



Improved Bacterial Foraging Optimization Algorithm with Comprehensive Swarm Learning Strategies

Xiaobing Gan and Baoyu Xiao(✉)

College of Management, Shenzhen University, Shenzhen 518060, China
winsonxiao2019@163.com

Abstract. Bacterial foraging optimization (BFO), a novel bio-inspired heuristic optimization algorithm, has been attracted widespread attention and widely applied to various practical optimization problems. However, the standard BFO algorithm exists some potential deficiencies, such as the weakness of convergence accuracy and a lack of swarm communication. Owing to the improvement of these issues, an improved BFO algorithm with comprehensive swarm learning strategies (LPCBFO) is proposed. As for the LPCBFO algorithm, each bacterium keeps on moving with stochastic run lengths based on linear-decreasing Lévy flight strategy. Moreover, illuminated by the social learning mechanism of PSO and CSO algorithm, the paper incorporates cooperative communication with the current global best individual and competitive learning into the original BFO algorithm. To examine the optimization capability of the proposed algorithm, six benchmark functions with 30 dimensions are chosen. Finally, experimental results demonstrate that the performance of the LPCBFO algorithm is superior to the other five algorithms.

Keywords: Bacterial foraging optimization · Lévy flight · Comprehensive swarm learning strategies · Function optimization

1 Introduction

Bacterial foraging optimization (BFO) [1], a novel nature-oriented algorithm proposed by Passino, mainly simulates the behaviors of *Escherichia coli* in the process of searching for nutrients. During the foraging context, bacteria generally take four significant actions involved in chemotaxis, reproduction, elimination, and dispersal. Besides that, as for function optimization, the position of one bacterium can be regarded as a feasible solution in the search region. Bacteria adjust their own positions by tumbling based on random directions and swimming with certain step sizes to constantly find out the optimal location. Because of many advantages like the strong robustness and good performance in local search, the BFO algorithm has gradually become popular and until now, it has been applied to sorts of practical fields such as facility layout [2], feature selection [3], training kernel extreme learning machine [4] and so on.

At present, relevant theoretical researches about the BFO algorithm are still in the initial stage. In other words, compared with the other traditional swarm intelligent algorithms such as particle swarm optimization (PSO) [5] and genetic algorithm (GA) [6], the development of the BFO algorithm is not enough mature. Moreover, researches have consistently shown that the BFO algorithm has poor capability both in convergence rate and optimization accuracy, especially for high-dimensional function problems. To elevate the performance of the original algorithm, plenty of improved methods and distinguished hybrid mechanisms [7–11] have been brought up. On the one hand, as the significance of the process of chemotaxis, numerous strategies were proposed to enhance it. Niu et al. [7, 8] improved the chemotactic step size through linear and non-linear decreasing strategies. Experimental results verified the improved algorithm had better performance than the standard BFO. In [9], Wang et al. proposed the other BFO variant (BFO-DX) that incorporates a novel mechanism of progressive exploitation into the chemotaxis operator for improving the ability of local exploration. Another part of the improvement is for the hybridization of the BFO algorithm with other algorithms. Biswas et al. [10] demonstrated an efficient optimization technique including BFO with PSO operator to enhance swarm learning and the global search ability of the original BFO algorithm. With the purpose to tackle with optimization problems more effectively, Zhao et al. [11] combined the gravitational search method with the BFO algorithm and then applied the proposed algorithm (EBFO) into optimizing the Lorenz system.

Although the above-proposed strategies have made great progress, there might have some potential demerits like poor performance of the convergence rate. As for the BFO algorithm, the chemotactic step size is set as a constant number regardless of what the condition of each bacterium is in. Aiming at coping with these disadvantages effectively, the paper introduces a linear-decreasing Lévy flight strategy to randomly generate the length of the motion route for each bacterium, which helps to balance the local and global search. More importantly, comprehensive swarm learning consisting of the cooperative communication with the current global best bacterium and the competitive learning mechanisms are adopted in the paper so as to improve the convergence accuracy of the BFO algorithm and furtherly increase the diversity of the general community to effectively alleviate the premature matter.

