An Improved Chaos Bird Swarm Optimization Algorithm

Damin Zhang¹,*, Juqing Yang² and Ping Yang³

¹,²,³College of Big Data and Information Engineering, Guizhou University, Guiyang, China

*Corresponding author e-mail: 1203813362@qq.com

Abstract: Aiming at the defect that bird-swarm algorithm (BSA) is easily trapped in the local optimum and appears premature convergence for high-dimensional functions, an improved chaos bird-swarm optimization algorithm (LCDE-BSA) is proposed in the paper. Logistic chaotic mapping is used to initialize the population and make the initial distribution to the entire population of space. Population catastrophe is adopted to escape from local optimum. Differential evolution algorithm is used to enhance the population diversity and improve the optimizing efficiency by mutation operation, crossover operation and selection operation on individuals except the current best individual. Experiments on six criteria test functions indicate that LCDE-BSA has better global search ability and convergence properties than BSA.

1. Introduction

Optimization problem is an application research focus and attracts extensive attention from the whole society at present. However, many optimization problems are so complex that it is very difficult to get the optimal solutions within a reasonable time limit. In recent years, many scholars imitate the intelligent behavior of biological systems to propose a lot of swarm intelligence optimization algorithms which provide efficient ways to find the solution of complex problems.

Bird swarm algorithm (BSA) is proposed by Xian-Bing Meng in 2015 [1]. It is a new global optimization algorithm with few adaptable parameters, high convergence accuracy and strong robustness inspired from the simulation of birds’ foraging behavior, vigilance behavior and flight behavior [2-3]. Studies suggested that BSA is good performance to solve extreme problems, but there are local optimal problems and premature convergence for high-dimensional functions [3-4].

In this paper, logistic chaotic mapping, population catastrophe and differential evolution are introduced to bird swarm algorithm to enhance its optimal ability. Logistic chaotic mapping is used to initialize the population and make the initial distribution to the entire population of space. Population catastrophe is adopted to escape from local optimum. Differential evolution algorithm is used to enhance the population diversity by producing temporary initial population. Experiments on selected
criteria test functions are carried out to verify the improved algorithm.

2. Fundamentals

2.1 Bird Swarm Algorithm

BSA is an effective optimization algorithm with the characteristics of simple process, good expansibility and etc.[1-2].

Let \( N \) virtual birds fly and forage for food. Suppose \( x_i^t (i \in \{1, 2, \ldots, N\}) \) express the position of the \( i \)th bird at \( t \). The birds’ behaviors can be described as follows:

(1) Foraging behavior is described as follows:

\[
x_{i,j}^{t+1} = x_{i,j}^t + (p_{i,j} - x_{i,j}^t) \times C \times \text{rand}(0, 1) + (g_{i,j} - x_{i,j}^t) \times S \times \text{rand}(0, 1)
\]

(1)

(2) Vigilance behavior is described as follows:

\[
x_{i,j}^{t+1} = x_{i,j}^t + A_1 (\text{mean}_j - x_{i,j}^t) \times \text{rand}(0, 1) + A_2 (p_{i,j} - x_{i,j}^t) \times \text{rand}(-1, 1)
\]

(2)

Where, \( A_1 \) and \( A_2 \) can be described mathematically as:

\[
A_1 = a_1 \times \exp \left( -\frac{P_{\text{Flt}}}{\text{sum Fit} + \varepsilon} \times N \right)
\]

\[
A_2 = a_2 \times \exp \left( \frac{P_{\text{Flt}} - P_{F_{\text{lim}}}}{P_{F_{\text{lim}}}} \times \text{sum Fit} + \varepsilon \right) \times \text{sum Fit} + \varepsilon
\]

\( a_1 \) and \( a_2 \) are constants in \([0, 2]\). \( \varepsilon \) is a smaller constant.

(3) Flight behavior is described as follows:

\[
x_{i,j}^{t+1} = x_{i,j}^t + \text{randn}(0, 1) \times x_{i,j}^t
\]

(3)

\[
x_{i,j}^{t+1} = x_{i,j}^t + (x_{i,j}^t - x_{i,j}^t) \times FL \times \text{randn}(0, 1)
\]

(4)

Where, FL is in \([0, 2]\).

2.2 Chaotic mapping

The chaotic system has the properties of sensitive to the initial conditions. The chaotic signals generated by deterministic systems have the quality of genus-randomness. Its curve is determined by the initial value and chaos mapping parameters.

