.00000	**0.00000**	**0.00000**	**0.00000**	**0.00000**	**0.00000**	**0.00000**	**0.00000**
F36	T	**6.96727**	**17.03972**	**40.35873**	**5.99338**	**13.20624**	**47.68523**	**41.46806**	**9.33609**	**3.34007**
	P	**0.00000**	**0.00000**	**0.00000**	**0.00000**	**0.00000**	**0.00000**	**0.00000**	**0.00000**	**0.00147**
F37	T	**5.61489**	**25.85856**	**42.18177**	**3.30937**	**6.81484**	**48.12225**	**35.62649**	**47.84830**	1.11279
	P	**0.00000**	**0.00000**	**0.00000**	**0.00161**	**0.00000**	**0.00000**	**0.00000**	**0.00000**	0.27039
F38	T	1.69894	1.78434	**5.59128**	**6.88176**	**8.33642**	**8.88374**	**8.12090**	**8.16238**	−0.51924
	P	0.09469	0.07960	**0.00000**	**0.00000**	**0.00000**	**0.00000**	**0.00000**	**0.00000**	0.60557
F39	T	**9.36429**	**7.37392**	**9.28330**	**10.05893**	**14.35105**	**5.84382**	**5.00058**	**5.60192**	1.65264
	P	**0.00000**	**0.00000**	**0.00000**	**0.00000**	**0.00000**	**0.00000**	**0.00001**	**0.00000**	0.10381
F40	T	−0.43678	**11.69433**	**35.33066**	−5.60942	**17.46573**	**10.89866**	**8.13816**	**7.87328**	**3.83052**
	P	0.66389	**0.00000**	**0.00000**	0.00000	**0.00000**	**0.00000**	**0.00000**	**0.00000**	**0.00032**
F41	T	**4.35862**	**14.22154**	**16.74675**	1.48492	**9.80169**	**15.66797**	**14.96744**	**5.80590**	**4.72921**
	P	**0.00005**	**0.00000**	**0.00000**	0.14298	**0.00000**	**0.00000**	**0.00000**	**0.00000**	**0.00001**
F42	T	−1.38252	−4.50767	**15.43646**	**4.52930**	**16.69730**	**4.56234**	**9.21876**	**3.09944**	**6.34291**
	P	0.17211	0.00003	**0.00000**	**0.00003**	**0.00000**	**0.00003**	**0.00000**	**0.00299**	**0.00000**
F43	T	−3.47194	−4.55733	1.60336	**2.79230**	−2.75394	**3.65178**	**2.75586**	−0.90215	−0.76350
	P	0.00098	0.00003	0.11429	**0.00708**	0.00785	**0.00056**	**0.00781**	0.37071	0.44826
F44	T	**28.06442**	−3.44540	**23.40168**	**2.21312**	**14.57672**	**3.18891**	**3.21238**	**3.97707**	**2.74686**
	P	**0.00000**	0.00107	**0.00000**	**0.03084**	**0.00000**	**0.00230**	**0.00215**	**0.00020**	**0.00800**
F45	T	−3.55284	−2.96160	**11.57150**	**3.00856**	**9.59609**	**8.14252**	**10.92957**	**9.49231**	**2.05113**
	P	0.00077	0.00443	**0.00000**	**0.00388**	**0.00000**	**0.00000**	**0.00000**	**0.00000**	**0.04478**
F46	T	−2.50497	−2.99652	**11.65085**	**7.18729**	**14.61425**	**5.25779**	**5.87732**	**2.57563**	−1.48264
	P	0.01508	0.00401	**0.00000**	**0.00000**	**0.00000**	**0.00000**	**0.00000**	**0.01258**	0.14358
F47	T	**8.98665**	1.10043	**5.81391**	**4.96272**	**8.83864**	**6.94172**	**7.63707**	**6.84961**	**3.22617**
	P	**0.00000**	0.27569	**0.00000**	**0.00001**	**0.00000**	**0.00000**	**0.00000**	**0.00000**	**0.00206**
F48	T	−2.08995E+13	−2.15758	−4.30336E+04	**4.11727**	**3.21777**	**5.66938**	**2.21585**	**3.02585**	**4.40829**
	P	0.00000	0.03511	0.00000	**0.00012**	**0.00212**	**0.00000**	**0.03064**	**0.00369**	**0.00005**
F49	T	1.96584	−4.13892	−6.95976	1.79700	**4.52992**	−109.24755	−109.25044	−109.27495	**2.25320**
	P	0.05411	0.00011	0.00000	0.07754	**0.00003**	0.00000	0.00000	0.00000	**0.02804**
F50	T	−14.24590	−3.54490	−1.84809	−3.64301	**2.49945**	−10.93030	−15.14135	−7.94073	−3.15544
	P	0.00000	0.00078	0.06969	0.00058	**0.01529**	0.00000	0.00000	0.00000	0.00254
F51	T	0.65422	0.50525	**7.94725**	**5.35959**	**8.14779**	1.75888	**5.25514**	1.76198	−0.41303
	P	0.51556	0.61530	**0.00000**	**0.00000**	**0.00000**	0.08387	**0.00000**	0.08334	0.68111
F52	T	**3.85682**	−4.72137	−0.02531	**7.75919**	0.29628	**3.43129**	**5.45158**	**8.73187**	−2.03530
	P	**0.00029**	0.00002	0.97989	**0.00000**	0.76807	**0.00111**	**0.00000**	**0.00000**	0.04640
F53	T	−63.20117	1.99838	−59.24825	**9.51431**	**9.19405**	**6.24567**	**8.04197**	**7.39735**	**5.59681**
	P	0.00000	0.05037	0.00000	**0.00000**	**0.00000**	**0.00000**	**0.00000**	**0.00000**	**0.00000**
F54	T	−10.02046	−2.94965	−10.02816	**3.30750**	**10.76868**	**3.35860**	**4.19197**	**3.37793**	1.03517
	P	0.00000	0.00458	0.00000	**0.00162**	**0.00000**	**0.00139**	**0.00010**	**0.00131**	0.30489
F55	T	−15.96627	−5.53732	−12.67155	**2.93465**	**11.88726**	**5.35552**	**4.30672**	**7.16346**	**3.69768**
	P	0.00000	0.00000	0.00000	**0.00478**	**0.00000**	**0.00000**	**0.00006**	**0.00000**	**0.00048**
B		10	10	19	25	27	27	27	26	20
W		14	14	7	2	1	2	2	2	2
S		6	6	4	3	2	1	1	2	8

