A Hybrid Bird Mating Optimizer Algorithm with Teaching-Learning-Based Optimization for Global Numerical Optimization

Qingyang Zhang, Guolin Yu*, Hui Song

[1]*Research Institute of Information and System Computation Science, Beifang University of Nationalities, China.*

Abstract   Bird Mating Optimizer (BMO) is a novel meta-heuristic optimization algorithm inspired by intelligent mating behavior of birds. However, it is still insufficient in convergence of speed and quality of solution. To overcome these drawbacks, this paper proposes a hybrid algorithm (TLBMO), which is established by combining the advantages of Teaching-learning-based optimization (TLBO) and Bird Mating Optimizer (BMO). The performance of TLBMO is evaluated on 23 benchmark functions, and compared with seven state-of-the-art approaches, namely BMO, TLBO, Artificial Bee Bolony (ABC), Particle Swarm Optimization (PSO), Fast Evolution Programming (FEP), Differential Evolution (DE), Group Search Optimization (GSO). Experimental results indicate that the proposed method performs better than other existing algorithms for global numerical optimization.

Keywords   Bird Mating Optimizer, Teaching-Learning-Based Optimization, Hybrid Mechanism, Global Numerical Optimization

1. Introduction

As an open and demanding problem, global numerical optimization is of a great importance in various real-word areas. Since a great number of science, engineering and geography problems could be formulated as optimization problems, the efficient and optimization algorithms are always needed to tackle increasingly complex actual problems. Generally speaking, these algorithms can mainly be classified into two categories: traditional techniques and modern heuristic methods. Traditional approaches have been used to solve many practical problems successfully. However, the real-world problems are becoming more and more complex. The large scale optimization issues are associated with multimodality, differentiability and dimensionality. It is too difficult to search the exact optimal value by using traditional algorithms. Thus, modern heuristic techniques with simple and powerful search capabilities aroused the attention of scholars, such as ant colony optimization [1], differential evolution [2], particle swarm optimization [3], simulated annealing [4], artificial bee colony [5] and group search optimizer [6] etc. It has been proved that they are more efficient than traditional techniques in various application fields.

Recently, A. Askarzadeh have been proposed a bird mating optimizer (BMO) algorithm based on bird mating phenomenon [7]. Bird mating process is similar to an optimization process in which each bird attempts to breed a quality brood as much as possible. Using distinct patterns to move though the search space is the main difference between BMO and other intelligence algorithms. This feature helps to avoid premature convergence and maintain population diversity. Thus, BMO with simple concept and easy implementation has been successfully applied to

---

*Correspondence to: Guolin Yu (Email: guolin_yu@126.com). Research Institute of Information and System Computation Science, Beifang University of Nationalities. China (750021).

solve different complicated problems. The problem of artificial neural network (ANN) training is solved by BMO, and this method yields better results than the other classifiers [8]. BMO method is proved to be an efficient candidate for estimating the unknown parameters of fuel cell polarization curve [9]. An efficient parameter estimation process is provided by BMO, and the better performance results in gaining the coordinates of the maximum power point (MPP) more accurately [10]. Parameter identification of FC models is solved based on BMO algorithm. Simulation results demonstrate the superior performance of it [11].

BMO algorithm shows promising performance in solving optimization problems. However, it is not efficient in identifying the high performance regions of a solution space. For some complicated problems, BMO shows premature convergence or poor efficiency. Teaching-Learning-Based optimization (TLBO) with fast speed has been widely used to solve kinds of optimization problems successfully [12, 13, 14, 15, 16]. Hence, an effective hybrid algorithm with the advantages of TLBO and BMO, named TLBMO, is proposed to improve solution quality and accelerate the convergence speed. In TLBMO, the two operators of BMO and TLBO are chosen as candidates. Each individual in the current population selects one of them based on probability proportional and generates new potential solutions. In addition, a calculation formula of probability is developed empirically. Five performance criteria are adopted to compare our technique with other existing approaches.

The rest of this paper is organized as follows. Bird mating optimizer algorithm is indicated in Section 2. Section 3 shortly summarizes the original TLBO. Our proposed approach is described in detail in Section 4. Numerical simulation results are given in Section 5. Finally, conclusion is presented in Section 6.

2. Bird Mating Optimizer Algorithm

BMO is a population-based and evolutionary-based algorithm. The population is called society. Each society member represents a feasible solution, is called bird. Males and females are the main components of the society. The females with the most promising genes, contains parthenogenetic and polyandrous. The males are categorized into three unequal components, namely, monogamous, polygynous and promiscuous. In total, BMO possesses five updating patterns, they are explained below in detail.

Monogamy is a mating system that a male bird only mates with a female one. Each monogamous bird selects its interesting female from parthenogenetic and polyandrous by a probabilistic approach and mates with her. The new brood is calculated as follows:

$$
\begin{aligned}
&x_{brood} = x + \omega \times \overrightarrow{r}_. \times (x^i - x) \\
&if \quad r_1 > mcf \\
&x_{brood}(m) = lb(m) - r_2 \times (lb(m) - ub(m)), \\
&end
\end{aligned}
\tag{1}
$$

where a $1 \times D$ interesting female individual (x^i) can influence a $1 \times D$ offspring brood (x_{brood}) to some extent depending on the time-varying weight (ω), D is the dimension of problem, $\overrightarrow{r}$ denotes a $1 \times D$ vector which each element is a random number with uniform distribution between 0 and 1, mcf is the mutation control factor, distributed between 0 and 1, m is a random number varying 1 and D, r_i's are random numbers in the range [0, 1], and lb and ub are $1 \times D$ vector which each member denotes the lower and upper bounds of the each dimension of problem, respectively.

Polygamy implies a mating system that each polygynous bird chooses several female individuals as partners and mates with them. The offspring brood is calculated from the following equation:

$$
\begin{aligned}
&x_{brood} = x + \omega \times (\sum_{j=1}^{K} \overrightarrow{r_j}_. \times (x_j^i - x)) \\
&if \quad r_1 > mcf \\
&x_{brood}(m) = lb(m) - r_2 \times (lb(m) - ub(m)), \\
&end
\end{aligned}
\tag{2}
$$

where K is the number of female birds, x_j^i is the jth female bird.

