A Filtering Mechanism Based Optimization for Particle Swarm Optimization Algorithm

Ji Weidong and Zhu Songyu

Harbin Normal University
kingjwd@126.com, 1005854757@qq.com

Abstract

Particle swarm optimization algorithm is one of the most common algorithms for optimization. Because of the code is simple convenient operation, and good robustness advantages of attention. Particle swarm optimization algorithm is popularly, however, there is operation precision which is not high, with the poor population diversity .Many scholars have made this algorithm improved. This paper proposes a new particle swarm algorithm improvement program, introduced the particle swarm optimization algorithm to update them. With the introduction of filtering mechanism, it reduced the number of particles, and made a disturbance on the global extreme, to a more specific tracking for specific issues. Improved cut particle swarm optimization algorithm based on filtering mechanism (ELPSO) makes its operation speed and more accurate.

Keywords: optimization, particle swarm optimization algorithm, filtering mechanism

1.Introduction

Swarm intelligence algorithm gathers the wisdom of nature. They work together for the whole task and every single individual only play one part of the job. Each unit only needs to do the job around it but others. The whole group will finish the work perfectly well by a simple interaction between each individual and get the optimal solution of the problem.

Particle swarm optimization algorithm of swarm intelligence is an algorithm to simulate birds foraging foods [1].It puts forward to solve many complex problems and brings convenience for some optimization of engineering problems. In recent years, many scholars at home and abroad improve the basic particle swarm optimization algorithm in light of some insufficient [2-9]. Among them, the Chinese scholars Wang Hu put forward multi-objective particle swarm optimization algorithm based on the Pareto entropy[10]; Xin-Min Tao put forward mutation particle swarm optimization algorithm by multi-scale [11]; Qing-Jian Ni put forward a particle swarm optimization algorithm of the cluster structure based on variable dynamic probability[12]; and Xin-Chao Zhao put forward a disturbance of multi-phase particle swarm optimization algorithm based on non-uniform variation [13], *etc*. They make the particle swarm algorithm improved in searching [14,15].

In this paper, the particle swarm optimization algorithm based on the filtering mechanism, is on the basis of the basic particle swarm optimization algorithm. The filter factor is introduced into it. The algorithm cut the number of particles at a particular time and it add a disturbance on the global extreme. It reduce the number of particles in the particle swarm algorithm based on adaptive fitness values and then use the most elite particle to approach the optimal value.

2. Improved Particle Swarm Optimization Algorithm

2.1 General Situation of the PSO

The particle swarm optimization algorithm (PSO) is put forward by Dr. Kennedy and Dr. Eberhart in 1995. This idea is from the simulation of birds feeding on behavior. When the birds are searching for foods, each bird has its own position and speed and they are regularly in queue. According to the whole swarm conditions, they compared to adjust their own position and speed in the optimal state. The basic particle swarm algorithm steps are as follows: First, we initialize the particle's position, speed and calculate the fitness value of particles, and then record the particle's position. After several iterative cycles, we update the particle's position and speed and record the data at the same time. When all the iteration is completed, we record the optimal position of the particles. We use the falling formula:

The update formula of particle's velocity:

$$v_{ij}(t+1) = wv_{ij}(t) + c_1 r_1 \left[\left(p_{ij} \right)_j (t) - x_{ij}(t) \right] + c_2 r_2 \left[p_{gj}(t) - x_{ij}(t) \right]$$

Eq. 1）

The update formula of the particle's location:
$$x_i(t+1) = x_i(t) + v_i(t+1)$$
$$1 \le i \le n, 1 \le j \le d$$

Eq. 2）

Among them, the 't' means time; 'c_1, c_2' are the accelerating constants，it is usually an integer; 'W' means inertia weight value; 'r_1, r_2' are random numbers between [0, 1]; 'x_{ij}' is the particle's location on the 'j' dimension; 'v_{ij}' is the particle's velocity on the 'j' dimension; 'p_i' is the individual extreme; 'p_g' is the global extreme.

