A Novel Biogeography-Based Optimization Algorithm with Momentum Migration and Taxonomic Mutation

Xinchao Zhao[1(✉)], Yisheng Ji[1], and Junling Hao[2]

[1] School of Science, Beijing University of Post and Telecommunications, Beijing 100876, China
zhaoxc@bupt.edu.cn
[2] School of Statistics, University of International Business and Economics, Beijing 10029, China

Abstract. Biogeography-based optimization (BBO) algorithm is not good at dealing with regions where function values change dramatically or barely. A novel biogeography-based optimization algorithm is proposed in this paper based on Momentum migration and taxonomic mutation. The momentum item is added to the original migration operation of BBO. It makes the algorithm more advantageous in dealing with regions where function values change dramatically or barely. At the same time, taxonomic mutation strategy divides the solutions into three categories: promising class, middle class and inferior class. Promising solutions do not take part in this mutation operation. Solutions of middle class use balanced differential mutation, and inferior solutions adopt exploration-biased random mutation. This strategy further increases the diversity of population. The simulation experiments are carried out with different types of CEC2014 benchmark functions. The proposed algorithm is compared with other algorithms and shows stronger global search ability, faster convergence speed and higher convergence accuracy.

Keywords: Biogeography-based optimization · BBO · Momentum · Differential mutation · Random mutation

1 Introduction

Biogeography-based optimization (BBO) algorithm was proposed by Dan Simon in 2008 [1], which is used to study the geographic distribution mathematical algorithm of organisms in the optimization problem.

BBO algorithm attracts widely attention from theory, development and application. Ma and Simon [2] proposed blended biogeography-based optimization for constrained optimization. Cai et al. [3] introduced a biogeography-based algorithm based on evolutionary programming. Ma [4] found that cosine migration curve provides the best performance among six different models. Bhattacharya et al. [5] proposed biogeography-based optimization for different economic Load dispatch problems. Some more BBO variants are also appeared [6–9]. Bhattacharya and Chattopadhyay [11] applied biogeography-based optimization to solve different optimal power flow problems. BBO algorithm

is also used to classification problems [12] and task scheduling scheme [13, 14] and Estimation of landslide susceptibility [10].

However, BBO algorithm still doesn't work very well for some functions whose values change drastically or barely. Later stage of BBO algorithm is prone to lose population diversity. Aiming at these problems, this paper proposes a novel BBO algorithm based on momentum migration and taxonomic mutation.

The remainder of the paper is organized as follows. Section 2 reviews BBO algorithm. Section 3 describes the proposed algorithm MTBBO with momentum migration and taxonomic mutation. Simulation results and analysis are elaborated in Sect. 4. The work is concluded in Sect. 5.

2   Biogeography-Based Optimization (BBO) Algorithm

BBO is a population-based optimization algorithm, which sets each solution as a habitat, fitness of solution as Habitat Suitability Index (*HSI*), and component of each solution as Suitable Index Variable (*SIV*). There are many factors that affect SIVs in natural solutions, such as rainfall, land area, plant diversity, address diversity, and climate. A good habitat has a high HSI, while a poor habitat has a low *HSI*. The algorithm simulates the searching process of species searching for the best adaptive islands. BBO mainly consists of migration operation and mutation operation. The main operations of BBO algorithm are as follows.

2.1   Migration Operation

Each habitat H_i has corresponding immigration rate λ_i and emigration rate μ_i. Both two parameters are closely related to *HSI*. In general, high HSI habitat has more species, and species tend to saturation. For high *HSI* habitat, there will be a high trend of outward migration. At this time, the emigration rate is high and the immigration rate is low due to the pressure of species competition. For low HSI habitat population, it tends to have low emigration rate and high immigration rate.

Assuming habitat H_i currently accommodates S_i species. S_{max} is maximum number of species. λ_i and μ_i are immigration rate and emigration rate of H_i, respectively, which are adjusted as Eq. (1).

$$\begin{cases} \lambda_i = I \times \left(1 - \frac{S_i}{S_{max}}\right) \\ \mu_i = E \times \frac{S_i}{S_{max}} \end{cases} \tag{1}$$

Where *I* is the maximum immigration rate and *E* is maximum emigration rate. Migration operation is described as Algorithm 1.