Polyandry is also a mating system that several monogamous males are selected for each polyandrous bird probabilistically. The way by which a polygynous bird breeds is same as (2).

Promiscuity denotes another mating system in which one male and multiple females have unstable relationships. These birds are generated using a chaotic sequence. The way by which a promiscuous bird breeds is same as (1).

Parthenogenesis is the last mating system, means each female can raise brood without the help of males. Each female's genes are passed to her brood by making a small change in her genes probabilistically. The resultant brood is presented as follows:

$$
\begin{aligned}
&for \quad i = 1:D \\
&\quad if \quad r_1 > mcfp \\
&\quad x_{brood}(i) = x(i) + mu \times (r_2 - r_3) \times x(i), \\
&\quad end \\
&end
\end{aligned}
\tag{3}
$$

where $mcfp$ denotes the mutation control factor,mu is the step size.

The steps of BMO are provided as follow:
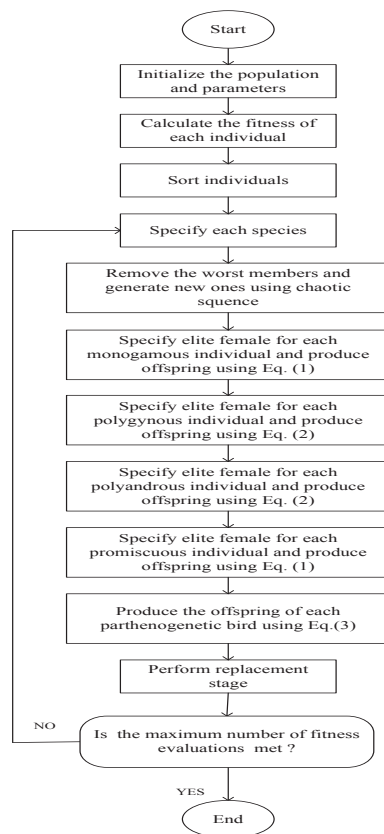
3. Teaching-learning-based optimization

TLBO is a new swarm intelligent optimization algorithm, the algorithm mimics teaching-learning phenomenon in classroom. The process of the algorithm mainly includes two parts, Teacher Phase and Learner Phase.

Teacher Phase

This phase of TLBO mimics the learning of the students through the teacher.During this phase, the teacher spares no effort to move on the average results of the entire classroom up to his or her level in terms of knowledge.x_{new} and x_i denotes $1 \times D$ vectors, D is the dimension of problem, the former are generated by the latter using the following update formula:

$$x_{new} = x_i + r \times (x_{teacher} - T_f \times x_{mean})$$
$$T_f = round(1 + rand), \tag{4}$$

where r is a random number in the range [0, 1], T_f is a teaching factor that can be either 1 or 2. $x_{teacher}$ and x_{mean} are $1 \times D$ vectors which indicates the best individual and the mean of entire population, respectively.

Learner Phase

This phase means a student increases his or her knowledge with the help of group discussions. Each student (x_i)randomly chooses a classmate $(x_j, j \neq i)$for enhancing his or her knowledge level if the classmate own more knowledge than his or her.

$$\begin{cases} x_{new} = x_i + r \times (x_j - x_i), & if \; f(x_j) < f(x_i) \\ x_{new} = x_i + r \times (x_i - x_j), & otherwise \end{cases} \tag{5}$$

4. TLBMO

4.1. Search strategy

As mentioned before, the core idea of TLBMO is that two evolution strategies, BMO operators and TLBO mechanisms, are employed to produce new individuals. That is to say, two kinds of strategies can be used for updating current population. We suppose that Pc is the probability of employing BMO strategy, the other strategy is employed with the probability of $1 - Pc$. $rand_i(0,1)$ is a random number.If $rand_i(0,1)$ is bigger than or equal to Pc, BMO operator performs. Otherwise, the TLBO strategy will be employed to generate new individual. In this sense, the abilities of exploration and exploitation are improved effectively.

4.2. Selection probability

The ultimate success of an optimization algorithm depends mostly on its ability to keep a good balance between exploitation and exploration. This ability helps algorithm to accelerate the convergence speed and improve the quality of solution. Exploitation means the concentration of the algorithm's search at the vicinity of current solutions and exploration refers to generates new solutions in as yet untested regions of search space. If a method is unable to keep balance between global and local search, it will be stagnation or get trapped in a local optima.

As far as we know that different operator produces different influence on the performance of algorithm. If one operator is emphasized in same phase, some primary information about solution space may be ignored. In the beginning, TLBO operator with powerful exploration ability is employed to expand the solution space. Thus, a larger Pc value is employed, because the higher Pc values, the higher probability of employing TLBO operator. Thereafter, BMO operator is used to search locally to get the best solution. In order to strengthen the exploitation ability, the Pc value is decreasing in the evolution process. In general, each individual has different levels of exploration and exploitation ability in solving various problems.

