An Optimized Bat Algorithm Combining Local Search and Global Search

Chunting Li*, Zigang Lian*, Tipeng Zhang
School of Electronics and Information Engineering, Shanghai DianJi University, Shanghai, 201306, China
*a186003010118@st.sdju.edu.cn
*bCorresponding author’s e-mail: 18616832710@163.com

Abstract. This paper proposes an optimized Bat algorithm combining local search and global search to solve the problem of low accuracy of bat algorithm and easy to fall into local optimum in the later stage. The optimized Bat algorithm enhance the diversity of bat algorithm, improve the search accuracy and convergence speed of bat algorithm, And by solving a variety of different types of functions, it is proved that the optimized bat algorithm that combines local search and global search proposed in this paper is feasible.

1. Introduction
Bat Algorithm (BA) [1] is an algorithm for finding the optimal solution proposed by Professor Yang Xin-She in 2010 by simulating bat foraging through echolocation. It has the advantages of simple model, fast convergence speed and strong versatility. However, the bat algorithm has the problems of being easy to fall into local extremum, low solution accuracy, and slow convergence speed in the later stage of the algorithm. In response to these problems, many scholars at home and abroad have made corresponding improvements to the bat algorithm. In 2016, AO Topal et al. proposed a nature-inspired bat algorithm (DVBA) [2]. There are two types of bats in the population: explorer bat and developer bat, which improves the diversification and search performance of bat algorithms. In 2017, Benouaret et al., in order to overcome the premature convergence of the bat algorithm, proposed a directional bat algorithm (dBA) [3] to solve the continuous optimization problem. This algorithm introduced echolocation into the bat algorithm and enhanced the search of the bat algorithm. ability. In 2018, Al-Betar et al. used the island model in the population evolution model to optimize the bat algorithm in order to preserve the diversity of the bat population, and proposed an island-shaped bat heuristic algorithm [4], and carried out the sensitivity of the main parameters of the algorithm. The analysis has verified the influence of the parameters on the convergence speed. In 2019, MR et al. aimed at the situation that the bat algorithm is easy to fall into the local optimal value, and improved the bat algorithm by providing inertial weights to make it have better search performance, and proved through the test function that the improved bat algorithm (mBA) has better performance. Good search performance [5].

2. Bat algorithm
The bat algorithm is a meta-heuristic intelligent search algorithm. Its principle is to initialize the position and velocity of the bat, treat them as the solution in the problem space, and find the optimal fitness function value of the problem. In the iteration, the bat individual Adjust the pulse rate and
loudness, move toward the optimal bat individual, and finally find the optimal solution. This process is called the algorithm search process [6].

Assuming that there are \( n \) bats distributed in the d-dimensional space, each bat is a point in the d-dimensional space. First, initialize the bat colony, determine the initial speed and initial position of each bat in the colony, and update it according to equations (1)–(3). At time \( t \), the frequency of bat \( i \) is \( f_i \), speed is \( v_i \), position is \( x_i \), at time \( t \), the position of the bat \( i \) is updated to \( x_i^{t+1} \), and the speed is updated to \( v_i^{t+1} \), the bat individual’s update formula for its own frequency, speed, and position [35] is as follows:

\[
f_i = f_{\text{min}} + (f_{\text{max}} - f_{\text{min}}) \beta
\]

\[
v_i^{t+1} = v_i^t + (x_i^t - g_{\text{best}}) f_i
\]

\[
x_i^{t+1} = x_i^t + v_i^{t+1}
\]

In the formula: \( f_i \) indicates the frequency of the bat's pulse at the current moment; \( f_{\text{max}} \) indicates the maximum frequency of the bat; \( f_{\text{min}} \) indicates the minimum frequency of the bat; \( \beta \) is a random number in [-1,1]; \( g_{\text{best}} \) indicates the current global optimal individual of the bat Is the current global optimal solution.