Table 7
Comparisons of five algorithms for SISO system over 30 runs.

merits	BSA	CLPSO	TLBO	jDE	LBSA
best	1.1248E−04	3.6081E−04	**5.6342E−06**	9.5704E−04	3.4287E−05
mean	2.6561E−04	1.5722E−03	1.1618E−03	1.6750E−03	**2.5628E−04**
Std.	**1.1451E−04**	1.6350E−03	9.0796E−04	3.4113E−04	2.7032E−04
MT(sec)	3.9780E+01	4.5516E+01	3.0174E+01	**2.33404 E+01**	3.0302E+01
TStd.	3.4299E+00	1.8671E+00	5.4764E−01	6.0298E−01	**1.0252E−01**
SR	30	24	30	27	30

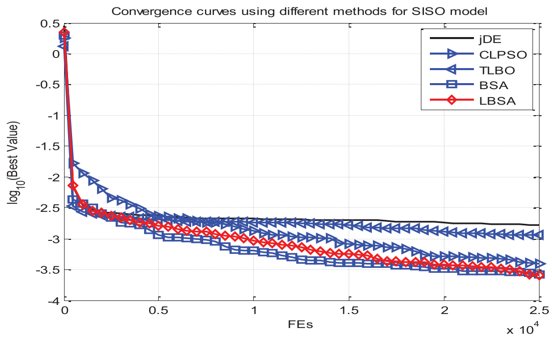

Fig. 5. The average fitness curves of five algorithms for SISO system.

6.1. Nonlinear system identification

A. SISO nonlinear system

The SISO nonlinear system used in this experiment is given as follows:

$$y = \sin(2x)e^{-2x} \tag{18}$$

In this experiment, the size of network was $1 \times 5 \times 1$, and the number of variables that must be determined was 16. In addition, 124 pairs of data from Eq. 18 were chosen as the training samples. To clearly show the results, only five optimisation algorithms were used to optimise the neural network. The population size of all algorithms was 50, the maximum FE was 25,000, and the other parameters of the different algorithms were the same as those used in CEC2005. Table 7 shows the training results of the five algorithms over 30 independent runs. Here, 'best' is the best value of the MSE, 'mean' is the average best value of the MSE, 'Std.' is the standard deviation of the best solutions, and 'MT' is the mean convergence time of the algorithms to an acceptable solution. If the algorithm did not converge to an optimal solution in all 30 runs, 'MT' is the mean time of the algorithm in which it was ended by the maximum FE. Furthermore, 'TStd.' is the standard deviation of the time, and 'SR' is the successful ratio of 30 independent runs. The acceptable solution is 0.002. The table shows that the best and mean MSE of LBSA was the smallest, and the Std. of BSA was the smallest of the five algorithms. The MT of jDE was the smallest, and the TStd. of LBSA was the smallest. The successful ratios of four algorithms were 100%. The convergence process according to the FEs is shown in Fig. 5, which shows that the convergence speed of BSA was faster than that of the other algorithms, and the average convergence accuracy of LBSA was higher than that of the other algorithms. The training samples of different algorithms were the same. To show the modelling effectiveness of different algorithms, the ANNs with best parameters were simulated. The modelling curves with the best parameters of different algorithms are displayed in Fig. 6, and the absolute errors are also shown in the corresponding curves for the algorithms. Fig. 6 shows that the testing error of ANN with LBSA was better than that of the other four algorithms.