Algorithm 1: Migration Operation

1: For i=1 to N do
2:      For d=1 to D do
3:          If habitat H_i is selected with λ_i
4:              If habitat H_j is selected with μ_j
5:                  $H_i(d) \leftarrow H_j(d)$;
6:              End if
7:          End if
8:      End for
9: End for

2.2 Mutation Operation

Some unexpected events cause some properties of the habitat change, such as HSI and the number of species. Mutation rate is determined by species probability. According to biogeography, when number of species in habitat is too large or too small, species probability is low. When the number of species in habitat is moderate, species probability is high. Equation (2) gives the relationship of Mutation rate m_i and species probability P_i. Species probability P_i is decided by number of species S_i [1] (In this paper, $S_{max} = N$ and $S_i = S_{max} - i$).

$$m_i = m_{max} \times \left(1 - \frac{P_i}{P_{max}}\right) \qquad (2)$$

where m_{max} is the maximum mutation rate and P_{max} is maximum species probability. Mutation operation is described as Algorithm 2.

Algorithm 2: Mutation Operation

1: For i=1 to N do
2：   For d=1 to D do
3：   If $H_i(d)$ is selected with probability m_i;
4：   Replace $H_i(d)$ with a randomly generated SIV;
5：   End if
6:     End for
7: End for

---

3   Proposed Algorithm: MTBBO

It is known that randomness of migration operation may make probability of migrating to better solution is not very large. The simple mutation operation also leads to the fact that mutation is not so ideal. In view of these problems, momentum idea in migration operation and taxonomic mutation idea in mutation operation are introduced in this paper. Then a novel biogeography-based optimization algorithm with momentum migration and taxonomic mutation is proposed.

3.1 Momentum Migration

3.1.1 Cosine Migration Model

The relationship between migration rate and the number of species in the original BBO is linear as Eq. (1). But this is not in accordance with the situation in nature, and cosine function is more in line with actual relationship in nature. When number of habitat species is large or small, the change of emigration rate and immigration rate is a little slow. When number of species in habitat is moderate, the change of emigration rate and immigration rate is a little fast. Some migration models are discussed in the literature [4], and they concluded that the cosine model performs best. So it is a better way to choose cosine migration model, which is indicated as Eq. (3).

$$\begin{cases} \lambda_i = \frac{I}{2} \times \left[\cos\left(\pi \times \frac{S_i}{S_{max}} \right) + 1 \right] \\ \mu_i = \frac{E}{2} \times \left[-\cos\left(\pi \times \frac{S_i}{S_{max}} \right) + 1 \right] \end{cases} \tag{3}$$

Where I is the maximum immigration rate and E is maximum emigration rate.

3.1.2 Momentum Migration Operation

The migration operation in original BBO is not so ideal. So the concept of momentum item of deep learning is introduced to make migration more ideal [15]. First of all, the concept of "gradient" is extended so that it can be applied to swarm intelligence algorithms. For the migration result of H_i is H_i', the "gradient" of H_i is defined as Eq. (4).

$$\widetilde{\nabla} f(H_i) = \frac{H_i' - H_i}{\left\| H_i' - H_i \right\|_2} \tag{4}$$

Momentum migration operator is described as Algorithm 3.

Algorithm 3: Momentum Migration Operation

1: Initialization: momentum coefficient α, learning rate η, step v=0.
2: While FE≤maxFE do
3:       Follow algorithm 1 and get the pre-migration result H_i' for each solution H_i.
4:       Calculate the gradient $\widetilde{\nabla} f(H_i)$ for each H_i, and select 10 gradients randomly to calculate their average gradients g_t;
5:       Step update: v=α*v+η*g_t;
6:       Solution update: H_i=H_i+v;
7: End while

3.2 Taxonomic Mutation

Taxonomic mutation operation is adopted for individual dependent mutation scale in this paper. Solutions are divided into three classes: optimal class, middle class and poor class. Solutions of promising class are not executed mutation operation. Mutation is only for middle and inferior class, but mutation operations of middle class and poor

class are different. For the solutions of middle class, their own information is somewhat beneficial. So these solutions are modified with differential mutation. The crossover rate P_C is introduced to judge whether it is cross or not. For the solutions of inferior class, they are normally far away from global optimal solution, so mutations are implemented on them which are independent of their own information.