The bat algorithm has a local search mechanism. If the bat meets the conditions, the new position of the bat is randomly generated around it. The update formula is as follows:

\[
x_{\text{new}}(i) = x_{\text{old}} + \varepsilon A_i
\]

In the formula, \( x_{\text{old}} \) represents a randomly selected solution from the current optimal solution set; \( \varepsilon \) is a random number in [-1,1]; \( A_i \) represents the average loudness of all bats at time \( t \).

During the bat's predation, the rate and loudness of pulses are constantly updated. As the iteration progresses, the pulse loudness \( A \) will gradually decrease, and the rate of pulse emission \( R \) gradually increases. The update formula is as follows:

\[
A_i^{t+1} = \alpha A_i^t
\]

\[
R_i^{t+1} = R_i^0 \left[ 1 - \exp (-\gamma t) \right]
\]

In the formula, \( 0 < \alpha < 1 \), \( \gamma > 0 \), and both are constant; \( A_i^t = 0 \) indicates the state where the bat temporarily stops transmitting pulses, when \( t \rightarrow \infty \), \( A_i \rightarrow 0 \), \( R_i \rightarrow R_i^0 \).

3. Bat algorithm combining local and global

When using the bat algorithm to solve the optimization problem, there is often more than one extreme value. In the later iteration of the BA algorithm, the bats in the population tend to the optimal area, which reduces the diversity of the population [7], in [8], The bat population moves toward the best bat individual position, which reduces the diversity of the bat population and the bat individual is also easy to fall into a local extreme. In order to solve this problem, this paper proposes the Local and Global Combine Bat Algorithm (Local and Global Combine Bat Algorithm).

3.1. The mathematical model

Suppose that in a d-dimensional target search space, there is a population of bats including n virtual bats. At time \( t \), the frequency, speed and position of bat \( i \) in the population are \( f_i \), \( v_i \) and \( x_i \) respectively. \( g_{\text{bat}} = (p_{g_1}, p_{g_2}, \ldots, p_{g_d}) \) is the optimal position searched by the entire bat population at the current moment, \( i_{\text{bat}} = (x_{i_1}, x_{i_2}, \ldots, x_{i_d}) \) is the historical optimal search position of the bat at the current
moment, \( l_{best} = (p_{l1}, p_{l2}, \ldots, p_{ln}) \) is the optimal position searched by the current population. At time \( t+1 \),
the position of bat \( i \) is updated to \( x_{i}^{t+1} \), and the speed is updated to \( v_{i}^{t+1} \). In LGCBA, the update formula [10] of bat individuals for their own frequency, speed, and position is as follows:

\[
f_{i} = f_{min} + (f_{max} - f_{min}) \beta
\]

\[
v_{i}^{t+1} = v_{i}^{t} + \left(C_{1}(x_{i}^{t} - \cdot) + C_{2}(1 - a^{t}) \cdot (x_{i}^{t} - l_{best}) \right) f_{i}
\]

\[
x_{i}^{t+1} = x_{i}^{t} + v_{i}^{t+1}
\]

Where: \( C_{1} \) and \( C_{2} \) are learning factors, \( C_{1} \) is a constant with a value interval of \([0.1,2]\), and \( C_{2} \) is a constant with a value interval of \([-1,2]\); \( a \) is a constant with a size between \([0,1]\).

The recursive equation of the bat algorithm (LGCBA) that combines local and global search proposed in this paper mainly updates the new speed of bats through four parts: The speed \( v_{i}^{t} \) of the previous iteration of bat \( i \); The distance between the current position of bat \( i \) and the optimal position \( g_{best} \) of the current historical population; The distance between the current position of bat \( i \) and the optimal position \( l_{best} \) in the population at the previous iteration; The distance between the current position of the bat \( i \) and the historical optimal position \( i_{best} \) of the bat. Bat \( i \) uses formulas (5)–(6) to update the position, and continuously updates the next position through the recursive formula.

