

Research on Key Problems of Multi-objective Evolution Algorithm Based on Cloud Model and Chaotic Particle Swarm

Yang Li, Liping Wang

Pingxiang University, Pingxiang, Jiangxi, China, 337000

Keywords: Cloud model; Chaos theory; Particle swarm optimization (PSO); multi-objective optimization

Abstract: Particle swarm optimization (PSO) is one of the most widely used intelligent algorithms, which hinders its further promotion because of its inherent shortcomings. This paper analyses the iterative characteristics and rules of particle swarm optimization, summarizes the causes and solutions of defects, and improves them one by one. From early to late stage, it improves particle swarm optimization algorithm and improves the performance of algorithm. The experimental results show that the comprehensive optimization model proposed in this paper has good applicability. The chaos particle swarm optimization algorithm based on cloud model is feasible in solving the multi-objective optimization problem. The results show that the comprehensive model algorithm proposed in this paper effectively solves the shortcomings of PSO algorithm, such as premature and easy to fall into local optimum, and the algorithm is effective and feasible for solving multi-objective reactive power optimization problems.

1. Introduction

In the real world, many engineering applications are actually multi-objective optimization problems, and there may be conflicts between them. Therefore, many practical engineering optimization problems usually have multiple conflicting objective functions [1]. Multi-objective optimization is the tradeoff between multiple targets and a set of Pareto optimal solutions is obtained. Because of the complexity of the multi-objective optimization problem, the traditional methods in operational research are difficult to solve independently [2, 3]. Evolutionary algorithm is a stochastic optimization method to simulate the natural evolutionary process, with highly parallel mechanism, the multiple objectives are optimized at the same time, the operation can generate multiple Pareto optimal solutions at a time, and evolutionary algorithm is the most suitable for solving multi-objective optimization problems, and from the birth of the multi-objective evolutionary algorithm [4]. The chaos particle swarm optimization algorithm based on cloud model is proved to be an effective way to solve multi-objective optimization problems, which effectively solves the shortcoming of particle swarm optimization algorithm, which is premature and easy to fall into local optimum.

2. Chaos Particle Swarm Optimization Algorithm

2.1 The idea of particle swarm optimization

The particle swarm optimization (PSO) algorithm is a new global optimization evolutionary algorithm proposed by Eberhart. The algorithm is derived from the simulation of bird predator behaviour. The particle swarm optimization (PSO) algorithm initializes a group of random particles first and then finds the optimal solution by iteration [5, 6]. One is the optimal solution found by the particle itself, that is, the individual extreme value pBEST. The other is the optimal solution that is now found by the whole population, which is called the global extreme value GDEF. At the same time, the particles update their speed and position according to the following 2 formulas:

$$V = \omega \times V + C_1 \times rand() \times (pBest - Present) + C_2 \times rand() \times (gBest - Present) \quad (1)$$

$$Present = Present + V \quad (2)$$

V is the velocity of the particle, and the Present is the current position of the particle. $rand()$ is a random number between (0,1), and C_1 and C_2 are called learning factors. According to the above formula, the particle eventually flies to the location of the optimal solution in the solution space, and the search process is over.

2.2 Chaos thought and algorithm flow

Chaos refers to the random motion state that is obtained by the equation of motion. The typical chaotic system Logistic mapping equation is as follows:

$$x_{k+1} = \mu x_k (1 - x_k), 0 \leq x_0 \leq 1 \quad (3)$$

Among them, μ is a control parameter, and x_k is a variable, $k = 0, 1, 2, \dots$.

The basic idea of chaos optimization is firstly generated a set of optimization variables with the same number of chaotic variables, in a similar way to chaos into the carrier so as to make the chaotic state variables, while the range of amplification of chaotic motion to traverse the range of optimization variables, and then direct search using chaotic variables [7, 8]. With the characteristics of ergodicity randomness, chaos is combined with PSO algorithm to form a hybrid algorithm, which can make local search more efficient [9]. The specific algorithm flow is as follows:

Step 1: chaos initialization. Random generation of a set of patterns selected particles, chaos initializes the position and speed of the particle. The adaptive value of the initial particle is calculated by the fitness function, and the individual extremum and the global extremum of the initial particle are recorded.

Step 2: update the Particle Swarm. The position and velocity of the particles are updated respectively, and the fitness of the particles is calculated by the fitness function.

Step 3: Use step 2 particle location as chaotic variable, update the chaotic sequence, generate the location of new particles after chaos, and calculate fitness of particles through fitness function.

Step 4: compare the better solutions of two adaptive values. The local optimal solution of the particle is updated and the global optimal solution is updated.

Step 5: if the maximum number of iterations is reached, the global optimal solution $gBest$ is returned; otherwise, jump to step 2.