3.2.1 Differential Mutation Operation

For the solutions of middle class, their information has good pattern, which are hoped to incorporated into the mutation process. There are many mutations strategies in differential evolution algorithm. The operator with the randomly selected solution and the best solution is chosen. To enhance the performance, another different search behaver is also adopted for these solutions with crossover rate P_C. Differential mutation operation is described as Algorithm 4.

Algorithm 4: Differential Mutation operation

1: For i=1 to N do
2:      Select 3 solutions (H_{i1}, H_{i2}, H_{i3}) and a dimension (d_rand) randomly;
3 :   For d=1 to D do
4 :   If rand<mi
5 :   If (rand<=P_C) or (d==d_rand)
6 : $H_i(d) = H_{i1}(d) + F*(H_{i2}(d) - H_{i3}(d))$;
7 :   Else
8 : $H_i(d) = H_{min}(d) + rand*(H_{max}(d) - H_{min}(d))$;
9 :   End if
10 :   Else
11 :   If (rand<=P_C) or (d=d_rand)
12 : $H_i(d) = H_{i1}(d)$;
13 :   Else
14 : $H_i(d) = H_{best}(d) + F*(H_{i2}(d) - H_{i3}(d))$;
15 :   End if
16:         End if
17:     End for
18:End for

---

where $H_{best}(d)$ is the *d*-th component of the best solution, $H_{max}(d)$ and $H_{min}(d)$ are the maximum and the minimum of the *d*-th component of all solutions.

3.2.2 Random Mutation Operation

For inferior class, their beneficial information is few. So random mutation with a self-adaptive interval enclosed by the current population is adopted here. Mutates between the maximum and the minimum of the one dimension is randomly chosen from the current population which is described as Algorithm 5.

Algorithm 5:   Random Mutation
1: For i=1 to N do
2:     For d=1 to D do
3:         If $H_i(d)$ is selected with probability m_i;
4:             $H_i(d)=H_{min}(d)+rand*(H_{max}(d)-H_{min}(d))$;
5:         End if
6:     End for
7: End for

3.3 Elitism Strategy

To preserve the current best solution, elitism strategy is adopted. Two best solutions are chosen for elitism strategy and not to participate in the migration operation.

3.4 MTBBO Optimization Process

Based on the above operations, MTBBO algorithm with momentum migration operation and taxonomic mutation is as follows:

(1) Initialize *N* habitats randomly and the necessary parameters;
(2) Sort and calculate the relevant parameters: *N* habitats are sorted according to *HSI*. Calculating the indicators for each habitat, number of species, emigration rate, immigration rate, species probability and mutation rate. The best quarter solutions are classified into promising class, the worst quarter are classified into inferior class and the rest belongs the middle class;
(3) Migration operation: Elitist strategy and the rests habitats take part in migration operation according to Algorithm 3;
(4) Mutation operation: Middle class carries on differential mutation according to Algorithm 4, inferior class carries on random mutation according to Algorithm 5;
(5) Selection operation based on HSI;
(6) Determine whether the termination criterion is satisfied.

4 Numerical Experiment and Analysis

To verify the performance of MTBBO, MTBBO is compared with classical BBO [1] and BBO-EP [3] based on CEC2014 benchmark functions [16].

4.1 Benchmark Functions and Parameter Settings

All functions are selected from CEC2014 benchmark suite [16] as follows. f1 and f3 are unimodal functions. f14 and f15 are simple multimodal functions. f17, f19, f20 and f21 are hybrid functions. f26 and f30 are composition functions. For convenience, these functions are relabeled as F1–F10. In this paper, population size N = 50, dimension D = 50, search ranges are all $[-100, 100]^D$, 30 independent runs conducted in MATLAB 2017b, momentum coefficient $\alpha = 0.99$, learning rate $\eta = 0.001$, crossover rate $P_C = 0.5$, difference coefficient F = 0.6.