B. MISO nonlinear system

The MISO nonlinear system used in this experiment is given as follows:

$$y(t+1) = \frac{y(t)y(t-1)[y(t)+2.5]}{1+y^2(t)+y^2(t-1)} + u(t)$$
$$u(t) = \sin(\frac{2\pi t}{25}) \tag{19}$$

The size of the neural network was $3 \times 5 \times 1$, and the number of the variables of optimisation algorithms was 26. In addition, 200 pairs of data from Eq. 19 were chosen as the training samples. The training parameters of the five algorithms were the same as those used in the SISO system. The acceptable MSE is 0.002. Table 8 shows the results of the five different algorithms. The results show that the MSEs in terms of the best, mean, and Std. with LBSA were the smallest among the five algorithms. The training time of jDE was the shortest, and the TStd. of BSA was the smallest. TLBO and LBSA had higher success rates than the other algorithms. Fig. 7 shows the convergence processes of the five algorithms. This figure shows that

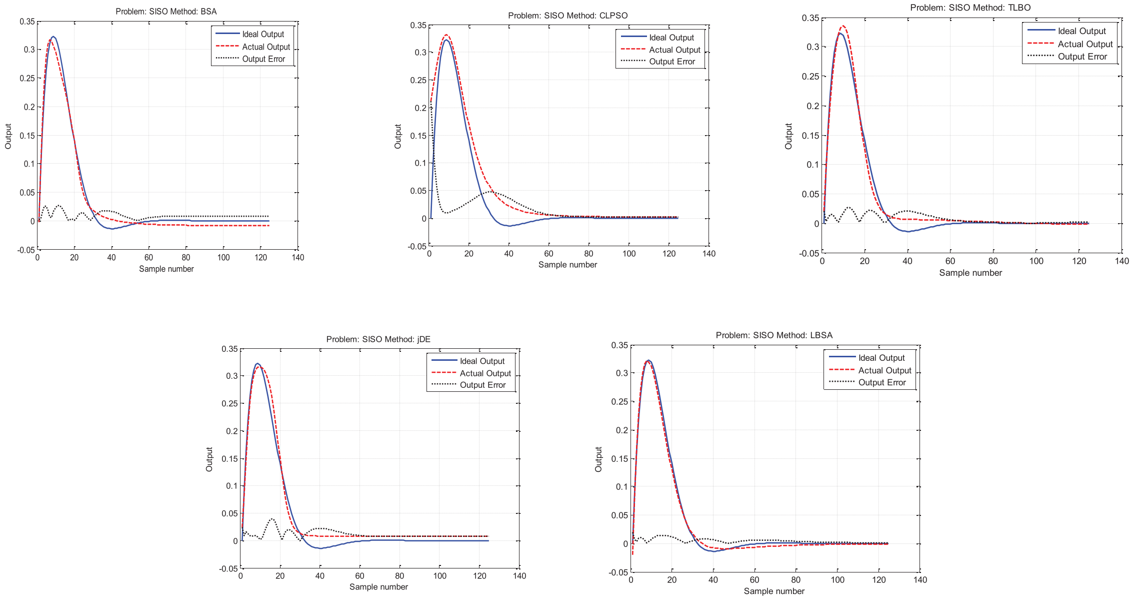

Fig. 6. The model and absolute error with optimized parameters of different algorithms for SISO system.

Table 8
Comparisons of five algorithms for MISO system over 30 runs.

Algorithms	BSA	CLPSO	TLBO	jDE	LBSA
best	8.8840E−03	1.6381E−02	7.3263E−04	2.1372E−03	**6.7464E−04**
mean	1.5097E−02	3.5236E−02	6.1081E−03	8.7818E−03	**3.3982E−03**
Std.	3.3759E−03	1.9407E−02	4.4069E−03	5.2586E−03	**1.8934E−03**
MT(sec)	26.8041	29.1934	29.6413	**20.3282**	26.8631
TStd.	**1.1074E−01**	1.7947E−01	1.1810E+00	1.4113E−01	1.2264E−01
SR	0	0	11	0	15

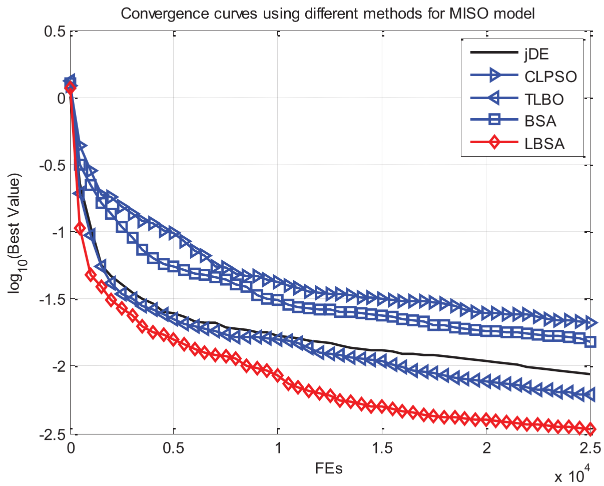

Fig. 7. The average fitness curves of five algorithms for MISO system.

the convergence speed of LBSA was higher than that of the other algorithms. The modelling curves with the best solutions of different algorithms are shown in Fig. 8, which shows that the absolute errors of LBSA and TLBO were relatively smaller than those of the other algorithms.