Table 1 The pseudo-code of TLBMO method is presented as follows:

Algorithm: TLBMO
Begin
Initialize parameters and halting criterion
Generate and evaluate the initial population x
While(the halting criterion not satisfied)
Sort birds
The promiscuous birds are removed and generated by chaotic sequence
Calculate the probability Pc
For each bird x_i of the society
If $randi(0,1) \geqslant Pc$
BMO strategy is used to the current population
Else
TLBO strategy is applied to the current population
End If
End For
Evaluate and update the offspring
End while
End
Output the obtained global best solution

The selection probability expression is empirically proposed below:

$$Pc(t) = sin(exp(-3t/T)), \qquad (6)$$

where, t and T denotes the current and the maximum number of iteration, respectively. Figure 1 introduces an example of Pc assigned for an iteration of 100.
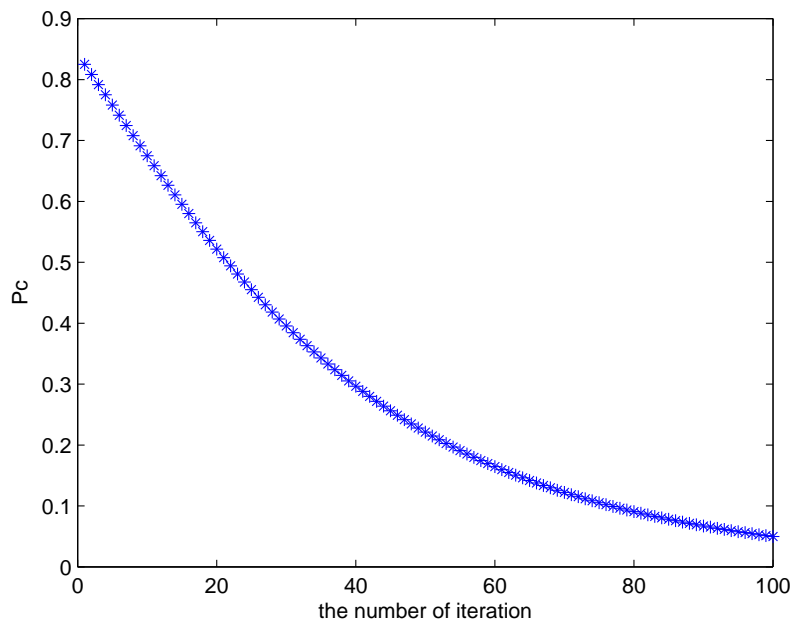

Figure.1 Probability graph of Pc assigned for an iteration of 100.

5. Experimental studies

23 benchmark functions are selected to evaluate the performance of TLBMO. A detailed description of these functions is presented in Table 2. These functions are divided into three classes [17]:
1) Unimodal functions $f_1 - f_7$,
2) Basic multimodal functions with several local minima $f_8 - f_{13}$, and
3) Low-dimensional functions with a few local minima $f_{14} - f_{23}$.

5.1. Experimental setup

In all experiments, In order to maintain a comparison as fair as possible, the maximum number of fitness evaluations($MaxFES$) is set to $D*10000$ for each algorithm [22]. Moreover, all experiments are run 50 times independently. The following parameters are used in this paper:

- The control parameters of the ABC algorithm are provided as follows [5]: population size N=20, number of onlookers=number of employed bees=0.5*population size, $limit$=number of onlookers*D.
- The parameter setting of the FEP algorithm is summarized as follows [17]: population size N=100, tournament size q=10, initial η =3.
- The parameters of GSO follow the suggestions from [6]: population size N=48, initial head angle $\varphi_0 = \frac{\pi}{4}$, the constant $\alpha = round(\sqrt{D+1})$, maximum pursuit angle $\theta_{max} = \frac{\pi}{\alpha^2}$, maximum turning angle $\alpha_{max} = \frac{\theta_{max}}{2}$, the maximum pursuit distance $l_{max} = \|\boldsymbol{U} - \boldsymbol{L}\| = \sqrt{\sum_{i=1}^{n}(\boldsymbol{U}_i - \boldsymbol{L}_i)^2}$, where $\boldsymbol{U}_i$ and $\boldsymbol{L}_i$ are the upper and lower bounds for the ith dimension.
- The parameter of TLBO only requires the population size=50 [15].
- For DE, the following parameters are employed [21]: the mutation strategies: "$DE/rand/1$", population size N=100, differential amplification factor $F = 0.5$, crossover probability constant $CR = 0.9$.
- According to [19], the parameters of PSO are set as follows: population size $N = 40$, positive constant $c1 = c2 = 1.49445$, inertia weight $w_{max} = 0.9$, $w_{min} = 0.4$.
- The same parameters are adopted by BMO [7] and TLBMO: population size $N = 200$, monogamous=100, polygynous=60, promiscuous=20, polyandrous=10, parthenogenetic=10, interesting individual=3, the number of monogamous for polygynous=10, mcf= 0.9, mu= 0.001, w_{max}= 2.5, w_{min}= 0.25, $mcfp_{min}$= 0.1, $mcfp_{max}$= 0.9.

All simulations are implemented on the same platform with Intel Core i5-3210M CPU 2.50GHz, 4G of RAM, and windows7 with MATLAB R2010a.

5.2. Performance criteria

This paper employs five performance criteria to evaluate the performance of TLBMO. These criteria are introduced as follows.

- *Accuracy* [22]: The mathematical expression is

$$Accuracy = |F(\hat{x}) - F(x^*)| \tag{7}$$

  where $\hat{x}$ is the best solution found by the algorithm in a run,x^* is the global optimal solution. The minimum accuracy is recorded when reaching the maximum number of function evaluation($MaxFES$). The average and standard deviation of the accuracy values are also calculated.
- *TestFES* [18]: The criteria records the number of fitness evaluations when the value-to-reach (VTR) is achieved.

Table 2 the twenty-three benchmark functions used in our study.