Logistic mapping[7] is used widely in practice. The Logistic chaotic system has complex dynamical behaviors, it can be described as difference equation (5).

\[
\lambda_{i+1} = \mu \times \lambda_i \times (1 - \lambda_i)
\]

(5)

\( \lambda \in [0, 1] \), \( i = 0, 1, 2, \ldots \), \( \mu \) is in \([1, 4]\). Studies suggested that \( \mu \) is closer to 4, \( \lambda \) is closer to the average distribution between 0 and 1. Meanwhile the system is completely chaotic when \( \mu \) is 4.

Initial population is an important part in intelligent optimization algorithm, which influences the convergence rate and the final solution quality.[5-6][8-9] In this paper, Logistic chaotic mapping is used to initialize the population, which makes full use of the information solution space to improve the algorithm efficiency.
2.3 Population catastrophe

Catastrophe refers that large number of individuals has been dead because of the great environment changes in biological evolution process. Only a few excellent individuals could survive and reproduce.\textsuperscript{[10-11]}

In the algorithm processing, the algorithm could fall into local optimum when the optimal value did not change for successive generations. At this time, the population catastrophe is adopted to enable algorithm escape from local optimum. That is to say, only reserve the best individual and initialize the other individuals again.

2.4 Differential Evolution Algorithm

The differential evolution is an efficient algorithm for global optimization. The basic idea is as follow: Firstly to produce the intermediate populations by recombination with the current individual differences, and then to get the new population by the competition of parents and offspring.\textsuperscript{[12-14]}

There are mutation, crossover and selection operations in differential evolution processing. The mutation operator can maintain population diversity and improve the global searching ability. The crossover operator can accelerate convergence and improve the local search ability. The selection operator has historical memory and can preserve excellent individuals.

1) Mutation operation: The mutant vector is generated according to the formula.

\[ v_{i}^{t+1} = x_{i}^{t} + F \times (x_{r1}^{t} - x_{r2}^{t}) \] \hspace{1cm} (6)

\( x_{r1}, x_{r2}, x_{r3} \) are three different individuals randomly selected. \( i \) must be different from \( r_{1}, r_{2}, r_{3} \). \( F \) is scaling factor.

2) Cross operation: Crossover operation is used to increase population diversity, and it can be achieved with the formula.

\[ u_{i}^{t+1} = \begin{cases} v_{i,j}^{t}, & r\text{and}(0,1) \leq CR \text{ or } j = \text{rand}(1, n) \\ x_{i,j}^{t}, & r\text{and}(0,1) > CR \text{ or } j \neq \text{rand}(1, n) \end{cases} \] \hspace{1cm} (7)

Where, \( CR \) is in \([0,1]\).

3) Selection operation: The fitness value of trial vector is compared with the fitness value of target vector according to greedy criterion, and then determine whether the test vector to be admitted to the next generation. Selection can be achieved with the formula.

\[ x_{i}^{t+1} = \begin{cases} u_{i,j}^{t+1}, & f(u_{i,j}^{t}) < f(x_{i,j}^{t}) \\ x_{i,j}^{t}, & f(u_{i,j}^{t}) \geq f(x_{i,j}^{t}) \end{cases} \] \hspace{1cm} (8)

3. Improved bird swarm optimization algorithm

We proposed an improved chaotic bird optimization algorithm which integrates chaos mapping, population catastrophe and differential evolution (referred to as LCDE-BSA). The basic idea of this algorithm is that logistic chaotic mapping is used to initialize the population and make the initial distribution to the entire population of space to improve the solution efficiency, population catastrophe is adopted to escape from local optimum, differential evolution is used to enhance the population diversity and improve the optimizing efficiency by mutation operation, crossover operation and
selection operation on individuals except the current best individual. Improved algorithm process is as follows:

**Step1:** Initializing the parameters of algorithm.

**Step2:** Initializing the population with (5) and randomly select individuals as the global optimal.

**Step3:** Updating the population with equations (1), (2), (3) and (4), and meanwhile, to judge the current global optimal with threshold. If the optimal individuals are not updated after L generations, the current global optimal individual is retained and jump to Step4.

**Step4:** Generating temporary population with formulas (6) and (7), to evaluate the temporary population and calculate the individual fitness value in the population.

**Step5:** Comparing original population to temporary population with formula (8), and meanwhile, to reserve the optimal into the next iteration.

**Step6:** Recording and reserving the new fitness value and global optimal.

**Step7:** Repeating Step3-Step6.

4. Experimental Results

In order to verify the new algorithm effectiveness, Test experiments have been performed with six benchmark functions to compare the improved algorithm (referred to as LCDE-BSA) with classic algorithm (referred to as BSA).