In evolutionary algorithm, in order to maintain the diversity of evolutionary population, to simulate the biological phenomena in nature using niche technology, the niche technology based on sharing mechanism is the most representative, such as MOGA, NPGA, NSGA algorithm used in diversity sharing strategy. The main idea of this strategy is to categorize all the non-inferiority individuals according to the non-inferiority and dominance of individuals, and assign all the non-inferior individuals into one class, and give the same fitness to the non-inferior solutions [10]. Then we calculate the niche number of every Pareto optimal solution according the following formulas, and the fitness is divided by the original fitness of the individual by the number of small habitat.

3. Adaptive Chaotic Particle Swarm Optimization (PSO) Algorithm

The basic PSO algorithm initialization is random, it will have certain blindness, random initialization although we can ensure the uniform distribution of the initial groups, but can not guarantee the quality of each particle, may cause some particles away from the optimal solution,

thus affecting the convergence speed of the algorithm [11]. In view of the shortcomings of the above basic PSO algorithm, this paper proposes an adaptive chaos particle swarm optimization algorithm based on chaos optimization:

(1) The initial particle swarm is initialized by chaos optimization and ergodicity. A series of initial solution groups are generated, and the initial population is selected from it. This effectively improves the diversity of particle swarm algorithm and ergodicity of particle search.

(2) The inertia weight in particle swarm optimization algorithm is adaptively adjusted according to the fitness values of each particle, so as to enhance the ability of "global rough detection" and "local fine excavation" and also improve the convergence speed and accuracy of the optimization algorithm [12].

(3) According to the size of the fitness value, we can determine the advantages and disadvantages of the current particles, and perform chaos operations on the selected particles, so as to help the lazy particles jump out of the local solution area and search for the global optimal solution quickly.

The basic idea based on the ACPSO algorithm is to initialize the chaotic algorithm first and optimize the initial particle group, so as to ensure the diversity of the population. The direction of flight is guided by particle swarm optimization. In order to speed up convergence, we choose the particles with high fitness to operate in chaos, so that particles can jump out of the local extreme region and search for the global optimal solution quickly. At the same time, it adaptively adjusts its inertia weight coefficient to improve the global and local search capability according to the target function value. The algorithm flow is shown in Figure 1:

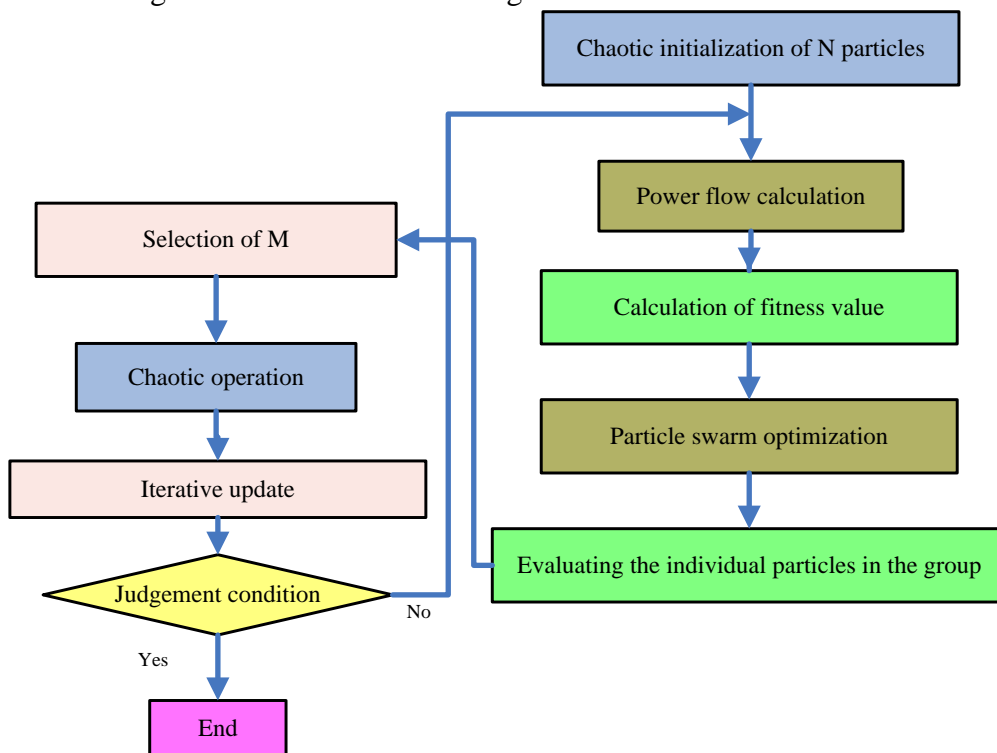


Figure 1. ACPSO algorithm flowchart

According to the operation process, we draw the average optimal fitness curve graph of the multi-objective case under the basic particle swarm optimization algorithm and the chaos particle swarm optimization algorithm, namely the convergence contrast diagram, as shown in Figure 3. It can be seen that when PSO algorithm solves the scheduling problem of the project, it achieves convergence effect through 15 evolutionary operations, while the chaotic PSO can achieve convergence only by 5 evolutionary operations. It can be seen that the ACPSO algorithm has higher convergence and less variance, and the results are relatively stable.