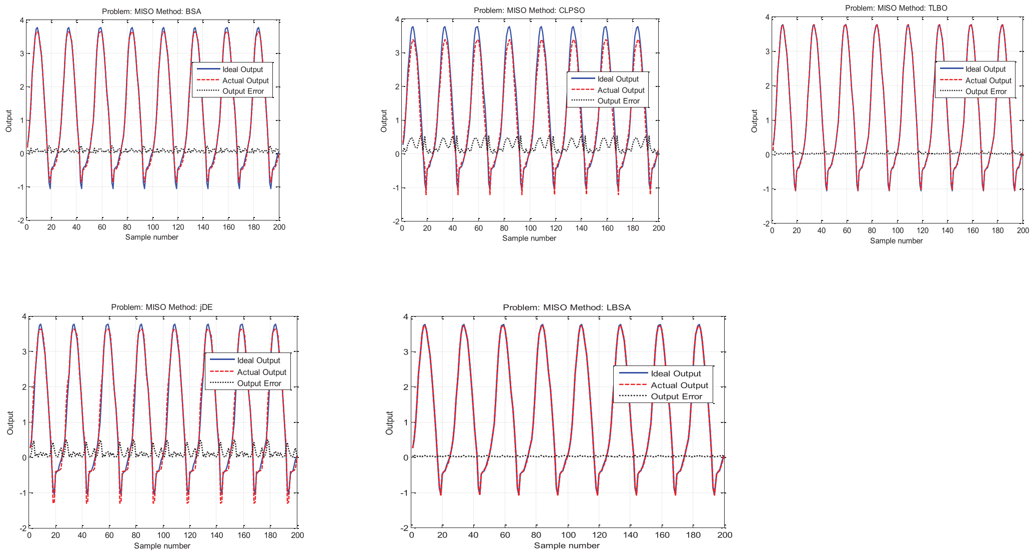

Fig. 8. The model and absolute error with optimized parameters of different algorithms for MISO system.

Table 9
Comparisons of five algorithms for MG model over 30 runs.

Algorithms		BSA	CLPSO	TLBO	jDE	LBSA
Training error	best	8.2669E−04	2.1381E−03	**3.2300E−04**	4.5339E−04	4.1732E−04
	mean	1.6452E−03	3.7599E−03	**5.0356E−04**	7.8107E−04	6.4917E−04
	Std.	4.9096E−04	1.1021E−03	**1.9239E−04**	2.1723E−04	2.4519E−04
Testing error	best	6.6138E−04	2.0534E−03	**2.4551E−04**	3.0173E−04	2.9654E−04
	mean	1.3129E−03	3.4043E−03	**4.0320E−04**	6.1615E−04	5.2960E−04
	Std.	4.2395E−04	9.1403E−04	**1.5625E−04**	2.1616E−04	2.3275E−04
Training time	MT(sec)	99.5724	106.4755	103.3196	**81.5764**	99.6394
	TStd.	**3.3169E−01**	7.3082E−01	4.6065E+00	5.9686E−01	3.6864E−01
SR		21	0	30	23	30

6.2. Chaotic time series prediction

A. Mackey-Glass (MG) chaotic time series

The Mackey-Glass chaotic system [33,44] is described as follows:

$$\dot{x}(t) = \frac{ax(t-\tau)}{1+x^{10}(t-\tau)} - bx(t) \tag{20}$$

where $\tau = 17$, $a = 0.2$, and $b = 0.1$. In general, the earlier four points $x(t-18)$, $x(t-12)$, $x(t-6)$, and $x(t)$ are used to predict the next point $x(t+1)$. The goal of this experiment is to build the single-step-ahead prediction model of MG, as shown in Eq. 21:

$$\hat{x}(t+1) = f(x(t-18), x(t-12), x(t-6), x(t)) \tag{21}$$

To predict the MG model, the ANN should contain four input units and one output unit, and the units of the hidden layer in this experiment was five. The population size of all algorithms was 50, and the maximum FE was 25,000. The remaining parameters were the same as those described in Section 6.1. The 1000 pairs of data from the model of Eq. 20 were chosen as the training and testing samples. The former 500 pairs of data were used as training samples, and the rest were selected as testing samples. The former 17 states were randomly generated. MSE was chosen as the fitness function of all algorithms, and the acceptable solution is 0.002. The merits in terms of mean, Std., success rate, and running time of the training and testing process are shown in Table 9. These results come from 30 independent runs of each of the five algorithms. The table shows that TLBO generally outperformed the other algorithms. The mean of the training error and testing error obtained by LBSA was ranked second of the five algorithms. The success rate of TLBO and LBSA was 100%. The training time of jDE was the smallest, and the TStd. of BSA was the smallest of the five algorithms. The table also shows that the performance of LBSA was worse than that of TLBO for the MG model. The convergence process of the average MSE for all algorithms is displayed in Fig. 9. This figure indicates that the convergence speed of TLBO was faster than that of the other algorithms for the MG model. LBSA was the second fastest in the experiment. The training and testing curves for 1000 samples are shown in Fig. 10. These results indicate that the approximate accuracy of TLBO and LBSA was high.