3.2. The Implementation steps

Assuming that the minimum value of \( f(x) \) is sought, the population size is \( n \), and the number of iterations is \( N \). The basic implementation steps of LGCBA are as follows:

Step1: Initialize the bat population. In the \( d \)-dimensional space, a set of initial solutions are distributed. Maximum pulse loudness \( A^{0} \), maximum pulse rate \( R^{0} \), search pulse frequency range \([f_{min}, f_{max}]\), attenuation coefficient \( \alpha \) for volume, enhancement coefficient \( \gamma \) for search frequency, search accuracy \( \epsilon \) or maximum number of iterations \( N \).

Step2: Initialize the position \( x_{i} \) of the bat randomly, and find the optimal solution \( g_{best} \) of all current bats and the optimal bat position \( l_{best} \) in the current population according to the current fitness value, and record the individual bat at this time as the most historical individual Merit value \( i_{best} \).

Step3: The search pulse frequency, speed and position of the bat are updated according to equations (7)–(9).

Step4: Generate a uniformly distributed random number \( rand \), if \( rand > r \), randomly perturb the current optimal solution, generate a new solution, and judge the new solution out of bounds.

Step5: Generate a uniformly distributed random number \( rand \), if \( rand < A \) and \( f(x_{i}) < f(x') \), accept the new solution generated by Step4, and then update the pulse loudness \( A' \) and the pulse emission rate \( R' \) according to the following formulas (5) and (6).

Step6: Sort the fitness values of all bats and find the current global optimal solution \( g_{best} \) and optimal fitness value.

Step7: Sort the fitness values of all bats and find the current global optimal solution \( g_{best} \) and optimal fitness value.

Step8: Compare all bat individuals in the current population with their individual historical optimal value \( i_{best} \), and update the \( i_{best} \).

Step9: Repeat steps Step2–Step5 until the set optimal solution conditions are met or the maximum number of iterations is reached.

Step10: Output the global optimal value and optimal solution.
4. Simulation experiment

4.1. Parameter setting

Using the classic bat algorithm (BA) and the local and global bat algorithm (LGCBA) proposed in this paper, the maximum number of iterations is 1000, the population of bats is 100, the acoustic loudness $A_0$ is 0.8, and the pulse emission rate $r_0$ is 0.5. When the learning factor, and parameters are set to 1.8, -1, and 0.8, respectively, the 8 functions to be tested are tested with dimensions of 10, 30, and 50 respectively, in order to ensure the objectivity and effectiveness of the experiment The two algorithms run 8 test functions 30 times each.

4.2. Test function

The 8 test functions selected in this chapter have their own characteristics, the graphs in the space are different, and the difficulty of solving is also different. It can test the optimization accuracy of the algorithm and the convergence speed of the algorithm. The specific test functions selected in this article are as follows:

1) Sphere function:
   \[ f_1(x) = \sum_{i=1}^{n} (x_i - 50), \text{ and } x_i \in [-100,100]^n. \]

2) Quadric function:
   \[ f_2(x) = \sum_{i=1}^{n} \left( \sum_{j=1}^{i} x_j \right)^2, \text{ and } x_i \in [-100,100]^n. \]

3) Schwefel function:
   \[ f_3(x) = \max_i \{|x_i|, 1 \leq i \leq n\}, \text{ and } x_i \in [-100,100]^n. \]

4) Rastrigin function:
   \[ f_4(x) = 10 \times n + \sum_{i=1}^{n} (x_i^2 - 10 \times \cos(2 \pi x_i)), \text{ and } x_i \in [-5.12,5.12]^n. \]

5) Ackley function:
   \[ f_5(x) = -20 \exp\left(-0.2 \sqrt{\frac{1}{n} \sum_{i=1}^{n} x_i^2}\right) - \exp\left(\frac{1}{n} \sum_{i=1}^{n} \cos(2 \pi x_i)\right) + 20 + e, \text{ and } x_i \in [-32,32]. \]