4.2 Numerical Experiment

Results of BBO, BBO-EP and MTBBO are statistically shown in Table 1. "Min", "Mean", "Median" and "STD" indicate the minimum function error, the mean function error, the median function error and the standard deviation, respectively. The numbers achieving the best Mean of three algorithms are listed in the last row.

Table 1. Experimental Results of BBO, BBO-EP and MTBBO on CEC2014 Benchmark.

Fun	Item	BBO	BBO-EP	MTBBO
F1	Min	6.2770E+06	2.0654E+06	**2.7712E+05**
	Mean	1.3915E+07	5.2787E+06	**9.5158E+05**
	Median	1.2998E+07	4.5628E+06	**7.5741E+05**
	STD	5.6907E+06	2.4820E+06	**6.0452E+05**
F2	Min	4.3809E+03	3.4642E+03	**3.4908E+02**
	Mean	1.4757E+04	1.1885E+04	**3.2182E+03**
	Median	1.4349E+04	1.1369E+04	**1.8839E+03**
	STD	6.8686E+03	4.1480E+03	**3.0363E+03**
F3	Min	1.4003E+03	1.4003E+03	**1.4002E+03**
	Mean	1.4005E+03	1.4004E+03	**1.4003E+03**
	Median	1.4004E+03	1.4004E+03	**1.4003E+03**
	STD	2.4567E−01	1.8716E−01	**3.8149E−02**
F4	Min	1.5143E+03	1.5151E+03	**1.5078E+03**
	Mean	1.5256E+03	1.5260E+03	**1.5168E+03**
	Median	1.5244E+03	1.5241E+03	**1.5160E+03**
	STD	6.7787E+00	7.6171E+00	**5.3862E+00**
F5	Min	1.6847E+06	3.3400E+05	**2.4812E+04**
	Mean	4.4889E+06	1.5499E+06	**3.4019E+05**
	Median	4.1148E+06	1.4566E+06	**2.1782E+05**
	STD	2.3226E+06	8.3385E+05	**3.2738E+05**
F6	Min	1.9221E+03	1.9174E+03	**1.9108E+03**
	Mean	1.9507E+03	1.9195E+03	**1.9155E+03**
	Median	1.9465E+03	1.9192E+03	**1.9156E+03**
	STD	1.4230E+01	**1.4798E+00**	2.5134E+00
F7	Min	5.5132E+03	4.5715E+03	**2.2905E+03**
	Mean	1.7698E+04	1.4706E+04	**6.7526E+03**
	Median	1.5233E+04	1.2871E+04	**4.6224E+03**
	STD	7.6960E+03	8.2104E+03	**6.4113E+03**
F8	Min	1.0547E+06	4.0301E+05	**2.5053E+04**
	Mean	4.7210E+06	2.1493E+06	**2.1726E+05**
	Median	3.6827E+06	2.1358E+06	**1.5151E+05**
	STD	2.9399E+06	9.5448E+05	**2.8995E+05**

(*continued*)

Table 1. (*continued*)

Fun	Item	BBO	BBO-EP	MTBBO
F9	Min	2.7004E+03	**2.7003E+03**	**2.7003E+03**
	Mean	2.7472E+03	2.7327E+03	**2.7005E+03**
	Median	2.7007E+03	2.7006E+03	**2.7005E+03**
	STD	5.7614E+01	6.0570E+01	**1.1179E−01**
F10	Min	1.2787E+04	1.2688E+04	**1.1365E+04**
	Mean	1.8686E+04	1.6709E+04	**1.5014E+04**
	Median	1.8453E+04	1.6804E+04	**1.4848E+04**
	STD	3.5709E+03	2.2759E+03	**2.1685E+03**
Number of best Mean		0	0	10

Generally speaking, it can be observed from Table 1 that MTBBO achieves best results from all the functions when comparing with its competitors. For unimodal function and simple multimodal function, MTBBO is significantly better than BBO and BBO-EP. For hybrid function and composition function, MTBBO is slightly superior to BBO and BBO-EP.