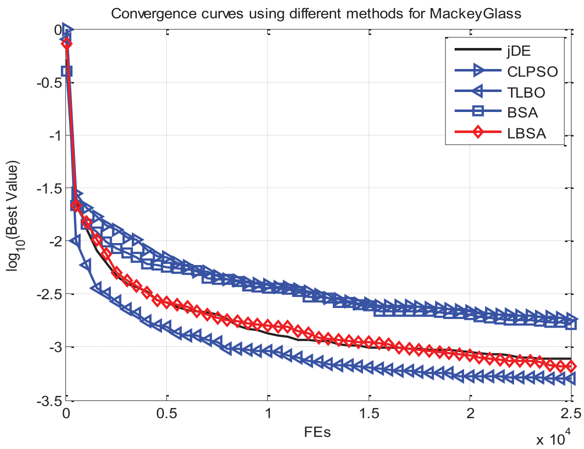

Fig. 9. The convergence curves of five algorithms for MG model.

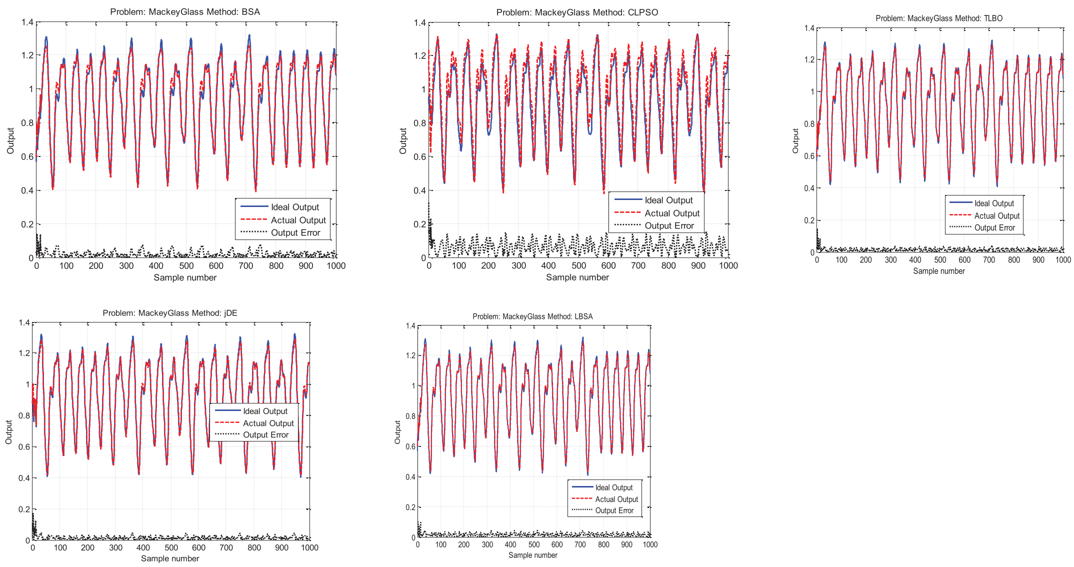

Fig. 10. The training and testing results with optimized ANN for MG model with different algorithms.

B. Box-Jenkins chaotic time series

The second chaotic system is the Box-Jenkins gas furnace model. The Box-Jenkins gas furnace Dataset was recorded from a combustion process of a methane-air mixture. Here, 296 pairs of data (x(t), u(t)) from t = 1 to 296 were chosen as the training and testing samples, and all the data sets were normalised. In addition, x(t) was the output CO_2 concentration, and u(t) was the input gas flow rate. Further, $u(t-4)$ and $x(t-1)$ were commonly used to predict the next state $x(t)$ as follows:

$$\hat{x}(t) = f(u(t-4), x(t-1)) \tag{22}$$

The configuration of the ANN was two input units, five hidden units, and one output unit. The first 142 points were selected as the training samples, and the other 150 were used as the testing samples. The training parameters of all algorithms were the same as those used in the MG model experiment. The results of all algorithms are shown in Table 10. Fig. 11 represents the changes in average MSE for 30 independent runs for the five algorithms. The training and testing results with the optimal parameters of ANN for different algorithms are shown in Fig. 12. The table shows that LBSA outperformed the other algorithms with respect to training and testing errors. The mean training time of jDE was the smallest of all the algorithms, and the TStd. of TLBO was the smallest. Four algorithms converged to an acceptable solution. Figs. 11 and 12 indicate that the convergence speed of LBSA was better than that of the other algorithms and that LBSA achieved comparable performance for training and testing samples.