The rest of this paper is structured as follows: Sect. 2 briefly introduces relevant principles of the standard BFO algorithm. Section 3 details the proposed LPCBFO algorithm. Experimental results are described in Sect. 4. The final section draws the conclusion and future work.

2 The Standard Bacterial Foraging Optimization Algorithm

As a new heuristic optimization technique, the BFO algorithm [1] properly imitates the crucial actions of *E. coli* generating in the process of obtaining food, mainly including chemotaxis, reproduction, elimination, and dispersal.

2.1 Chemotaxis

Owing to far away from harmful materials efficiently and obtain nutrients in a faster way, each bacterium gradually moves towards the objectives through tumbling and swimming.

On the one hand, during choosing one direction randomly in the search space, bacteria continuously move on with fixed run lengths. After tumbling, bacteria could not insist on swimming along the same search direction until the updated position gets worse or the number of practicable movements is up to the limitation N_s . The new position of the bacterium i in $j + 1$ th chemotaxis, k th reproduction and l th elimination and dispersal $\theta^i(j + 1, k, l)$ is shown as the Eq. (1) where $\theta^i(j, k, l)$ represents the last position of the bacterium i ; $C(i)$ is the step size and $\Delta(i)$ means a random direction vector whose all elements range from -1 to 1 .

$$\theta^i(j + 1, k, l) = \theta^i(j, k, l) + C(i) \times \frac{\Delta(i)}{\sqrt{\Delta^T(i)\Delta(i)}}. \quad (1)$$

2.2 Reproduction

Abiding by the main idea about “Survival of the Fittest” of Darwin’s Evolutionary Theory, the process of reproduction in the BFO algorithm mirrors that the healthier bacteria are more likely to have the remarkable capability of reproduction to maintain the whole of swarm population while poor-nourished individual will be eliminated in the end. In the BFO algorithm, the health degree of the bacterium i is noted as $f_{i,health}$ that can be measured by the sum of fitness value among its lifecycle. Relatively, the corresponding mathematical expression can be presented as (2).

$$f_{i,health} = \sum_{j=1}^{N_c} J(i, j, k, l). \quad (2)$$

where N_c represents the total number of chemotaxis as the lifecycle of the bacterium i ; $J(i, j, k, l)$ is the fitness value of the bacterium i in j th chemotaxis, k th reproduction, l th elimination and dispersal. Then through ascending order for the health value of all bacterium, half of healthier bacteria ($S_r = SS/2$) are able to split into two bacteria whose both of them have the same position while the rest of bacteria are given up.

2.3 Elimination and Dispersal

In reality, due to the dramatic change of its living environment all the time, bacteria might be confronted with lots of unpredicted risks including dynamic temperature change in the local region or invasion of kinds of detrimental substances. Therefore, when suffering from these adverse and unexpected conditions, a part of bacteria needs to disperse to another favorable location as soon as possible. Based on it, after the process of reproduction, the bacterium i randomly migrates to another new position θ' with a certain probability P_{ed} , otherwise it remains in the current location.

3 LPCBFO Algorithm

3.1 Linear-Decreasing Lévy Flight Strategy

Lévy flight strategy [12] sheds light on a stochastic motion process of certain objects in the search environment where the run length C strictly complies with Lévy distribution

probability. The related mathematical equation [13] can be defined as (3) where μ ranges from 1 to 3.

$$f(C) = C^{-\mu}. \quad (3)$$

Up to now, the Lévy flight algorithm has been broadly used for imitating the trajectory of foraging behaviors about many creatures like bumblebees [14], and fruit flies [15] and so far, some favorable research developments have been achieved. Moreover, during the process of searching based on the Lévy flight method, objects can move on with smaller step sizes frequently and larger lengths occasionally, which is to a certain extent beneficial to balance the local exploration and global exploitation for optimization problems. According to these advantages of the Lévy flight strategy, the paper tries to add it to promote $C(i)$ as the Eq. (4) [12]. Furthermore, inspired by [7], we adopt the linear-decreasing Lévy flight strategy followed as the Eq. (5), seeking to improve the convergence accuracy of the standard BFO algorithm.