4.3 Converging Curves of the Average Best Fitness

To visually examine the evolutionary trends of three algorithms, eight converging curves of the average best fitness are illustrated in Fig. 1.

In Fig. 1, x-axis shows the number of calculated function values and y-axis shows average function values. It can be seen that MTBBO has best searching ability, especially for unimodal function and hybrid function. The evolutionary trend curves in Fig. 1 and the experimental in Table 1 are cooperative and support each other.
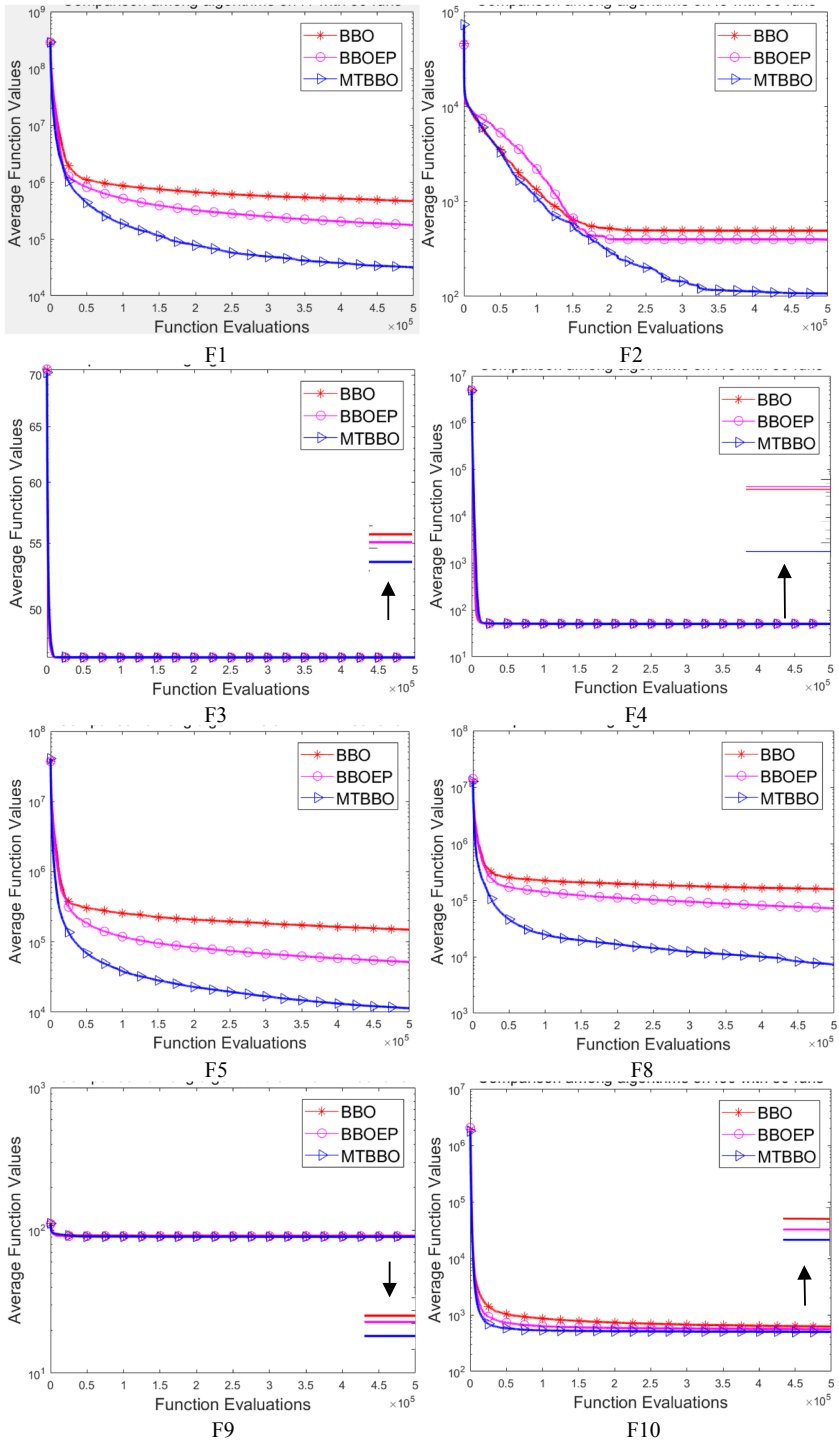
Fig. 1. Converging curves of the average best fitness.