Table 10
Comparisons of five algorithms for BJ model over 30 runs.

Algorithms		BSA	CLPSO	TLBO	jDE	LBSA
Training error	best	2.6298E−04	6.3800E−04	2.2648E−04	2.3587E−04	**2.2342E−04**
	mean	3.6289E−04	1.7420E−03	2.7155E−04	2.5739E−04	**2.4860E−04**
	Std.	8.5631E−05	1.0690E−03	7.1154E−05	2.0510E−05	**1.9922E−05**
Testing error	best	8.2501E−04	2.2737E−03	9.9168E−04	9.2444E−04	**1.0593E−04**
	mean	1.2824E−03	3.7696E−03	1.3044E−03	1.1370E−03	**1.1334E−03**
	Std.	2.7461E−04	1.1847E−03	2.5881E−04	1.6602E−04	**1.6730E−04**
Training time	MT(sec)	36.2576	3.6172E+01	36.8254	**28.1255**	36.3184
	TStd.	1.9027E−01	9.5481E−01	**1.3737E−01**	4.4479E−01	2.5097E−01
SR		30	21	30	30	30

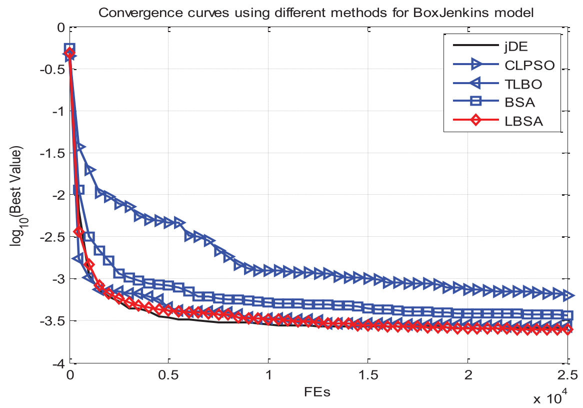

Fig. 11. The training and testing results for MG model with different algorithms.

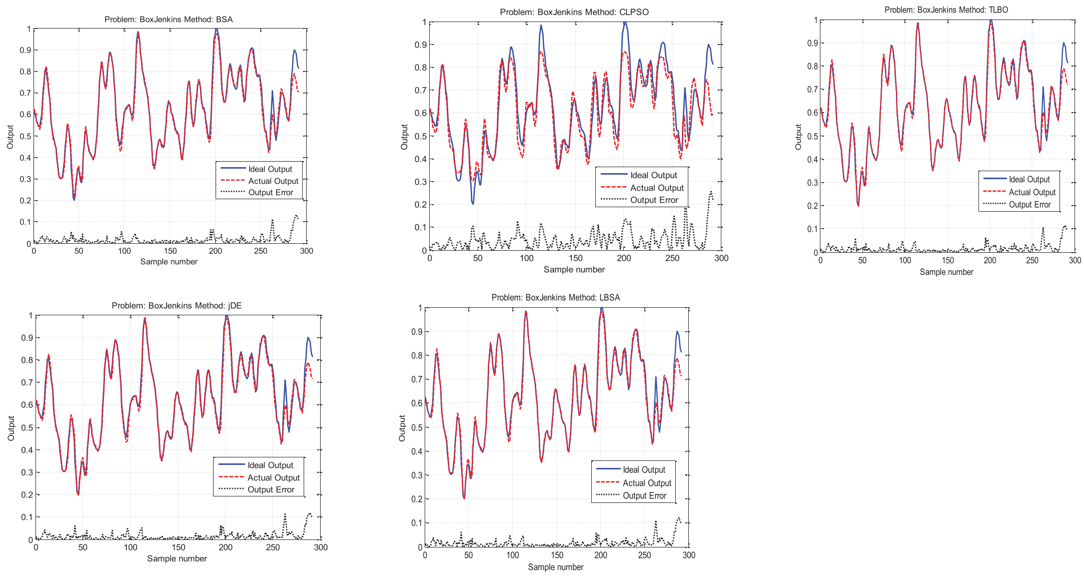

Fig. 12. The training and testing results with optimized ANN for BJ model with different algorithms.

7. Conclusions

In this paper, the basic BSA is extended to LBSA by a learning guidance method that modifies the BSA mutation process. Learning from the best individual is used in the mutation equation to improve the convergence speed of the algorithm, and the diversity of the algorithm is improved by avoiding the worst individual of the current generation. The three learning methods are integrated into one equation. Compared to TLBO, there is only one FE for an individual per iteration. The basic framework of BSA is not changed, and the simple character of BSA is retained. The performance of LBSA was tested on two classical test sets (CEC2005 and CEC2014), and the comparisons with nine other algorithms indicate that LBSA achieved comparable performance in most cases. The testing results for nonlinear system modelling and chaotic time series prediction by ANN also show that LBSA had some challenging performances for most models.