### 2.2 The Improvement Direction of the PSO

The particle swarm optimization algorithm provides great convenience to solve practical optimization problem for scholars. Its high search speed and high efficiency has been widely applied and the code is very simple. The optimal solution of the problem can be calculated by several short lines. It also has good characteristic of robustness. But the particle swarm optimization algorithm mentioned in the literature is not good at dealing with the problem of discrete classes, it is easy to fall into local optimum.The global optimal particle swarm optimization algorithm's convergence is fast, but the population diversity is poorer. So the global optimal particle swarm optimization algorithm easily plunged into local optimum. The Local particle swarm optimization algorithm is better at species diversity. It can search the optimal solution much better but the rate of convergence is poor. Although many improved algorithm is hard to make a trade-off between the convergence speed and local optimal, but has never very satisfactory results. The purpose of this paper is to improve the standard particle swarm optimization algorithm and introduce the filter factor. It reduces the particle numbers of the population according to the fitness value and adds the Cauchy disturbance on global extreme. This algorithm speeds up the convergence rate and improves the search accuracy.

2.3 Reduction of the Particle Swarm Optimization Algorithm

Reduction of the particle swarm optimization algorithm based on filtering mechanism (eliminate PSO, ELPSO), is a kind of improved algorithm based on standard particle swarm optimization (PSO). It has simple operation and it is easy to implement. Its core idea is to enrichment the particle numbers. After initial the numbers and positions of

particles, we group the iterations and calculate the fitness value of each particle. When a certain number of iterations are done, we compare each particle's fitness value with the average value of all particles fitness. And then we eliminate parts of initial particle by filtering mechanism. The remaining particles called the 'filter particles' can do iterative operation again. The particles who are eliminated is called 'out particles', it will no longer participate in the following iterative operation. For avoid falling into local optimum we add a disturbance on the global extreme value so that the particles can search the position near the optimal value constantly. Repeat it until the iteration is done, the particle swarm is approaching to the optimal position. This algorithm improves the searching precision and guarantees the diversity of population. It is easy to calculate and convergence after cutting the number of particles gradually.

Algorithm flow chart is shown in figure 1:
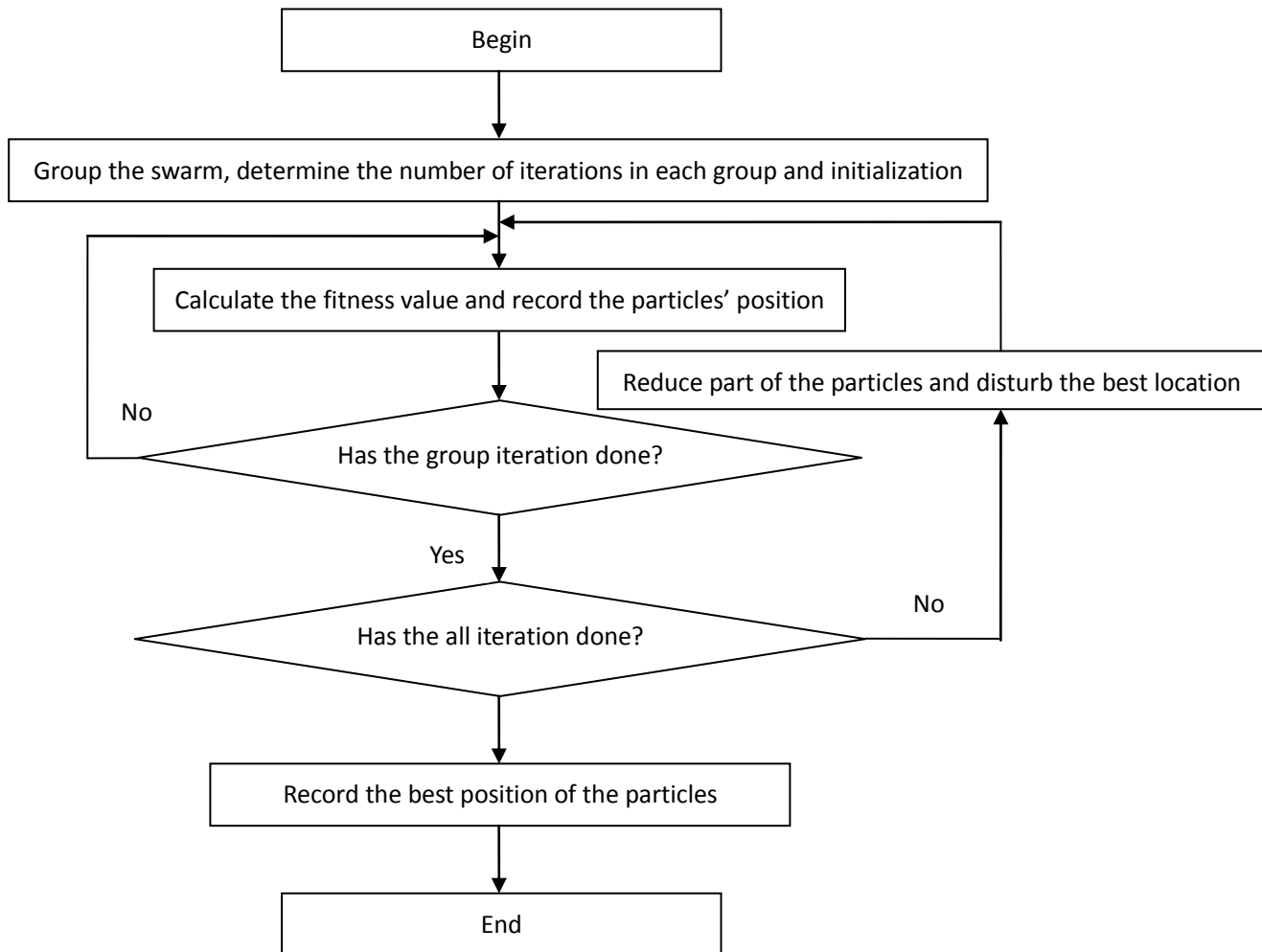
Figure 1. The Algorithm Flow Chart of the ELPSO