$$C(i) = \frac{u}{|v|^{1/\beta}}, \beta \in [0.3, 1.99], u \sim N(0, \sigma_u^2), v \sim N(0, 1). \quad (4)$$

$$\sigma_u = \left\{ \frac{\Gamma(1 + \beta) \sin\left(\frac{\pi\beta}{2}\right)}{\Gamma\left(\frac{1+\beta}{2}\right) \times 2^{\frac{\beta-1}{2}} \times \beta} \right\}^{1/\beta}.$$

$$C'(i) = \left(C_{min} + \frac{iter_{max} - iter_{current}}{iter_{max}} \right) \times C(i). \quad (5)$$

$$\theta^i(j+1, k, l) = \theta^i(j, k, l) + C'(i) \times \frac{\Delta(i)}{\sqrt{\Delta^T(i) \Delta(i)}}. \quad (6)$$

3.2 Cooperative Learning Strategy

As for the original BFO algorithm, there is a lack of swarm communication in the whole of the bacterial population, which may have a great negative impact on the convergence capability and accuracy. By contrast, when having great opportunities to communicate with other individuals during the process of searching for nutrients, bacteria could obtain so enough useful information that they could make appropriate adjustments for their current position in time. Consequently, with illumination by the group learning in PSO [5] and the hybridization algorithm based on PSO operator [10], the paper also incorporates corresponding cooperative learning mechanisms into the BFO algorithm. After tumbling and swimming of each bacterium, they have a chance to learn from the global best particle θ_{best} to adjust their current positions. Additionally, the linear-decreasing inertia weight [16] is conducted to modulate the velocity $v'(i)$ of each individual. The updating position of bacterium i $\theta^i(j+1, k, l)$ is as follows:

$$v'(i) = \left[\omega_{max} - \frac{iter_{current} \times (\omega_{max} - \omega_{min})}{iter_{max}} \right] \times v_i^{last} + c \times rand \times (\theta_{best} - \theta^i(j+1, k, l)). \quad (7)$$

$$\theta^i(j+1, k, l) = v^i(i) + \theta^i(j+1, k, l). \quad (8)$$

where c means the study rate; θ_{best} represents the current global best position; $iter$ means iterations.

3.3 Competitive Learning Strategy

Although the cooperative swarm learning mechanism can enhance the capability for global exploitation and the rate of convergence, bacteria are easy to trap into the local best solution. With regarding in the enhancement of premature problem effectively and improve the diversity of the whole bacterial population, we are inspired by the main principle of competitive swarm optimization (CSO) [17] which is proposed by Cheng and then adopt the pairwise bacterial competitive mechanism into the basic BFO algorithm. Firstly, when all of the bacteria SS accomplished the process of chemotaxis, they are randomly divided into $SS/2$ couples. After pairwise competition in each couple, the better individual called winner directly go to the next chemotaxis process while the loser with worse fitness value updates its current position by learning from the winner and then performs the next chemotaxis step. The competitive learning mechanism can be represented as:

$$\begin{aligned} v^{loser}(j+1, k, l) = & \varepsilon \{ rand1 \times v^{loser}(j, k, l) + rand2 \times [\theta_w(j, k, l) - \theta_{loser}(j, k, l)] \\ & + \varphi \times rand3 \times [\theta_{w_{center}}(j, k, l) - \theta_{loser}(j, k, l)] \}. \end{aligned} \quad (9)$$

$$\theta^{loser}(j+1, k, l) = v^{loser}(j+1, k, l) + \theta^{loser}(j+1, k, l). \quad (10)$$

where $\varepsilon = \frac{2}{2-c-\sqrt{c^2-4c}}$ [18], $\varphi = \varphi_{min} + (\varphi_{max} - \varphi_{min}) \times rand$, ε, φ are constraint factors; $\theta_w(j, k, l)$ is the position of the more competitive bacterium; $\theta_{w_{center}}(j, k, l)$ is the average position vector of total winners (Table 1).