Test function	D	S	Minimum
$f_1(x) = \sum\limits_{i=1}^{D} x_i^2$	30	$[-100, 100]^D$	0
$f_2(x) = \sum\limits_{i=1}^{D} |x_i| + \prod\limits_{i=1}^{D} |x_i|$	30	$[-10, 10]^D$	0
$f_3(x) = \sum\limits_{i=1}^{D} (\sum\limits_{j=1}^{i} x_j)^2$	30	$[-100, 100]^D$	0
$f_4(x) = \max\limits_{1 \le i \le D} \{|x_i|\}$	30	$[-100, 100]^D$	0
$f_5(x) = \sum\limits_{i=1}^{D-1} [100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2]$	30	$[-30, 30]^D$	0
$f_6(x) = \sum\limits_{i=1}^{D} (\lfloor x_i + 0.5 \rfloor)^2$	30	$[-100, 100]^D$	0
$f_7(x) = \sum\limits_{i=1}^{D} i x_i^4 + random[0, 1)$	30	$[-1.28, 1.28]^D$	0
$f_8(x) = \sum\limits_{i=1}^{D} -x_i \sin(\sqrt{|x_i|})$	30	$[-500, 500]^D$	-12569.5
$f_9(x) = \sum\limits_{i=1}^{D} [x_i^2 - 10\cos(2\pi x_i) + 10]$	30	$[-5.12, 5.12]^D$	0
$f_{10}(x) = -20\exp(-0.2\sqrt{\frac{1}{D}\sum\limits_{i=1}^{D} x_i^2}) - \exp(\frac{1}{D}\sum\limits_{i=1}^{D}\cos 2\pi x_i) + 20 + e$	30	$[-32, 32]^D$	0
$f_{11}(x) = \frac{1}{4000}\sum\limits_{i=1}^{D} x_i^2 - \prod\limits_{i=1}^{D}\cos(\frac{x_i}{\sqrt{i}}) + 1$	30	$[-600, 600]^D$	0
$f_{12}(x) = \frac{\pi}{D}\{10\sin^2(\pi y_i) + \sum\limits_{i=1}^{D-1}(y_i - 1)^2[1 + 10\sin^2(\pi y_{i+1})] + (y_i - 1)^2\}$ $+ \sum\limits_{i=1}^{D}\mu(x_i, 10, 100, 4)$	30	$[-50, 50]^D$	0
$f_{13}(x) = 0.1\{\sin^2(3\pi x_1) + \sum\limits_{i=1}^{D-1}(x_i - 1)^2[1 + \sin^2(3\pi x_{i+1})]$ $+(x_n - 1)[1 + \sin^2(2\pi x_n)]\} + \sum\limits_{i=1}^{D}\mu(x_i, 5, 100, 4)$	30	$[-50, 50]^D$	0
$f_{14}(x) = [\frac{1}{500} + \sum_{j=1}^{25}\frac{1}{j + \sum\limits_{i=1}^{2}(x_i - a_{ij})^6}]^{-1}$	2	$[-65.536, 65.536]^D$	1
$f_{15}(x) = \sum\limits_{i=1}^{11}[a_i - \frac{x_1(b_i^2 + b_i x_2)}{b_i^2 + b_i x_3 + x_4}]^2$	4	$[-5, 5]^D$	0.0003075
$f_{16}(x) = 4x_1^2 - 2.1x_1^4 + \frac{1}{3}x_1^6 + x_1 x_2 - 4x_2^2 + 4x_2^4$	2	$[-5, 5]^D$	-1.0316285
$f_{17}(x) = (x_2 - \frac{5.1}{4\pi^2}x_1^2 + \frac{5}{\pi} - 6)^2 + 10(1 - \frac{1}{8\pi}\cos x_i + 10)$	2	$[-5, 10] \times [0, 15]$	0.398
$f_{18}(x) = [1 + (x_1 + x_2 + 1)^2(19 - 14x_1 + 3x_1^2 - 14x_2 + 6x_1 x_2 + 3x_2^2)] \times$ $[30 + (2x_1 - 3x_2)^2(18 - 32x_1 + 12x_1^2 + 48x_2 - 36x_1 x_2 + 27x_2^2)]$	2	$[-2, 2]^D$	3
$f_{19}(x) = -\sum\limits_{i=1}^{4} c_i \exp[-\sum\limits_{j=1}^{4} a_{ij}(x_j - p_{ij})^2]$	3	$[0, 1]^D$	-3.86
$f_{20}(x) = -\sum\limits_{i=1}^{6} c_i \exp[-\sum\limits_{j=1}^{6} a_{ij}(x_j - p_{ij})^2]$	6	$[0, 1]^D$	-3.32
$f_{21}(x) = -\sum\limits_{i=1}^{5}[(x - a_i)(x - a_i)^T + c_i]^{-1}$	4	$[0, 10]^D$	-10
$f_{22}(x) = -\sum\limits_{i=1}^{7}[(x - a_i)(x - a_i)^T + c_i]^{-1}$	4	$[0, 10]^D$	-10
$f_{23}(x) = -\sum\limits_{i=1}^{10}[(x - a_i)(x - a_i)^T + c_i]^{-1}$	4	$[0, 10]^D$	-10

• *Acceleration rate* (AR) [18]: This criterion is applied to evaluate a methods convergence speed quantitatively.

$$AR = \frac{TestFES_{competitor}}{TestFES_{Method}}, \tag{8}$$

where $AR < 1$ denotes the method is slower than its competitor.
- *Convergence Graphs* [20]: The convergence graphs demonstrate that the mean accuracy performance of total runs, respectively. It is used to measure the convergence speed qualitatively.
- *Success rate(SR)* [20]: The SR is calculated as the number of successful runs to the total number of trials.