2.4 The Iteration Filtering Factor and the Cauchy Disturbance

Here is the first to introduce the concept of 'iterative filtering factor' (d). Since the initial state of the particle number is numerous, the optimal position of each particle is more dispersed and be in discrete state. In this paper, we use the adaptive filtering mechanism. And after a certain number of iterations, we eliminate some particles whose average fitness values are worse than the average fitness values of the entire population.

Then only the 'filter particles' are remaining in the swarm. With the increasing of the iterations, the number of 'out particles' is increase as well. The elite 'filter particles' gradually approach to the best position and solve the problem.

At the beginning of the iteration, the particles search more spread out. So we need to pass through a large number of iterations to make poor fitness values of particles go out of the swarm. This paper introduces to the strategy of grouping iterations and generally divides the iterative process into three groups with the number of iterations for 600 times. In addition the number of the first iterations is more than half the number of all iterations. We general directly select one half of the total number of the iterations, such as type (3). The number of the iterations after that reduces by the arithmetic progression. The 'iterative filter factor' value is determined by the number of iterations and group number. According to the formula (4) we can deduce the calculation formula of the 'iterative filter factor' such as type (5). When it is not an integer, please automatic integer to the nearest.

$$N_0 \geq \frac{N_{it}}{2} \qquad\qquad \text{Eq. 3）}$$

$$N_{it} = nN_0 + \frac{1}{2}n(n-1)d \qquad\qquad \text{Eq. 4）}$$

$$d = \left\langle \frac{2(N_{it} - nN_0)}{n(n-1)} \right\rangle \qquad\qquad \text{Eq. 5）}$$

Among them, 'N_{it}' means the total number of the iterations; 'N_0' is the first set of iterations; 'n' means the number of groups; the d means the iterative filter factor.

After the first set of iteration is complete, calculate the average fitness value of all particles which is called 'p_{avg}'. When the individual fitness values are worse than 'p_{avg}', filter out them. We let the rest of the particles who is the 'filter particles' to continue the second iteration. In the iteration of the second group, due to the reduction of the number of particles, we save much time. In this paper, the 'ELPSO' algorithm adds a Cauchy disturbance at the beginning of the second group on the global optimal value, as shown in formula (6) and calculates the fitness value p'_g. Compared it with the fitness value of p_g and preserve the one whose fitness value is superior. Also do the same Cauchy disturbance in the third group of iterative, which safeguards search precision of the particle swarm, avoids precocious into local optimum and affects the outcome.