4 Experimental Results and Discussion

To testify the optimization efficiency of the proposed LPCBFO algorithm, the convergency results of the LPCBFO algorithm are assessed by comparing with the original bacterial foraging optimization(BFO) [1] and other BFO variants involved the basic BFO algorithm with linear-decreasing strategy(BFO-LDC) [7], the algorithm improved by nonlinear-decreasing chemotactic step size(BFO-NDC) [8], the algorithm based on linear-decreasing Lévy flight strategy(LBFO) as well as the hybrid algorithm with PSO operator(BSO) [10]. In this paper, the population of bacteria SS for all involved algorithms is set in 50 and the total number of the process of chemotaxis N_C , reproduction N_{re} , elimination and dispersal N_{ed} respectively are 1000, 5 and 2. Thus, the total iteration in the experiment is $N_C \times N_{re} \times N_{ed} = 10000$. Besides, we also set $N_s = 4$, $P_{ed} = 0.25$. As for specified parameters settings in the LPCBFO algorithm, we set $C_{min} = 0.01$, $\beta = 1.99$, $\omega_{max} = 0.9$, $\omega_{min} = 0.4$, $\varphi_{min} = 0.01$, $\varphi_{max} = 0.2$, $c = 1.5$. In addition, other relevant parameters in the above improved BFO algorithms are as

Table 1. The pseudo code of the proposed LPCBFO algorithm

```

Initialize parameters and swarm of bacteria  $\theta^i(j, k, l), i \in (1, 2, 3, \dots, SS)$ 
Evaluate the fitness value  $J_i(j, k, l), i \in (1, 2, 3, \dots, SS)$ , the global best position  $\theta_{best}$  and
the global best fitness value  $J_{best}$ 
For  $l = 1 : N_{ed}$ 
    For  $k = 1 : N_{re}$ 
        For  $j = 1 : N_c$ 
            For  $i = 1 : SS$ 
                Update  $J_i(j, k, l)$ 
                The bacterium  $i$  updates its position with the equation (6)
                Calculate the fitness value  $J_i(j + 1, k, l)$ 
                Let  $m = 0$  (initialize counter for swim length)
                While  $m < N_s$ 
                     $m = m + 1$ 
                    IF the updated fitness value  $J_i(j + 1, k, l)$  gets better
                        The bacterium  $i$  preforms swimming using the equation (6)
                    End
                End
                Perform the cooperative swarm learning using equations (7), (8)
                Update  $J_i(j + 1, k, l), \theta_{best}$  and  $J_{best}$ 
            End
            Divide all bacterium into  $SS/2$  couples ( $SS$  is an even number)
            Compare bacteria' fitness value of each couple respectively
            Perform pairwise competitive learning mechanism using equations (9), (10)
        End
        Calculate the health value for each bacterium  $f_{i,health}$  using the equation (2)
        Sort out the bacteria  $f_{i,health}$  in ascending order and perform reproduction
    End
    The bacterium  $i$  is randomly located on a new position  $\theta'$  with probability  $P_{ed}$ 
End

```

follows. BFO-LDC settings: $C_{min} = 0.01$, $C_{max} = 1.5$; BFO-NDC: $C_{min} = 0.01$, $C_{max} = 1.5$, $\lambda = 4$; LBFO: $C_{min} = 0.01$, $\beta = 1.99$; BSO: the step length of each bacterium $iC(i) = 0.1$, $\omega = 0.8$, $c = 1.5$. According to the above parameter settings in each algorithm, six common benchmark functions in 30 dimensions are chosen to adequately identify the performance of the above six algorithms, involving four unimodal functions (Sphere, Rosenbrock, Schwefel's Problem and Sum of different powers) as well as two multimodal functions (Rastrigin, Ackley function). With regarding to the improvement of the results' reliability and availability, each algorithm is fully operated in 10 times for six benchmark functions.