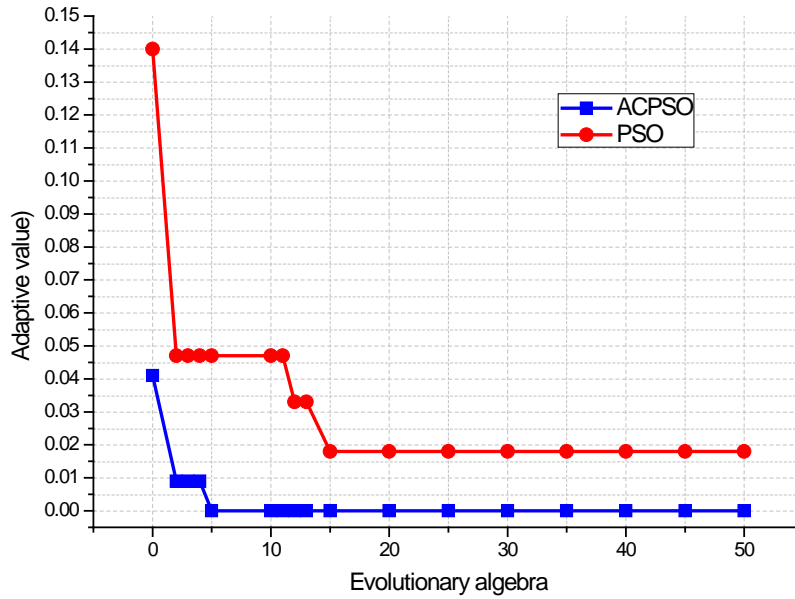


Figure 2. Comparison of operation results

4. Summary

The experimental results show that the comprehensive optimization model proposed in this paper has good applicability. The chaos particle swarm optimization algorithm based on cloud model is feasible in solving the multi-objective optimization problem. And its operation results prove that the basic PSO algorithm is easy to fall into local optimum, and it is easy to miss the optimal value. The ergodicity and fast convergence of chaotic search make it possible to avoid falling into local minimum. Therefore, the cloud model chaotic particle swarm optimization (PSO) achieves more ideal and scientific results than the basic particle swarm optimization algorithm in solving the problem.

Acknowledgement

This work was supported by Pingxiang Science and Technology Plan Project (2017GY004), and 2017 Key Projects for R&D in Jiangxi Province (20171BBE50049)

References

- [1] Wang N, Wei B, Wang J D, et al. Cloud Service Selection Based on Chaotic Multi-objective Particle Swarm Optimization Algorithm[J]. Computer Engineering, 2014.
- [2] He X, Pang X, Zhu D R, et al. Multi-objective reactive power optimization based on chaos particle swarm optimization algorithm[C]// International Symposium on Instrumentation and Measurement, Sensor Network and Automation. IEEE, 2014:1014-1017.
- [3] Xuan W U, Wang J. Reactive Power Optimization of Mining Area Power Grid Based on Multi-Objective Chaotic Particle Swarm Algorithm[J]. Low Voltage Apparatus, 2011.
- [4] Pei S Y, Zhou Y Q. A multi-objective particle swarm optimization algorithm based on the chaotic mutation[J]. Journal of Shandong University, 2010, 45(7):18-23.
- [5] He Y, Xu Q, Yang S, et al. Reservoir flood control operation based on chaotic particle swarm optimization algorithm[J]. Applied Mathematical Modelling, 2014, 38(17–18):4480-4492.
- [6] Pei S, Zhou Y. A multi-objective particle swarm optimization algorithm based on chaotic mutation[J]. 2010, 45(7).

- [7] Zhou L H, Liu Y H, Chen G L. A Feature Selection Algorithm to Intrusion Detection Based on Cloud Model and Multi-Objective Particle Swarm Optimization[C]// Fourth International Symposium on Computational Intelligence and Design. IEEE Computer Society, 2011:182-185.
- [8] Wu J, Xiong Y. A cloud model theory based adaptive particle swarm optimization algorithm for complementary power generation optimization of wind-energy, hydro-energy and natural gas[J]. Zhongguo Dianji Gongcheng Xuebao/proceedings of the Chinese Society of Electrical Engineering, 2014, 34:17-24.
- [9] Bo W, Wang J, Zhang H, et al. Dynamic service selection based on chaotic mutation multi-objective particle swarm algorithm[C]// International Conference on Computer Science and Network Technology. IEEE, 2014:113-117.
- [10] Gong W, Yan X, Liu X, et al. Parameter extraction of different fuel cell models with transferred adaptive differential evolution[J]. Energy, 2015, 86:139-151.
- [11] Zhao F, Jiang X, Zhang C, et al. A chemotaxis-enhanced bacterial foraging algorithm and its application in job shop scheduling problem[J]. International Journal of Computer Integrated Manufacturing, 2015, 28(10):1106-1121.
- [12] Xu S, Wang Y, Lu P. Improved imperialist competitive algorithm with mutation operator for continuous optimization problems[J]. Neural Computing & Applications, 2017, 28(7):1667-1682.