This article select the Cauchy distribution function as a disturbance, defined as shown in formula (7). When time is greater than zero, the Cauchy distribution function is for the type (8). Because the Cauchy distribution function is concentrated near the origin [6], with higher probability of the two wings [5], its random variable range is very large. Therefore, after adding the Cauchy disturbance the modifier of p_g is bigger also. What's more, joining the Cauchy disturbance also can maintain the diversity of population, help particles jump out of local optimal position at the same time, avoid the early mature and fall into the local optimum risk to a certain extent, which makes the algorithm has better performance in search.

$$p'_g = p_g + Cauchyrnd（\ ） \qquad\qquad \text{Eq. 6）}$$

$$f(x) = \frac{1}{\pi} \cdot \frac{t}{t^2 + x^2}, -\infty < x < \infty \qquad\qquad \text{Eq. 7）}$$

$$F(x) = \frac{1}{2} + \frac{1}{\pi} \cdot \arctan\left(\frac{1}{t}\right) \qquad \text{Eq. 8）}$$

In the formula, p_g is the global optimal particle position; p'_g is the particles position after adding disturbance; $Cauchyrnd(\ )$ is the creation of Cauchy random number; $f(x)$ and $F(x)$ are the definition of the Cauchy function.

3. The Experimental Contrast

3.1 The Experimental Setup

This paper chooses the most commonly used data to compare the experiment. The initial particle number is 60 and the total number of iterations is 600 times. The iterative operation is divided into three groups. Through the calculation iterative filter factor for d = 100.Therefore, the first group of iterations has 300 times, the second group for 200 times, the third group for 100 times.

Experimental function selection is as shown in table 1. Sphere is a relatively simple unimodal function and its minimum value is 0. Griewank is a non-convex function and its minimum value is 0. This can inspection algorithm the ability of searching optimization. Rosenbrock has a lot of local minimum value and it is suitable for test whether algorithm trapped in local optimal. Rastrigin is a multimodal one and it has a large number of local minima, with global minimum value 0. A general algorithm is difficult to calculate the optimal solution of the Rastrigin and it is suitable for inspecting the search ability of the algorithm.

Table 1. The Test Function

Function	The specific content
Sphere	$f(x) = \sum_{i=1}^{n} x_i^2$
Griewank	$f(x) = \frac{1}{4000} \sum_{i=1}^{n} x_i^2 - \prod_{i=1}^{n} \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$
Rosenbrock	$f(x) = \sum_{i=1}^{n} \left[100\left(x_{i+1} - x_i^2\right)^2 + \left(x_i - 1\right)^2 \right]$
Rastrigin	$f(x) = \sum_{i=1}^{n} \left[x_i^2 - 10\cos\left(2\pi x_i\right) + 10 \right]$

3.2 The Experimental Results

The experimental results are shown in table 2-table 5.

Table 2. SPHERE

Sphere	time（s）	The best	The worst	The average	The standard
PSO	9.903606	0.00058491	0.0138	0.0061	0.0051
ELPSO	5.842349	0.00017513	0.0071	0.002	0.0021

Table 3.   GRIEWANK

Greiwank	time（s）	The best	The worst	The average	The standard
PSO	16.094375	0.00141210	0.0303	0.0113	0.0079
ELPSO	9.081508	0.00003633	0.0132	0.0049	0.0049

Table 4.   ROSENBROCK

Rosenbrock	time（s）	The best	The worst	The average	The standard
PSO	4.908909	0.01981821	13.9284	3.7540	4.3857
ELPSO	3.259336	0.00004447	4.7069	1.4465	1.4402

Table 5.   RASTRIGIN

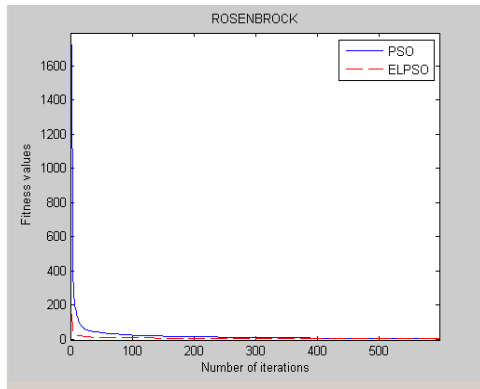| Rastrigin	time（s）	The best	The worst	The average	The standard
PSO	17.508821	84.1231315	596.1525	189.6407	166.8815
ELPSO	12.681422	66.8277432	202.8289	86.4044	107.7019