After 10 runtimes in 30 dimensions, relevant experimental data results are illustrated in Table 2. In Table 2, there mainly contain four necessary measurement metrics including the minimum solution ('Best'), the worst fitness value ('Worst'), the average value ('Mean'), and standard deviation ('Std'). Additionally, the convergence results about six test functions with 30 dimensions are respectively shown in Fig. 1. Generally, in the initial iteration of some benchmark functions like Rosenbrock and Sum of different powers, the LPCBFO algorithm has slower convergence speed than others. This phenomenon

might be contributed by the reason that bacteria with longer movement length in the early iteration stage are more committed to the global exploitation while they are poor in local search. However, we can observe that during the latter search period, the LPCBFO algorithm has greater capability to escape from the local best solution and its convergence accuracy is obviously more excellent than other algorithms'. It is likely to be accounted for comprehensive learning mechanisms and the enhancement of the diversity of the bacterial population.

Table 2. Comparison between LPCBFO and other algorithms with 30 dimensions

	BFO	BFO-LDC	BFO-NDC	LBFO	BSO	LPCBFO
Sphere	3.68e-01	6.26e-02	3.21e-03	5.88e-03	5.10e-06	1.39e-92
	5.62e-01	1.03e-01	4.72e-03	2.95e-02	2.47e-04	3.83e-78
	4.91e-01	8.57e-02	3.96e-03	1.74e-02	7.06e-05	3.83e-79
	7.07e-02	1.38e-02	5.01e-04	7.77e-03	7.84e-05	1.21e-78
Rosenbrock	6.09e+01	4.18e+01	1.43e+01	2.55e+01	2.29e+01	5.99e-02
	7.79e+01	7.22e+01	2.47e+01	2.81e+01	1.28e+02	4.08e+00
	7.10e+01	5.22e+01	2.12e+01	2.68e+01	3.69e+01	1.71e+00
	5.27e+00	8.10e+00	3.34e+00	1.10e+00	3.19e+01	1.99e+00
Schwefel	3.32e+00	2.42e+02	1.01e+02	1.25e+02	9.18e-02	6.30e-35
	5.14e+00	3.49e+04	2.65e+05	8.36e+02	2.52e-01	8.28e-28
	3.95e+00	1.04e+04	4.31e+04	3.00e+02	1.37e-01	1.01e-28
	5.03e-01	1.09e+04	8.73e+04	2.51e+02	5.36e-02	2.61e-28
Sum	1.58e-04	1.76e+03	1.21e+04	1.24e-07	1.49e-09	1.73e-96
	6.63e-04	5.48e+04	6.33e+06	3.20e-07	1.45e-07	8.38e-61
	3.91e-04	1.82e+04	9.32e+05	2.29e-07	2.22e-08	8.38e-62
	1.72e-04	1.82e+04	1.95e+06	5.96e-08	4.38e-08	2.65e-61
Rastrigin	1.07e+02	2.65e+02	1.48e+02	1.56e+02	2.99e+01	1.49e+01
	1.51e+02	3.50e+02	2.35e+02	1.86e+02	4.59e+01	2.49e+01
	1.36e+02	3.11e+02	1.84e+02	1.74e+02	3.73e+01	1.92e+01
	1.39e+01	2.74e+01	2.73e+01	9.71e+00	5.00e+00	3.22e+00
Ackley	1.68e+01	1.06e+01	1.83e+01	1.80e+01	6.18e+00	1.16e+00
	1.77e+01	1.96e+01	1.91e+01	1.806e+01	1.16e+01	4.30e+00
	1.74e+01	1.84e+01	1.89e+01	1.803e+01	9.49e+00	2.407e+00
	2.96e-01	2.77e+00	2.79e-01	1.48e-02	1.77e+00	8.26e-01