Future work will focus on making full use of the simple structure of BSA by designing a stronger diversity retaining method because LBSA cannot derive the global optimum for some functions in CEC2005 and CEC2014. In addition, extending the application domain of LBSA is an important research direction.

Acknowledgement

This work is partially supported by the National Natural Science Foundation of China (Grant Nos. 61572224, 61304082 and 41475017) and the National Science Fund for Distinguished Young Scholars (Grant No. 61425009). This work is also partially supported by the Major Project of Natural Science Research in Anhui Province (Grant No. KJ2015ZD36), the Natural Science Foundation in Colleges and Universities of Anhui Province (Grant No. KJ2016A639) and International Science and Technology Cooperation project of Anhui Province (Grant No. 10080703003). The authors would like to thank Professor Suganthan for providing PSO codes, and the associate editor and reviewers for their helpful comments.

References

[1] J. Brest, S. Greiner, B. Boskovic, et al., Self-adapting control parameters in differential evolution: a comparative study on numerical benchmark problems, IEEE Trans. Evol. Comput 10 (6) (2006) 646–657.
[2] Q. Cai, M.G. Gong, L.J. Ma, Greedy discrete particle swarm optimization for large-scale social network clustering, Inf. Sci. 316 (2015) 503–516.
[3] D.B. Chen, F. Zou, Z. Li, J.T. Wang, S.W. Li, An improved teaching-learning-based optimization algorithm for solving global optimization problem, Inf. Sci. 297 (2015) 171–190.
[4] D.B. Chen, C.X. Zhao, Particle swarm optimization with adaptive population size and its application, Appl. Soft. Comput. 9 (2009) 39–48.
[5] P. Civicioglu, Backtracking search optimization algorithm for numerical optimization problems, Appl. Math. Comput. 219 (15) (2013) 8121–8144.
[6] M. Ghasemi, M.M. Ghanbarian, S. Ghavidel, S. Rahmani, E.M. Moghaddam, Modified teaching learning algorithm and double differential evolution algorithm for optimal reactive power dispatch problem: a comparative study, Inf. Sci. 278 (2014) 231–249.
[7] M. Ghasemi, S. Ghavidel, M. Gitizadeh, E. Akbari, An improved teaching-learning-based optimization algorithm using Lévy mutation strategy for non-smooth optimal power flow, Electr. Pow. Ener. Syst. 65 (2015) 375–384.
[8] M.G. Gong, Y. Wu, Q. Cai, Discrete particle swarm optimization for high-order graph matching, Inf. Sci. 328 (2016) 158–171.
[9] K. Guney, A. Durmus, S. Basbug, Backtracking search optimization algorithm for synthesis of concentric circular antenna arrays, Int. J. Antenn. Propag. 2014 (2014) Article ID: 250841, 11 pages, http://dx.doi.org/10.1155/2014/250841.
[10] B. Jürgen, H. Jan, Discrete particle swarm optimisation for ontology alignment, Inf. Sci. 192 (2012) 152–173.
[11] H.S. Keesari, R.V. Rao, Optimization of job shop scheduling problems using teaching–learning-based optimization algorithm, Opsearch 51 (4) (2013) 545–561.
[12] J. Kennedy, R.C. Eberhart, Particle swarm optimization, Proc. IEEE Int. Conf. Neural Netw. 4 (1995) 1942–1948.
[13] X.L. Li, X.D. He, A hybrid particle swarm optimization method for structure learning of probabilistic relational models, Inf. Sci. 283 (2014) 258–266.
[14] J.Q. Li, Q.K. Pan, K. Mao, A discrete teaching–learning-based optimization algorithm for realistic flow shop rescheduling problems, Eng. Appl. Artif. Intell. 37 (2015) 279–292.
[15] J.J. Liang, A.K. Qin, P.N. Suganthan, S. Baskar, Comprehensive learning particle swarm optimizer for global optimization of multimodal functions, IEEE Trans. Evol. Comput. 10 (3) (2006) 281–295.
[16] J.J. Liang, B.Y. Qu, P.N. Suganthan, Problem Definitions and Evaluation Criteria for the CEC 2014, 2013, pp. 1–32. Technical Report.
[17] J. Lin, Oppositional backtracking search optimization algorithm for parameter identification of hyper chaotic systems, Nonlinear Dyn. (2014) 1–11.
[18] L.B. Ma, Y.L Zhu, Y. Liu, A novel bionic algorithm inspired by plant root foraging behaviors, Appl. Soft. Comput 37 (2015) 95–113.
[19] M.A. Medina, C.A.C. Coello, J.M. Ramirez, Reactive power handling by a multi-objective teaching learning optimizer based on decomposition, IEEE Trans. Power Syst. 28 (4) (2013) 3629–3637.
[20] R. Mendes, J. Kennedy, J. Neves, The fully informed particle swarm: simpler, maybe better, Appl. Soft. Comput. 8 (3) (2004) 204–210.
[21] H.B. Ouyang, L.Q. Gao, X.Y. Kong, et al., Hybrid harmony search particle swarm optimization with global dimension selection, Inf. Sci (2016) 318–337 346–347.
[22] V. Patel, V. Savsani, Optimization of a plate-fin heat exchanger design throughan improved multi-objective teaching–learning based optimization (MO-ITLBO) algorithm, Chem. Eng. Res. Des. 92 (2014) 2371–2382.
[23] T. Peram, K. Veeramachaneni, C.K. Mohan, Fitness-distance-ratio based particle swarm optimization, in: In Proc. Swarm Intelligence Symp., 2003, pp. 174–181.
[24] A.K. Qin, V.L. Huang, P.N. Suganthan, Differential evolution algorithm with strategy adaptation for global numerical optimization, IEEE Trans. Evol. Comput. 13 (2) (2009) 398–417.
[25] A. Ratnaweera, S. Halgamuge, H. Watson, Self-organizing hierarchical particle swarm optimizer with time varying accelerating coefficients, IEEE Trans. Evol. Comput 8 (3) (2004) 240–255.
[26] R.V. Rao, Review of applications of TLBO algorithm and a tutorial for beginners to solve the unconstrained and constrained optimization problems, Decis. Sci. Letters 5 (2016), doi:10.5267/j.dsl.2015.9.003.
[27] R.V. Rao, V.D. Kalyankar, Multi-pass turning process parameter optimization using teaching–learning-based optimization algorithm, Scientia Iranica 20 (3) (2013) 967–974.
[28] R.V. Rao, V.D. Kalyankar, G. Waghmare, Parameters optimization of selected casting processes using teaching–learning-based optimization algorithm, Appl. Math. Model. 38 (2014) 5592–5608.
[29] R.V. Rao, V. Patel, An elitist teaching–learning-based optimization algorithm for solving complex constrained optimization problems, Int. J. Ind. Eng. Comput. 3 (4) (2012) 535–560.
[30] R.V. Rao, V. Patel, An improved teaching–learning-based optimization algorithm for solving unconstrained optimization problems, Sci. Iran. Trans. D 20 (3) (2013) 710–720.
[31] R.V. Rao, V.J. Savsani, D.P. Vakharia, Teaching–learning-based optimization: an optimization method for continuous non-linear large scale problems, Inf. Sci 183 (1) (2012) 1–15.
[32] P.K. Roy, A. Sur, D.K. Pradhan, Optimal short-term hydro-thermal schedulingusing quasi-oppositional teaching learning based optimization, Eng. Appl. Artif. Intell. 26 (2013) 2516–2524.
[33] B. Samanta, Prediction of chaotic time series using computational intelligence, Expert Syst. Appl. 38 (9) (2011) 11406–11411.
[34] X. Song, X. Zhang, S. Zhao, L. Li, Backtracking search algorithm for effective and efficient surface wave analysis, J. Appl. Geophys. 114 (2015) 19–31.
[35] P.N. Suganthan, N. Hansen, J.J. Liang, Problem definitions and evaluation criteria for the CEC 2005 special session on real-parameter optimization, 2005, pp. 1–50. Technical Report.
[36] C.Y. Tsai, C.J. Chen, A PSO-AB classifier for solving sequence classification problems, Appl. Soft. Comput. 27 (2015) 11–27.