The simulation images are shown in figure 2-figure 5.
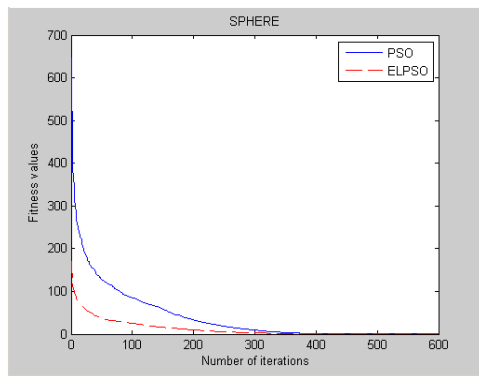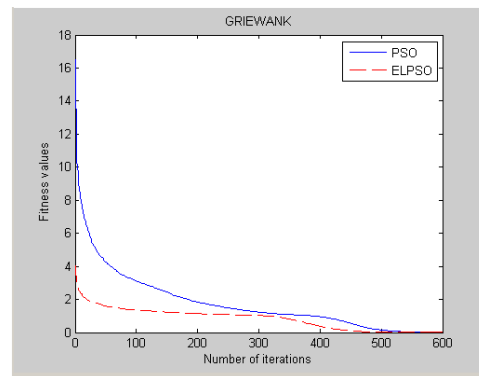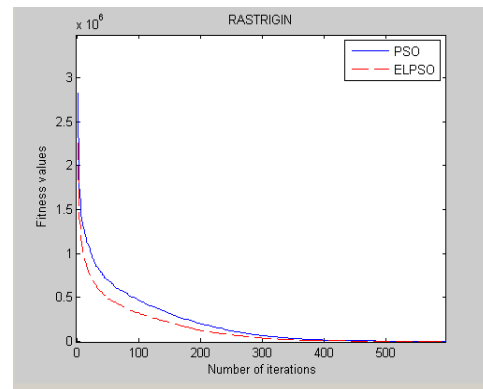

Figure 1. SPHERE　　　　　　**Figure 2. GRIEWANK**

Figure 3. ROSENBROCK   Figure 4. RASTRIGIN

3.3 The Results Analysis

This experiment compared the reduction of the particle swarm algorithm (PSO) with standard particle swarm optimization algorithm by three aspects: time, fitness value and standard deviation. Firstly, since the number of standard particle swarm is constant and ELPSO cut the number of particle after each group is over. So in terms of time, reduction of the particle swarm algorithm takes less time and searches faster in speed than the standard particle swarm algorithm. Obviously the experimental data also prove it well.

Secondly, the fitness values of functions reflect the accuracy of the algorithm. Because the standard particle swarm algorithm is easy to fall into the local optimal but the reduction of the particle swarm algorithm adds the Cauchy disturbance on the global optimal value which is conducive to jump out of local optimal. As well the experiments show that the fitness values of ELPSO are better than the standard particle swarm optimization algorithm, the search precision of ELPSO is higher. Finally, the standard deviations of functions can reflect how discrete the value is and is good at testing the robustness of the algorithm. Searching of PSO is randomness, searched data is discrete, the standard deviation is larger, and the algorithm is unstable. However, ELPSO use filtering mechanism to eliminate the discrete data to approach the optimal gradually. The standard deviation is small by the experiment and the algorithm reduces the dispersion that proves ELPSO has good robustness.

4. The Conclusion

This paper proposes a new particle swarm algorithm. The numbers of iterations are grouped in it and it introduces the filtering mechanism. It adds the Cauchy disturbance on the global optimal particle. The main idea of the algorithm is that for grouping the iterative process, according to the fitness value better or not, regularly to reduce the number of particle population. The individual fitness which is worse will be filter out, leaving fitness strong particles. And the optimal particles of each group add the Cauchy disturbance to search the location of the surrounding again, thus it improves the efficiency and superiority of the algorithm. The number of particles is remaining the same from beginning to end in PSO. It is prone to stagnation of the search and trap in local optimal problem of premature convergence. However, the reduction of the particle swarm optimization algorithm based on filtering mechanism search again and again through the remainder of the elite 'filter particle' closing to the optimal value. ELPSO slightly reduces the risk of trapping in local optimal and improves the performance of algorithm to a certain extent.