$$SR = \frac{\# \ of \ successful \ runs}{total \ runs}. \tag{9}$$

5.3. *Performance of TLBMO*

In order to present superiority of the new hybrid algorithm, TLBMO is compared with seven existing algorithms. The mean, standard deviation values and t-test for all benchmark functions are counted and listed in Table 3. The values of AR and SR are provided in Table 4. In addition, some representative convergence graphs are presented in Figure 2. The performance of TLBMO is analyzed from the following four aspects.
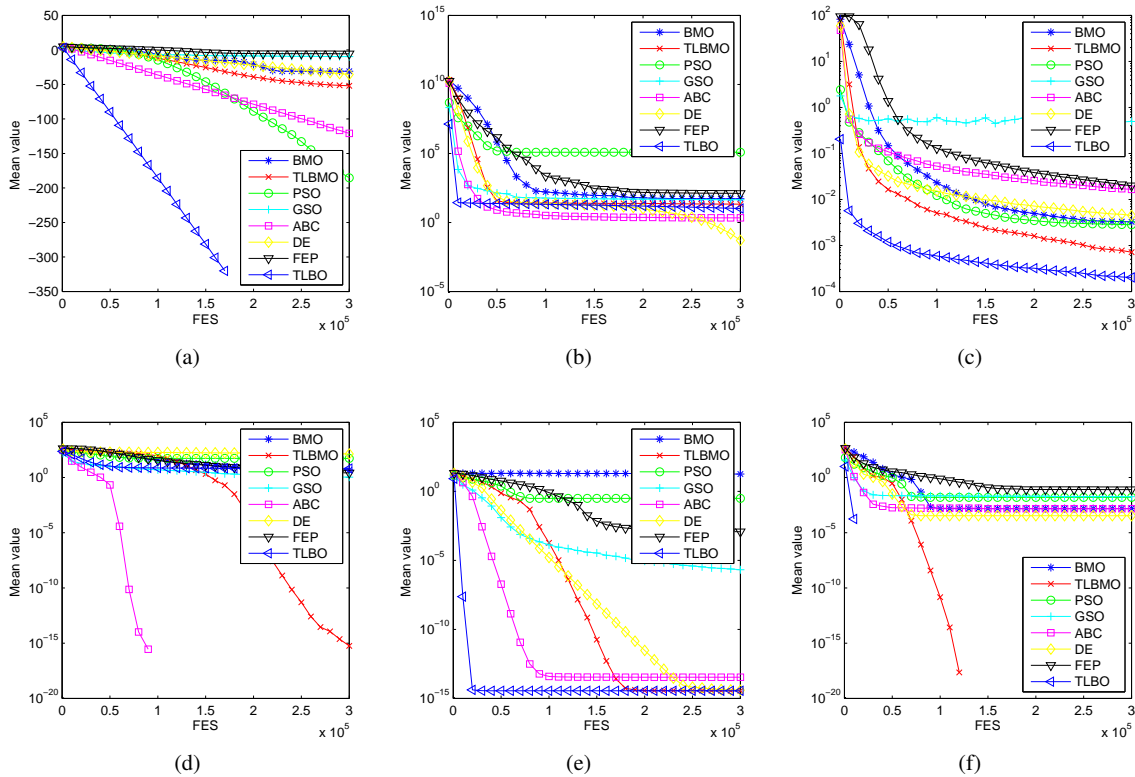
Figure.2 Convergence graph of overall algorithms on the selected functions, (a) f_1, (b) f_5, (c) f_7, (d) f_9, (e) f_{10}, (f) f_{11}.

From Table 3-4 and Figure 2, the following results and analysis can be obtained.

- *Accuracy analysis*: For unimodal functions, from Table 3, it is obvious that TLBMO is able to perform consistently better results than BMO on f_1, f_2, f_3, f_4, f_7, but there are weaker than TLBO. For f_5, there is no significant difference between TLBMO and TLBO, but it is superior to BMO. TLBMO, TLBO and BMO have the same final results on f_6.

  For multimodal functions with many local minima, from table 3 and figure 2, it is clear that TLBMO is significantly better than BMO and TLBO on function f_9, f_{12}. For f_8, f_{13}, TLBMO's final results are as good as BMO's and TLBO's. TLBMO and TLBO can find the global optima on f_{11}.

Table 3 Comparison of TLBMO with different algorithms.