The experiment parameters are shown in table 1. The benchmark functions are shown in table 2. The experimental results are shown in table 3 and table 4. The convergence curves of the two algorithms for solving problems are shown in figure 1.

| Algorithm   | Parameters                   |
|-------------|------------------------------|
| BSA         | \( C = S = 1.5; a1 = a2 = 1; PM \in [0.8,1]; FL \in [0.5,0.9]; FQ = 10 \) |
| LCDE-BSA    | \( C = S = 1.5; a1 = a2 = 1; PM \in [0.8,1]; FL \in [0.5,0.9]; FQ = 10, \mu = 4; CM = 2; LM = 5; CR = 0.1; F0 = 0.4 \) |
### Table 2. Benchmark functions

| Function | Function form | Bounds  | Optimum |
|-----------|---------------|---------|---------|
| Sphere    | \( f_1 = \sum_{i=1}^{n} x_i^2 \) | [-100, 100] | 0       |
| Griewank  | \( f_2 = \frac{1}{4000} \sum_{i=1}^{n} (x_i^2) - \frac{1}{\sqrt{n}} \prod_{i=1}^{n} \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1 \) | [-600, 600] | 0       |
| Rastrigin | \( f_3 = \sum_{i=1}^{n} [x_i^2 - 10 \cos(2\pi x_i) + 10] \) | [-5.12, 5.12] | 0       |
| Schwefel  | \( f_4(x) = \sum_{i=1}^{n} |x_i| + \prod_{i=1}^{n} |x_i| \) | [-10, 10] | 0       |
| Salomon   | \( f_5(x) = 1 - \cos(2\pi \sqrt{\sum_{i=1}^{n} x_i^2}) + 0.1 \sum_{i=1}^{n} x_i^2 \) | [-100, 100] | 0       |
| Rosenbrock| \( f_6(x) = \sum_{i=1}^{n} [100(x_{i+1} - x_i)^2 + (x_i - 1)^2] \) | [-100, 100] | 0       |

### Table 3. The results obtained by BSA and LCDE-BSA

| Function | Dimension | Algorithm | Best         | Mean         | Worst        |
|-----------|-----------|-----------|--------------|--------------|--------------|
| Sphere    | 20        | BSA       | 1.5695e-45   | 1.0973e-39   | 5.4348e-39   |
|           |           | LCDE-BSA  | 3.1386e-133  | 1.8430e-120  | 9.2148e-120  |
| Griewank  | 20        | BSA       | 0            | 0            | 0            |
|           |           | LCDE-BSA  | 0            | 0            | 0            |
| Rastrigin | 20        | BSA       | 0            | 0            | 0            |
|           |           | LCDE-BSA  | 0            | 0            | 0            |
| Schwefel  | 20        | BSA       | 1.153e-22    | 1.159e-19    | 5.6981-19    |
|           |           | LCDE-BSA  | 3.0463e-64   | 7.2343e-61   | 2.8814e-60   |
| Salomon   | 20        | BSA       | 1.6975e-16   | 7.0977e-14   | 2.9984e-13   |
|           |           | LCDE-BSA  | 3.415e-69    | 2.6556e-61   | 1.2807e-60   |
| Rosenbrock| 20        | BSA       | 5.7555e-20   | 1.4671e-05   | 7.3357e-05   |
|           |           | LCDE-BSA  | 0            | 0            | 0            |
Table 4. Mean iteration times of optimization

| Function  | Algorithm  | Global optimum | Mean iteration times |
|-----------|------------|----------------|---------------------|
| Sphere    | BSA        | 0              | 742                 |
|           | LCDE-BSA   | 0              | 125                 |
| Griewank  | BSA        | 0              | 46                  |
|           | LCDE-BSA   | 0              | 16                  |
| Rastrigin | BSA        | 0              | 52                  |
|           | LCDE-BSA   | 0              | 22                  |
| Schwefel  | BSA        | 0              | 1458                |
|           | LCDE-BSA   | 0              | 154                 |
| Salomon   | BSA        | 0              | 786                 |
|           | LCDE-BSA   | 0              | 125                 |
| Rosenbrock| BSA        | 0              | 395                 |
|           | LCDE-BSA   | 0              | 30                  |

Figure 1-1 Convergence curves of Sphere  
Figure 1-2 Convergence curves of Griewank

Figure 1-3 Convergence curves of Rastrigin  
Figure 1-4 Convergence curves of Schewel
The results shown in Table 3 indicate that BSA and LCDE-BSA are the same optimal performance on Griewank and Rastrigrin. However, LCDE-BSA is better optimal performance than RSA in Sphere, Schwefel, Salomon and Rosenbrock. That is to say, the improvement for RSA is effective.