Acknowledgements

References

[1] J. Kennedy and C. Eberhartr, "Particle swarm algorithm", Proceedings of the IEEE International Conference, **(1995)**; Neural, Newyorks.
[2] D. E. Goldberg, "Genetic algorithms in Search", Optimization, and Machine Learning, Reading MA: Addison-Wesley, **(2005)**.
[3] J. J. Lake, A. E. Duwel and R. N. Candler, "Particle swarm optimization for design of slotted MEMS resonators with low thermoelastic dissipation", Microelectromechanical systems, IEEE, **(2014)**.
[4] Y. Morsly, N. Aowf, M. S. Djouadi and M. Richardson, "Optimized hyperspectral band selection using particle swarm optimization", Selected topics in applied earth observations and remote sensing, IEEE, **(2014)**.
[5] J. Wei and Y. Wang, "Multi-objective fuzzy particle swarm optimization based on elite archiving and its convergence", Systems Engineering and Electronics, IEEE, **(2012)**.
[6] R. J. Zha, D. P. Zhang, C. H. Nie and B. W. Xu, "Multi-objective fuzzy particle swarm optimization based on elite archiving and its convergence", Chinese Journal of Computers, vol. 33, no. 10, **(2010)**.
[7] T. Wen, G. J. Sheng, Q. Guo and Y. Q. Li, "Web Service Composition Based on Modified Particle Swarm Optimization", Chinese journal of computers, vol. 5, no. 36, **(2013)**.
[8] Y. X. Shen, G. Y. Wang and C. H. Zeng, "Correlative Particle Swarm Optimization Model", Journal of Software, vol. 4, no. 22, **(2011)**.
[9] X. Chen, Q. Gu, Z. Y. Wang and D. X. Chen, "Framework of Particle Swarm Optimization Based Pairwise Testing", Journal of Software, vol. 12, no. 22, **(2011)**.
[10] H. Wang, G. G. Yen and X. Zhang, "Multiobjective Particle Swarm Optimization Based on Pareto Entropy", Journal of Software, vol. 5, no. 25, **(2014)**.
[11] X. M. Tao, F. R. Liu, Y. Liu, Z. J. Tong, "Multi-Scale Cooperative Mutation Particle Swarm Optimization Algorithm", Journal of Software, vol. 7, no. 23, **(2012)**.
[12] Q. J. Ni, Z. Z. Zhang, M. C. Wang and X. H. Chen, "Dynamic Probabilistic Particle Swarm Optimization Based on Varying Multi-Cluster Structure", Journal of Software, vol. 2, no. 20, **(2010)**.
[13] Z. X. Chao, L. G. Li, L. H. Qiu and Z. G. Shuai, "Particle Swarm Optimization Algorithm Based on Non-Uniform Mutation and Multiple Stages Perturbation", Chinese journal of computers, vol. 9, no. 37, **(2014)**.
[14] L. Zhang, "Improved Quantum-Behaved Particle Swarm Optimization for Power Economic Dispatch, Computer & Digital Engineering", vol. 8, no. 41, **(2013)**.
[15] X. Z. Zhang, H. M. Ma and X. P. Qian, "IIR digital filter design based on improved PSO algorithm", Computer Engineering and Design, vol. 8, no. 32, **(2011)**.

Authors
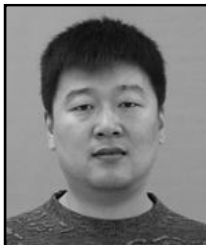
Ji Weidong, He received his M.Sc. in Control Theory and Control Engineering (2004), and PhD in Mechanical Design and Theory (2013) from Northeast Forestry University. He is a associate Professor of Computer Science and Information Engineering. Since 2013 he is a Master Instructor. His current research interests include different aspects of Neural Network and Swarm Intelligence.

Zhu Songyu, She is a graduate student of Harbin normal university. Her research direction is the swarm intelligence.