Function	Index	TLBMO	BMO	PSO	GSO	ABC	TLBO	DE	FEP
f_1	mean	$1.00e-052$	$1.11e-031$	$5.10e-186$	$2.13e-010$	$1.2e-121$	**0**	$2.29e-036$	$5.30e-008$
	std	$2.02e-052$	$7.73e-031$	0	$5.82e-010$	$3.08e-121$	0	$3.31e-036$	$3.75e-007$
	h		1	1	1	1	1	1	1
f_2	mean	$8.25e-026$	$2.00e-017$	$1.54e+002$	$5.30e-005$	$6.41e-061$	**$4.12e-278$**	$5.98e-014$	$1.24e-004$
	std	$8.51e-026$	$9.31e-017$	$1.31e+002$	$1.50e-004$	$8.88e-061$	0	$3.27e-013$	$8.74e-004$
	h		1	1	1	1	1	1	1
f_3	mean	$9.86e-021$	$2.28e+001$	$1.70e+003$	$2.97e+000$	$2.61e+003$	**$3.80e-091$**	$1.52e-005$	$8.40e-001$
	std	$1.82e-020$	$1.81e+001$	$3.49e+003$	$1.90e+000$	$5.63e+002$	$2.66e-090$	$1.42e-005$	$5.94e+000$
	h		1	1	1	1	1	1	1
f_4	mean	$6.39e-019$	$3.96e-001$	$1.04e-007$	$1.21e-002$	$9.81e-002$	**$2.32e-222$**	$3.03e-001$	$3.36e-002$
	std	$1.05e-054$	$3.29e-001$	$1.61e-007$	$4.15e-003$	$3.40e-002$	0	$6.25e-001$	$2.38e-001$
	h		1	1	1	1	1	1	1
f_5	mean	$2.13e+001$	$5.24e+001$	$1.21e+005$	$4.42e+001$	$2.14e+000$	$1.02e+001$	**$4.85e-002$**	$9.21e-001$
	std	$1.02e+000$	$4.14e+001$	$3.28e+005$	$3.68e+001$	$3.34e+000$	$1.95e+000$	$7.11e-002$	$6.51e+000$
	h		1	1	1	1	1	1	1
f_6	mean	**0**	**0**	$2.00e+002$	**0**	**0**	**0**	**0**	**0**
	std	0	0	$1.41e+003$	0	0	0	0	0
	h		0	0	0	0	0	0	0
f_7	mean	$7.14e-004$	$3.26e-003$	$2.81e-003$	$3.80e-002$	$1.65e-002$	**$2.02e-004$**	$4.47e-003$	$2.80e-004$
	std	$3.02e-004$	$1.57e-003$	$1.27e-003$	$3.04e-002$	$3.99e-003$	$6.92e-005$	$1.16e-003$	$1.98e-003$
	h		1	1	1	1	1	1	1
	b/s/w		6/1/0	6/0/1	6/1/0	3/1/3	0/1/6	5/1/1	4/1/2
f_8	mean	$3.51e+003$	$2.19e+003$	$2.85e+003$	$1.35e-002$	**$1.34e-002$**	$2.93e+003$	$3.98e+003$	$1.24e+004$
	std	$7.51e+002$	$3.97e+002$	$6.90e+002$	$5.34e-004$	$7.35e-012$	$6.48e+002$	$9.51e+002$	$1.57e+003$
	h		1	1	1	1	1	1	1
f_9	mean	$5.68e-016$	$4.82e+000$	$5.40e+001$	$1.07e+000$	**0**	$6.45e+000$	$1.38e+002$	$5.97e-002$
	std	$4.02e-015$	$2.68e+000$	$2.16e+001$	$8.96e-001$	0	$5.50e+000$	$2.52e-001$	$4.22e-001$
	h		1	1	1	0	1	1	0
f_{10}	mean	**$3.55e-015$**	$1.78e-001$	$3.08e-001$	$2.10e-006$	$3.42e-014$	**$3.55e-015$**	$3.91e-015$	$2.27e-005$
	std	0	$4.29e+000$	$1.99e+000$	$1.92e-006$	$4.29e-015$	0	$1.08e-015$	$1.61e-004$
	h		1	1	1	1	0	1	1
f_{11}	mean	**0**	$1.53e-003$	$1.59e-002$	$1.83e-002$	$1.48e-003$	**0**	$3.45e-004$	$1.56e-003$
	std	0	$3.91e-003$	$2.22e-002$	$1.65e-002$	$4.51e-003$	0	$2.44e-003$	$1.10e-002$
	h		1	1	1	1	0	0	0
f_{12}	mean	$7.39e-019$	$3.82e-014$	$4.15e-003$	$1.26e-013$	**$1.57e-032$**	$2.07e-003$	**$1.57e-032$**	$4.65e-010$
	std	$5.21e-018$	$2.70e-013$	$2.05e-002$	$2.02e-013$	$5.53e-048$	$1.47e-002$	$5.53e-048$	$3.29e-009$
	h		1	1	1	1	0	0	1
f_{13}	mean	$2.20e-003$	$1.76e-003$	$1.32e-003$	$8.51e-012$	**$1.35e-032$**	$2.96e-002$	**$1.35e-032$**	$6.78e-009$
	std	$4.44e-003$	$4.06e-003$	$3.61e-003$	$4.13e-011$	$1.11e-047$	$4.22e-002$	$1.11e-047$	$4.79e-008$
	h		1	1	1	1	1	1	1
	b/s/w		4/0/2	4/0/2	5/0/1	2/0/4	3/2/1	5/0/1	5/0/1
f_{14}	mean	$3.84e-006$	$3.84e-006$	$3.84e-006$	$3.84e-006$	$3.84e-006$	$3.84e-006$	$3.84e-006$	**$7.68e-008$**
	std	$7.26e-012$	0	$1.19e-016$	$2.08e-016$	$8.62e-017$	$1.03e-013$	1	$5.43e-007$
	h		1	1	1	1	1	1	1
f_{15}	mean	$1.24e-003$	$1.56e-003$	$1.63e-003$	$1.50e-003$	$1.13e-003$	$9.67e-004$	$1.13e-003$	**$3.78e-005$**
	std	$6.59e-004$	$6.07e-004$	$2.26e-003$	$2.26e-003$	$3.48e-004$	$6.55e-004$	$6.25e-004$	$2.67e-004$
	h		0	0	0	0	1	1	1
f_{16}	mean	**$4.65e-008$**	**$4.65e-008$**	**$4.65e-008$**	**$4.65e-008$**	**$4.65e-008$**	**$4.65e-008$**	**$4.65e-008$**	$4.83e-005$
	std	$1.23e-014$	0	$5.33e-017$	$6.73e-017$	$8.22e-017$	$1.36e-016$	$4.40e-017$	$3.41e-004$
	h		1	1	1	1	1	1	1
f_{17}	mean	**$3.58e-007$**	**$3.58e-007$**	**$3.58e-007$**	**$3.58e-007$**	**$3.58e-007$**	**$3.58e-007$**	**$3.58e-007$**	$8.46e-005$
	std	$3.16e-012$	0	0	$7.41e-013$	0	$9.16e-015$	0	$5.99e-004$
	h		1	1	1	1	1	1	1
f_{18}	mean	$-6.90e-014$	$-7.78e-014$	$-7.75e-014$	$-7.37e-014$	$6.40e-006$	$-7.77e-014$	**$-7.89e-014$**	$8.96e-012$
	std	$6.27e-015$	$7.86e-016$	$1.23e-015$	$3.56e-015$	$3.94e-005$	$6.96e-016$	$9.26e-016$	$6.34e-011$
	h		1	1	1	0	1	1	1
f_{19}	mean	**$-1.48e-007$**	**$-1.48e-007$**	**$-1.48e-007$**	**$-1.48e-007$**	$-1.42e-007$	**$-1.48e-007$**	**$-1.48e-007$**	$1.09e-002$
	std	$2.24e-016$	0	$8.79e-017$	$2.89e-016$	$2.76e-008$	0	0	$7.73e-002$
	h		1	1	1	0	1	1	0
f_{20}	mean	$8.17e-002$	$9.78e-002$	$4.23e-002$	$4.29e-002$	**$1.99e-006$**	$1.08e-002$	$1.03e-001$	$1.69e-002$
	std	$5.55e-002$	$4.63e-002$	$6.40e-002$	$5.78e-002$	$1.72e-016$	$3.27e-002$	$4.18e-002$	$1.20e-001$
	h		0	1	1	1	1	1	1
f_{21}	mean	$2.04e-001$	$1.21e+000$	$4.69e+000$	$2.02e+000$	$1.10e-005$	$9.39e-002$	**$3.21e-007$**	$1.02e-001$
	std	$1.01e+000$	$2.32e+000$	$2.09e+000$	$2.92e+000$	$4.59e-005$	$6.55e-001$	0	$7.21e-001$
	h		1	1	1	1	0	1	1
f_{22}	mean	**$-4.06e-005$**	$3.60e-001$	$3.20e+000$	$2.94e+000$	$3.63e-004$	$1.72e-001$	**$-4.06e-005$**	$1.06e-001$
	std	$1.50e-013$	$1.48e+000$	$3.09e+000$	$3.41e+000$	$1.74e-003$	$8.11e-001$	$8.61e-016$	$7.52e-001$
	h		1	1	0	1	1	1	1
f_{23}	mean	$1.34e-001$	$1.53e-001$	$2.43e+000$	$2.38e+000$	$5.98e-005$	$1.10e-001$	**$-9.82e-006$**	$1.08e-001$
	std	$9.48e-001$	$1.08e+000$	$3.22e+000$	$3.26e+000$	$3.55e-004$	$7.77e-001$	$2.51e-016$	$7.65e-001$
	h		1	1	1	1	1	1	1
	b/s/w		5/4/1	4/4/2	4/4/2	3/3/4	2/4/4	0/5/5	5/0/5
	$\sum b/s/w$		15/5/3	14/4/5	15/5/3	8/4/11	5/7/11	10/6/7	14/1/8