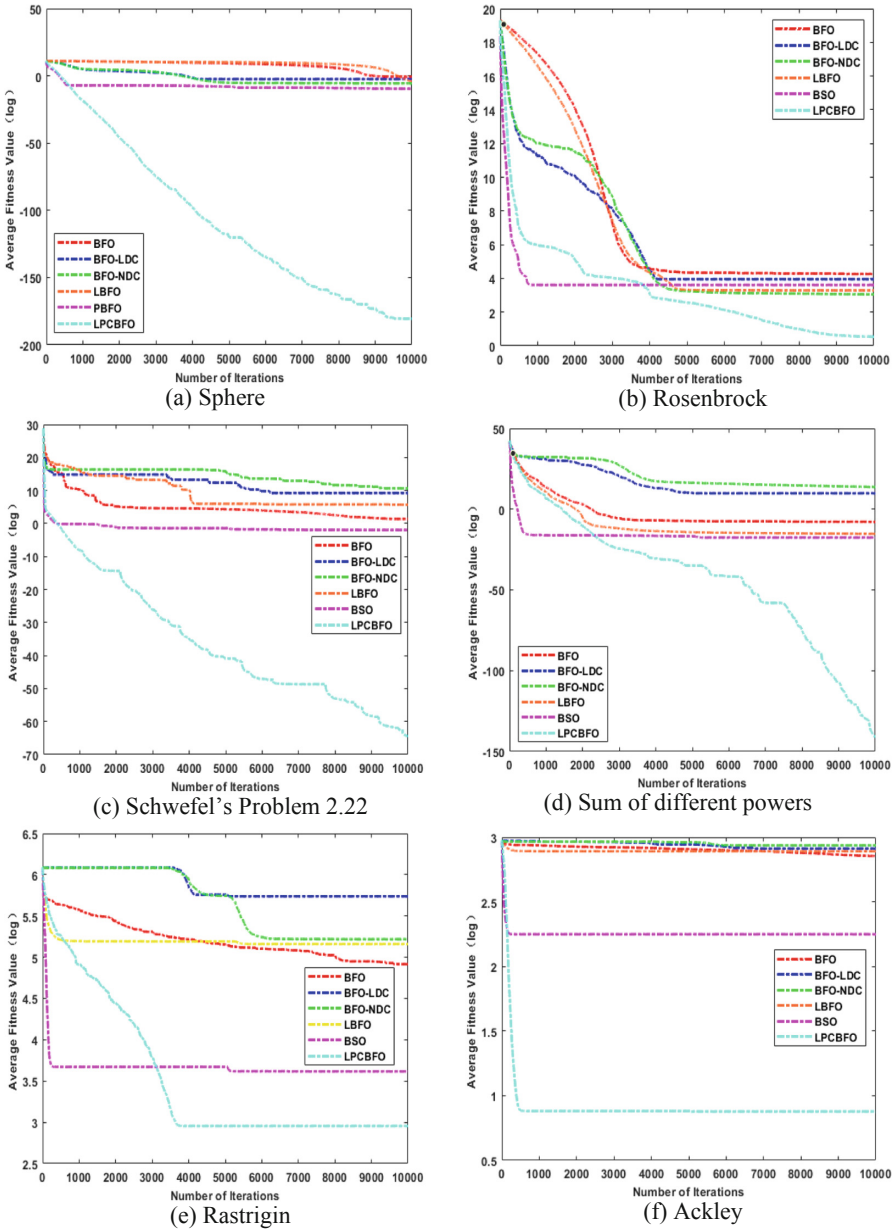


Fig. 1. Convergence results of six different algorithms with 30 dimensions

5 Conclusion and Future Work

In the paper, the improved bacterial foraging optimization with comprehensive swarm learning mechanisms (LPCBFO) is proposed. Compared with the standard BFO algorithm, the paper incorporates the linear-decreasing Lévy flight method to randomly generate the run length of each bacterium, which is in favor of balancing the local exploration and global exploitation. Depending on improving the convergence speed and solution accuracy of the algorithm, the cooperative swarm strategy during learning with the current optimal individual is fully considered. To avoid the premature issue effectively and promote the diversity of the overall population, each bacterium can be pairwise allocated randomly to conduct competitive learning. Finally, six benchmark functions with 30 dimensions are chosen to measure the performance of the proposed LPCBFO algorithm compared with the BFO algorithm and the other four variants. Experimental results show that the optimization effectiveness of the LPCBFO algorithm outperforms others.