The results shown in Table 4 indicate that LCDE-BSA is much better than BSA in the optimization process. The statistical results show that the mean iteration times of LCDE-BSA is much less than that of BSA.

Moreover, Convergence curves of the two algorithms for six benchmark functions show that the convergence speed of LCDE-BSA is faster than that of BSA, however, LCDE-BSA increases the time overhead.

5. Conclusion

Since bird swarm optimization algorithm has been proposed, many scholars have done research on it. Studies suggested that BSA has good optimization precision and good global search ability, but its population diversity is insufficient. It easily falls into local extrema and appeared premature convergence for high-dimensional functions. Aiming at the defects, we proposed an improved chaotic bird-swarm optimization algorithm. The new way is that logistic chaotic mapping is used to initialize the population and make the initial distribution to the entire population of space to improve the efficiency and quality of solution, population catastrophe is adopted to escape from local optimum, differential evolution is used to enhance the population diversity and improve the optimizing efficiency by mutation operation, crossover operation and selection operation on individuals except the current best individual. Experiments results on six criteria test functions indicate that improved algorithm (LCDE-BSA) has better performance in global search ability and convergence speed than classical algorithm (BSA).

Acknowledgments

This work was supported by the Guizhou Province Natural Science Foundation of China (grant number [2017]1047) and the Foundation of the Department of Science and Technology of Guizhou Province (grant number [2015]7637). The authors also gratefully acknowledge the useful suggestions of the reviewers and editors, which have processed and improved the presentation.
References

[1] Meng X B, Gao X Z, Lu L H, et al. A new bio-inspired optimization algorithm: bird swarm algorithm[J], Journal of Experimental & Theoretical Artificial Intelligence, 2015.

[2] Xiaolong Liu, Qianzhao Ning, Ping Cheng, et al. Bird Swarm Algorithm Based on Levy Flight[J], Computer Measurement & Control, 2016, 24(12): 194-197.

[3] Dongwen Cui, Bo Jin. Improved of Bird Swarm Algorithm and Application to Reservoir Optimal Operation[J]. Journal of China Three Gorges University (Natural Sciences), 2016, 38(6): 7-14.

[4] Ceng Zeng, Chunhua Peng, Kui Wang, et al. Multi-objective operation optimization of micro grid based on bird swarm algorithm[J], Power System Protection and Control, 2016, 44(13): 117-122.

[5] Shengsheng Wang, Juanjuan Yang, Sheng Chai. Artificial Bee Colony Algorithm with Chaotic Catfish Effect and Its Application[J], Acta Electronica Sinica, 2014, 42(9): 1731-1737.

[6] Liangdong Qu, Dengxu He. Novel artificial fish-school algorithm based on chaos search[J], Computer Engineering and Applications, 2010, 46(22): 40-42.

[7] Sabat S L, Kumar K S, Udgata S K. Differential evolution and swarm intelligence techniques for analog circuit synthesis[C]/2009 World Congress on Nature & Biologically Inspired Computing, 2009: 468-474.

[8] Huan Chen, Yongquan Zhou, Guangwei Zhao. Multi-population invasive weed optimization algorithm based on chaotic sequence[J], Journal of Computer Applications, 2012, 32(7): 1958-1961.

[9] Jianxia Yin, Hongyun Meng. Artificial bee colony algorithm with chaotic differential evolution search[J], Computer Engineering and Applications, 2011, 47(29): 27-30.

[10] Meiyi Liao, Yongjun Zhang. Study on the Effect of Cataclysm Operator on Genetic Algorithm[J], Computer Engineering and Applications, 2005, 41(13): 54-56.

[11] Yang Fu, Yiliu Jang, Zhenkun Li. Optimal allocation of distributed generation for microgrid based on hybrid quantum genetic algorithm[J], Power System Protection and Control, 2013, 41(24): 50-57.

[12] Xin Pan, Xiaozhi Gao. A fruit fly optimization algorithm based on differential evolution[J], Microcomputer & its Applications, 2015, 34(1): 23-25.

[13] Yuncheng Li, Jie Fang, Guobiao Cai. Hybrid optimization algorithm based on differential evolution and particle swarm optimization[J], Computer Engineering and Applications, 2009, 30(12): 2963-2965.

[14] Yuhong Duan, Yuelin Gao. A Particle Swarm Optimization Algorithm Based on Differential Evolution[J], Computer Simulation, 2009, 26(6): 212-215.

[15] Yan Yang, Ruqing Chen, Jinshou Yu. Study on differential evolution-particle swarm optimization based hybrid optimization algorithm and its application[J], Computer Engineering and Applications, 2010, 46(25): 238-241.