[a] $b/s/w$ indicates that TLBMO wins in b functions, tie in s functions and loses in w functions, compared with its competitors.

For multimodal functions with a few local minima, as indicated in table 3, there are no obvious difference between TLBMO and its competitors.

- *Significance analysis*: In order to explain the statistical significance of TLBMO, the Wilcoxon signed-rank test is employed. The numerical values "1" means that TLBMO is statistically inferior to other algorithms, "0" represents that TLBMO is equal to its competitor. The total number of TLBMO performs better, similar or worse than other algorithms are summarized in the last row of Table 3. The results demonstrate that there is significantly difference between TLBMO and BMO on the majority of test functions. The mean values of the final solutions reconfirm that the difference between them is not obvious.

Table 4 Comparison of TLBMO and its competitor.

Function	Index	TLBMO	BMO	PSO	GSO	ABC	TLBO	DE	FEP
f_1	AR		1.29	1.02	1.59	0.39	0.08	1.14	0
	SR	1	1	1	1	1	1	1	0.98
f_2	AR		1.03	0.19	0	0.44	0.10	1.56	0
	SR	1	1	0.22	0	1	1	1	0.98
f_3	AR		0	0.90	0	0	0.23	0	0
	SR	1	0	0.76	0	0	1	0	0.98
f_4	AR		0	0.44	0	0	0.09	0	0
	SR	1	0	0.24	0	0	1	0	0.98
f_5	AR		NaN	NaN	NaN	NaN	NaN	NaN	NaN
	SR	0	0	0	0	0	0	0	0.98
f_6	AR		2.16	1.51	0.80	0.37	0.06	0.87	0.06
	SR	1	1	0.98	1	1	1	1	1
f_7	AR		0	0.03	0	0	0.28	0	0
	SR	0.86	0	0.02	0	0	1	0.02	0.98
f_8	AR		1	0.21	0.25	0.11	0.26	0.51	0.01
	SR	1	1	1	1	1	1	1	1
f_9	AR		0.03	0	0.20	0.35	0.14	0	0
	SR	1	0.02	0	0.14	1	0.24	0	0.98
f_{10}	AR		0.03	0.82	0	0.47	0.09	1.25	0
	SR	1	0.02	0.96	0	1	1	1	0.98
f_{11}	AR		1.05	0.33	0.32	0.46	0.09	1.09	0
	SR	1	0.84	0.34	0.02	0.88	1	0.98	0.98
f_{12}	AR		1.00	0.74	0.63	0.28	0.37	0.82	0
	SR	1	1	0.96	1	1	0.98	1	0.98
f_{13}	AR		1.05	0.81	1.05	0.35	0.17	1.01	0
	SR	0.72	0.84	0.88	1	1	0.28	1	0.98
f_{14}	AR		0.97	0.83	0.50	0.48	1.42	1.27	0.04
	SR	1	1	1	1	1	1	1	1
f_{15}	AR		0.48	0.40	0.64	0.88	0.56	0.24	0
	SR	0.40	0.24	0.38	0.62	0.44	0.56	0.24	0.98
f_{16}	AR		0.96	0.89	0.21	0.35	0.63	0.86	0
	SR	1	1	1	1	1	1	1	0.98
f_{17}	AR		0.82	0.75	0.32	0.45	0.70	0.97	0
	SR	1	1	1	1	1	1	1	0.98
f_{18}	AR		0.73	0.95	0.28	1.25	0.60	0.58	0.05
	SR	1	1	1	1	0.92	1	1	1
f_{19}	AR		0.92	0.82	0.40	0.72	0.70	0.82	0
	SR	1	1	1	1	1	1	1	0.98
f_{20}	AR		0.25	0.58	0.22	0.43	1.11	0.18	0
	SR	0.30	0.18	0.68	0.64	1	0.84	0.14	0.98
f_{21}	AR		0.62	0.11	0.40	0.46	1.09	0.51	0
	SR	0.96	0.76	0.14	0.66	1	0.96	1	0.98
f_{22}	AR		0.74	0.41	0.23	0.58	1.13	0.63	0
	SR	1	0.94	0.46	0.56	0.94	0.94	1	0.98
f_{23}	AR		0.81	0.61	0.61	0.71	1.07	0.68	0
	SR	0.98	0.98	0.62	0.64	0.96	0.98	1	0.98
	$\sum SR$	20.20	14.80	13.70	13.50	18.10	19.80	16.40	22.60

- *Convergence speed*: Table 4 show that TLBMO converges faster than BMO, especially for function f_6. For example, compared with BMO, the AR value is 1.05, which means that TLBMO is on average 5% faster

than BMO for f_{11} function. Additionally, Figure 2 shows that TLBMO is able to converge to better solutions quickly and it has higher accuracy than BMO on the most of functions. From figure 2. it is clear that TLBO is significantly superior to all the other methods including TLBMO in the process of evolution, while TLBMO performs the second best on this benchmark function. Looking carefully at Figure 2 (c), PSO shows a faster convergence rate initially than TLBMO, however, it is outperformed by TLBMO after 25000 FES. we can observe from Figure 2 (d) that all the other algorithms show the almost same startpoint, and ABC has a faster convergence speed than other methods. Although slower, TLBMO eventually finds the global solution close to ABC. Totally, the convergence speed of BMO is improved in a certain extent.