5   Conclusion

Inspired by the momentum idea, momentum is introduced migration operation of BBO, and the momentum migration operator is proposed. An individual depended taxonomic mutation operation is implemented to different solutions. It divides solutions into three categories and difference individuals adopt different mutation operations, respectively. Finally, a novel BBO variant is proposed based on momentum migration and taxonomic mutation. In future, more properties and combining methods of momentum will be considered.

References

1. Simon, D.: Biogeography-based optimization. IEEE Trans. Evol. Comput. **12**, 702–713 (2008)
2. Ma, H., Simon, D.: Blended biogeography-based optimization for constrained optimization. Eng. Appl. Artif. Intell. **24**, 517–525 (2011)
3. Cai, Z., Gong, W., Ling, C.-X.: Research on a novel biogeography-based optimization algorithm based on evolutionary programming. Syst. Eng. Theory Pract. **30**, 1106–1112 (2010)
4. Ma, H.: An analysis of the equilibrium of migration models for biogeography-based optimization. Inf. Sci. **180**, 3444–3464 (2010)
5. Bhattacharya, A., Chattopadhyay, P.K.: Biogeography-based optimization for different economic load dispatch problems. IEEE Trans. Power Syst. **25**, 1064–1077 (2010)
6. Reihanian, A., Feizi-Derakhshi, M.-R., Aghdasi, H.S.: NBBO: a new variant of biogeography-based optimization with a novel framework and a two-phase migration operator. Inf. Sci. **504**, 178–201 (2019)
7. Zhang, S., Xu, S., Zhang, W.: A hybrid approach combining an extended BBO algorithm with an intuitionistic fuzzy entropy weight method for QoS-aware manufacturing service supply chain optimization. Neurocomputing **272**, 439–452 (2018)
8. Zhang, W., Liu, T., Zhang, Y., Jiang, Z.: Research on biogeography-based optimization algorithm based on self-adaptive performance. Comput. Simul. **35**, 277–282 (2018)
9. Saremi, S., Mirjalili, S., Lewis, A.: Biogeography-based optimisation with chaos. Neural Comput. Appl. **25**, 1077–1097 (2014)
10. Jaafari, A., Panahi, M., Pham, B.T.: Meta optimization of an adaptive neuro-fuzzy inference system with grey wolf optimizer and biogeography-based optimization algorithms for spatial prediction of landslide susceptibility. Catena **175**, 430–445 (2019)
11. Bhattacharya, A., Chattopadhyay, P.K.: Application of biogeography-based optimisation to solve different optimal power flow problems. IET Gener. Transm. Distrib. **5**, 70–80 (2011)
12. Alweshah, M.: Construction biogeography-based optimization algorithm for solving classification problems. Neural Comput. Appl. **31**, 5679–5688 (2019)
13. Tong, Z., Chen, H., Deng, X.: A novel task scheduling scheme in a cloud computing environment using hybrid biogeography-based optimization. Soft. Comput. **23**, 11035–11054 (2019)
14. Xiao, J., Zhang, W., Zhang, S.: Game theory-based multi-task scheduling in cloud manufacturing using an extended biogeography-based optimization algorithm. Concurr. Eng. Res. Appl. **27**, 314–330 (2019)

15. Sutskever, I., Martens, J., Dahl, G., Hinton, G.: On the importance of initialization and momentum in deep learning. In: Proceedings of the 30th International Conference on International Conference on Machine Learning, (ICML 2013), USA, vol. 28 (2013)
16. Liang, J.J., Qu, B.Y., Suganthan, P.N.: Problem definitions and evaluation criteria for CEC2014 special session and competition on single objective real-parameter numerical optimization, Nanyang Technological University (Singapore) and Zhengzhou University, China (2013)