In the future, we will contribute to the improvement of the BFO algorithm and attempt to deal with practical problems likes airline scheduling, logistic delivery, and vehicle routing planning by improved algorithms.

References

1. Passino, K.M.: Biomimicry of bacterial foraging for distributed optimization and control. *IEEE Control Syst. Mag.* **22**, 52–67 (2002)
2. Turanoglu, B., Akkaya, G.: A new hybrid heuristic algorithm based on bacterial foraging optimization for the dynamic facility layout problem. *Expert Syst. Appl.* **98**, 93–104 (2018)
3. Chen, Y., Li, Y., Wang, G., et al.: A novel bacterial foraging optimization algorithm for feature selection. *Expert Syst. Appl.* **83**, 1–17 (2017)
4. Chen, H., Zhang, Q., Luo, J., et al.: An enhanced bacterial foraging optimization and its application for training kernel extreme learning machine. *Appl. Soft Comput.* **86**, 1–24 (2020)
5. Eberhart, R., Kennedy, J.: Particle swarm optimization. In: *Proceedings of the IEEE International Conference on Neural Networks*, pp. 1942–1948. IEEE Press, New York (1995)
6. Holland, J.H.: *Adaptation in Natural and Artificial Systems: an Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence*. MIT Press (1992)
7. Niu, B., Fan, Y., Wang, H., et al.: Novel bacterial foraging optimization with time-varying chemotaxis step. *Int. J. Artif. Intell.* **7**, 257–273 (2011)
8. Niu, B., Wang, H., Tan, L., et al.: Improved BFO with adaptive chemotaxis step for global optimization. In: *2011 Seventh International Conference on Computational Intelligence and Security*, pp. 76–80. IEEE Press, New York (2011)
9. Wang, D., Qian, X., Ban, X., et al.: Enhanced bacterial foraging optimization based on progressive exploitation toward local optimum and adaptive raid. *IEEE Access* **7**, 95725–95738 (2019)
10. Biswas, A., Dasgupta, S., Das, S., Abraham, A.: Synergy of PSO and bacterial foraging optimization—a comparative study on numerical benchmarks. In: Corchado, E., Corchado, J.M., Abraham, A. (eds.) *Innovations in Hybrid Intelligent Systems*. ASC, vol. 44, pp. 255–263. Springer, Heidelberg (2007). https://doi.org/10.1007/978-3-540-74972-1_34
11. Zhao, W., Wang, L.: An effective bacterial foraging optimizer for global optimization. *Inf. Sci.* **329**, 719–735 (2016)

12. Mantegna, R.N.: Fast, accurate algorithm for numerical simulation of Lévy stable stochastic processes. *Phys. Rev. E* **49**, 4677 (1994)
13. Viswanathan, G.M., Raposo, E.P., Da Luz, M.G.E.: Lévy flights and super-diffusion in the context of biological encounters and random searches. *Phys. Life Rev.* **5**, 133–150 (2008)
14. Edwards, A.M., Phillips, R.A., Watkins, N.W., et al.: Revisiting Lévy flight search patterns of wandering albatrosses, bumblebees deer. *Nature* **449**, 1044–1048 (2007)
15. Reynolds, A.M., Frye, M.A.: Free-flight odor tracking in drosophila is consistent with an optimal intermittent scale-free search. *PLoS One* **2**, e354 (2007)
16. Shi, Y., Eberhart, R.: A modified particle swarm optimizer. In: 1998 IEEE International Conference on Evolutionary Computation Proceedings, pp. 69–73. IEEE Press, New York (1998)
17. Cheng, R., Jin, Y.: A competitive swarm optimizer for large scale optimization. *IEEE Trans. Cybern.* **45**, 191–204 (2014)
18. Zhang, L., Yu, H., Hu, S.: A new approach to improve particle swarm optimization. In: Cantú-Paz, E., et al. (eds.) *GECCO 2003. LNCS*, vol. 2723, pp. 134–139. Springer, Heidelberg (2003). https://doi.org/10.1007/3-540-45105-6_12