[37] H.F. Wang, I. Moon, S.X. Yang, et.al, A memetic particle swarm optimization algorithm for multimodal optimization problems, Inf. Sci. 197 (2012) 38–52.
[38] H. Wang, H. Sun, C.H. Li, Diversity enhanced particle swarm optimization with neighborhood search, Inf. Sci. 223 (2013) 119–135.
[39] L. Wang, F. Zou, X.H. Hei, An improved teaching–learning-based optimization with neighborhood search for applications of ANN, Neurocomputing 143 (2014) 231–247.
[40] L. Wang, Y. Zhong, Y.Y. Zhao, W. Wang, et al., A hybrid backtracking search optimization algorithm with differential evolution, Math. Probl. Eng. 501 (2015) 769245.
[41] Z.L. Yang, K. Li, A. Foley, C. Zhang, A new self-learning TLBO algorithm for RBF neural modelling of batteries in electric vehicles, 2014 IEEE Cong. on Evol. Comput. (CEC), 2014 July 6-11.
[42] Z.H. Zhan, J. Zhang, Y. Li, et al., Adaptive particle swarm optimization, IEEE Trans. Syst. Man Cy. B 39 (6) (2009) 1362–1381.
[43] C.J. Zhang, Q. Lin, L. Gao, Backtracking Search Algorithm with three constraint handling methods for constrained optimization problems, Expert Syst. Appl. 42 (2015) 7831–7845.
[44] L. Zhao, Y. Yang, PSO-based single multiplicative neuron model for time series prediction, Expert Syst. Appl. 36 (2) (2009) 2805–2812.
[45] F. Zou, L. Wang, X.H. Hei, et al., Teaching-learning-based optimization with dynamic group strategy for global optimization, Inf. Sci. 273 (2014) 112–131.