6) Penalized function:
   \[ f_6(x) = \frac{\pi}{n} \left(10 \sin^2(\pi y_1) + \sum_{i=1}^{n} (y_i - 1)^2 \times [1 + 10 \sin^2(\pi y_{i+1})] + (y_{n+1} - 1)^2\right) + \sum_{i=1}^{n} u(x_i,10,100,4), \text{ and } u(x,a,k,m) = \begin{cases} 0, & x > a, \\ k(x-a)^m, & -a \leq x \leq a, \text{ and } y_1 = 1 + \frac{1}{4} (x_1 + 1), x_i \in [-50,50]. \end{cases} \]

7) Axis parallel hyper-ellipsoid function:
   \[ f_7(x) = \sum_{i=1}^{n} i \times x_i^2, \text{ and } x_i \in [-5.12,5.12]^n. \]

8) Sum of different power function:
   \[ f_8(x) = \sum_{i=1}^{n} |x_i|^{(i+1)}, \text{ and } x_i \in [-1,1]^n. \]

4.3. Results and analysis

Under the above parameter settings, the bat algorithm (LGCBA) and the basic bat algorithm (BA) combined with local and global search are used to obtain the optimal value and worst value of the above 8 test functions, and bold For solutions with higher accuracy, the specific experimental data are shown in Table 1.
Table 1. The running results of two algorithms on 8 test functions.

| func | Dimen | BAT | LGCBAT | BAT | LGCBAT |
|------|-------|-----|--------|-----|--------|
|      |       | Optimal value | Worst value | Optimal value | Optimal value | Optimal value | Worst value | Optimal value | Optimal value |
| 1    | 10    | 3.6214e-05 | 9.1443e-05 | 6.8164e-06 | 9.2588e-06 | 1.2277e-04 | 2.6348e-04 | 1.2972e-04 |
| f_1  | 20    | 3.0948e-04 | 5.6793e-04 | 4.9955e-05 | 6.0027e-04 | 0.0015 | 6.3776e-04 | 3.8748e-04 | 0.0015 |
|      | 30    | 8.2969e-04 | 0.0015 | 4.1545e-07 | 0.0015 | 0.0024 | 0.0093 | 0.0015 | 0.0064 |
|      | 50    | 0.0033 | 0.0048 | 0.0030 | 0.0045 | 0.0280 | 407.6041 | 0.0219 | 84.0657 |
| 2    | 10    | 0.0025 | 0.0056 | 5.0914e-04 | 0.0057 | 0.0043 | 39.8153 | 1.4271e-04 | 9.9710 |
|      | 20    | 0.0016 | 1.8593 | 1.4711e-05 | 0.7163 | 0.0704 | 461.7633 | 1.2889264 | 0.0137 |
| f_3  | 30    | 0.0063 | 2.7132 | 0.0025 | 1.0938 | 0.3142 | 1.16602e+03 | 0.0373321 |
|      | 50    | 0.0024 | 2.2980 | 1.8094e-04 | 1.0800 | 50  | 19.8093 | 1.6260 | 1.0195e+03 |
|      | 10    | 0.0073 | 6.1326 | 0.0011 | 5.0926 | 30  | 1.3586e-06 | 1.3625e-06 | 8.0423e-07 | 3.1570e-06 |
| 3    | 20    | 0.0153 | 8.6971 | 0.0067 | 6.5609 | 20  | 4.6508e-06 | 3.1099 | 1.3282e-06 | 0.7403 |
|      | 30    | 0.0271 | 11.110 | 0.0067 | 6.5622 | 30  | 8.3157e-06 | 12.2209 | 1.2203e-06 | 6.2323 |
|      | 50    | 0.0406 | 12.083 | 0.0190 | 11.0106 | 50  | 1.8342e-05 | 22.3831 | 1.5160e-05 | 6.4895 |
|      | 10    | 1.6761e-04 | 5.4127 | 8.6689e-05 | 5.1203e-04 | 10  | 9.9927e-10 | 5.0922e-08 | 8.5155e-11 | 4.5264e-06 |
| f_4  | 20    | 0.0029 | 0.0066 | 0.0021 | 0.0068 | 20  | 6.7906e-09 | 4.7395e-07 | 4.5684e-09 | 1.3316e-07 |
|      | 30    | 0.0142 | 0.0365 | 0.0131 | 0.0305 | 30  | 1.1395e+07 | 4.0791e-08 | 2.7359e-08 | 3.6448e-07 |
|      | 50    | 0.1039 | 0.2982 | 0.1059 | 0.5444 | 50  | 1.3901e-07 | 1.2181e+18 | 3.8652e-08 | 3.9083e-07 |