- $Success\ ratio$: Table 4 reveals that the TLBMO has more superior searching ability than its competitors, and provides the much higher successful rate $\sum SR = 20.20$ than other existing algorithms. To be specific, our algorithm can effectively solve $(SR = 1)$ seventeen out of twenty-three test functions. It is worth mentioning that TLBMO can yield the smallest mean value with $SR = 1$ on function f_{11} and f_{22}. In general, the hybrid method is able to form stable search mechanism.

6. Conclusion

In order to accelerate the convergence speed and improve solution quality of BMO, a hybrid algorithm, named TLBMO, is proposed. This method can provide a better balance between exploration and exploitation in the evolving process. The performance of TLBMO algorithm is verified on a comprehensive set of well-known 23 benchmark functions. Experimental results demonstrate the good performance of TLBMO.

Acknowledgement

REFERENCES

1. M. Dorigo, and T. Stutzle, *Ant colony optimization.*, MAMIT Press. Cambridge, 2004.
2. R. Storn, and K. Price, *Differential evolution-a simple and efficient heuristic for global optimization over continuous spaces*, Joural of Global optimization, vol. 11, pp. 341-359, 1997.
3. J. Kennedy, and R. Eberhart, *Particle swarm optimization*, IEEE International conference on Neural Networks, pp. 1942-1948, 1995.
4. S. Kirkpatrick, C. D. Gelatt and M. P. Vecchi, *Optimization by simulated annealing*, Science, vol. 220, pp. 671-680, 1983.
5. D. Karaboga, *An idea based on honey bee swarm for numerical optimization*, Technical report-tr06, Kayseri, Turkey: Erciyes University, 2005.
6. S. He, Q. H. Wu and J. R. Saunders, *Group search optimizer: an optimization algorithm inspired by animal searching behavior*, IEEE Transactions on Evolutionary Computation, vol. 13, pp. 973-990, 2009.
7. A. Askarzadeh, *An optimization algorithm inspired by bird mating strategies*, Communication in Nonlinear Science and Numerical Simulation, vol. 19, pp. 1213-1228, 2014.
8. A. Askarzadeh, and A. Rezazadeh, *Artificial neural network training using a new efficient optimization algorithm*, Applied Soft Computing, vol. 13, pp. 1206-1213, 2013.
9. A. Askarzadeh, *Parameter estimation of fuel cell polarization curve using BMO algorithm*, International Journal of Hydrogen Energy, vol. 38, pp. 15405-15413, 2013.
10. A. Askarzadeh, and A. Rezazadeh, *Extraction of maximum power point in solar cells using bird mating optimizer-based parameters identification approach*, Solar Energy, vol. 90, pp. 123-133, 2013.
11. A. Askarzadeh, and A. Rezazadeh, *A new heuristic optimization algorithm for modeling of proton exchange membrane fuel cell: bird mating optimizer*, International Journal of Energy Research, vol.37, pp. 1196-1204, 2013.
12. R. V. Rao, and V. Patel, *Multi-objective optimization of two stage thermoelectric cooler using a modified teaching-learning-based optimization algorithm*, Engineering Applications of Artificial Intelligence, vol. 26, pp. 430-445, 2013.

13. R. V. Rao, and V. Patel, *An improved teaching-learning-based optimization algorithm for solving unconstrained optimization problems*, Scientia Iranica, vol. 20, pp. 710-720, 2013.
14. R. V. Rao, and V. Patel, *D. P. Vakharia. Teaching-Learning-Based Optimization: An optimization method for continuous non-linear large scale problems*, Information Sciences, vol. 183, pp. 1-15, 2012.
15. R. V. Rao, and V. Patel, *D. P. Vakharia. Teaching-learning-based optimization: An novel method for constrained mechanical design optimization problems*, Computer-Aided Design, vol. 43, pp. 303-315, 2011.
16. R. V. Rao, and V. D. Kalyankar, *Parameter optimization of modern machining process using teaching-learning-based optimization algorithm*, Engineering Applications of Artificial Intelligence, vol. 26, pp. 524-531, 2013.
17. X. Yao, Y. Liu, and G. M. Linm, *Evolutionary Programming Made Faster*, IEEE Transactions on Evolutionary Computation, vol. 3, pp. 82-102, 1999.
18. W. Y. Gong, Z. H. Cai, and C. X. Ling, *DE/BBO: A hybrid differential evolution with biogeography-based optimization for global numerical optimization*, Available from: http://embeddedlab.csuohio.edu/BBO/.
19. J. J. Liang, A. K. Qin, P. N. Suganthan, and S. Baskar, *Comprehensive learning particle swarm optimizer for global optimization of multimodal functions*, IEEE Transactions on Evolutionary Computation, vol. 10, no. 3, pp. 281-295, 2006.
20. W. Y. Gong, Z. H. Cai, C. X. Ling, and H. Li, *Enhanced differential evolution with adaptive strategies for numerical optimization*, IEEE Transactions on Evolutionary Computation, vol. 41, pp. 397-413, 2011.
21. S. Rahnamayan, H. R.Tizhoosh, and M. A. Salama, *Opposition-based differential evolution*, IEEE Transactions on Evolutionary Computation, vol. 12, pp. 64-79, 2008.
22. P. N. Suganthan, N. Hansen, and J. J. Liang, et al, *Problem definitions and evaluation criteria for CEC 2005 special session on real-parameter optimization*, http://www.ntu.edu.sg/home/epnsugan.