It can be seen from Table 1 that for the 8 classic test functions, both algorithms can find the optimal solution. The optimal solution obtained by the bat algorithm (LGCBA) that combines local and global search proposed in this paper is better than the optimal solution found by the classical bat algorithm. It can be seen that the bat algorithm that combines local and global search proposed in this paper (LGCBA) Higher precision.

The convergence curve of the algorithm reflects the convergence speed of the algorithm and the ability of the algorithm to jump out of local extremes, and is an important indicator to measure the performance of the algorithm. When the maximum number of iterations is 1000, the number of bats in the population is 100, and the dimension is 30, the convergence curves during the process of searching for the optimal solution using two algorithms for the above 8 test functions are shown in Figures 1.
Figure 1. Convergence curves of f1 ~ f8

It can be seen from Figure 1 that the local and global bat algorithm (LGCBA) proposed in this paper converges faster and the minimum fitness value found is less than the minimum fitness value found by the classic bat algorithm.
5. Conclusion
This paper proposes an optimized bat algorithm that combines local search and global search. Improve the search accuracy and convergence speed of the bat algorithm. The superiority of the improved bat algorithm proposed in this paper is proved by solving many different types of functions.

Acknowledgments
While studying for a master's degree, I received help and support from teachers and classmates. Special thanks to Professor Lian Zhigang for his academic guidance and help. I would also like to thank all the authors of the references in this article for allowing me to have a more in-depth research on the bat algorithm and for helping me write this article.

References
[1] Yang X S. A New Metaheuristic Bat-Inspired Algorithm[J]. Computer Knowledge & Technology, 2010, 284:65-74.
[2] Topal A O, Altun O. A novel meta-heuristic algorithm: Dynamic Virtual Bats Algorithm[J]. Information Sciences, 2016: S0020025516301888.
[3] Benouaret, Mohamed, Yang,. New directional bat algorithm for continuous optimization problems[J]. Expert Systems with Application, 2017.
[4] Al-Betar M A, Awadallah M A. Island bat algorithm for optimization[J]. Expert Systems with Application, 2018, 107(OCT.):126-145.
[5] M.R. Ramli, Z. Abal Abas, M.I. Desa, Z. Zainal Abidin, M.B. Alazzam. Enhanced convergence of Bat Algorithm based on dimensional and inertia weight factor[J]. Journal of King Saud University - Computer and Information Sciences,2019,31(4).
[6] YANG X S. A new meta heuristic Bat-Inspired Algorithm[J]. Nature Inspired Cooperative Strategies for Optimization, Spinger, 2010: 65-74.
[7] Cui Xueting, Li Ying, Fan Jiahao. Global chaotic bat optimization algorithm [J]. Journal of Northeastern University (Natural Science Edition), 2020,41(4):488-491,498.
[8] Chakri, Asma, Benouaret, Mohamed, Yang, Xin-She, et al. New directional bat algorithm for continuous optimization problems[J]. Expert Systems with Application,2017,69(Mar.):159-175.
[9] Liu Changping, Ye Chunming. Bat algorithm with Lévy flight characteristics[J]. Journal of Intelligent Systems, 2013, 03:24
[10] Jordehi A R. Chaotic bat swarm optimization (CBSO) [J]. Applied Soft Computing, 2014, 26:523–530.
[11] Meng X B, Gao X Z, Liu Y, et al. A novel bat algorithm with habitat selection and Doppler effect in echoes for optimization [J]. Expert Systems with Applications, 2015, 42(17-18):6350-6364.
[12] Yang X S, A H Gandomi. Bat algorithm: a novel approach for global engineering optimization [M]/ engineering computations, 2012, 29:464-483.