Improvement on PSO with Dimension Update and Mutation

Hui Xu
School of Computer science and Technology
School of Mineral Resource and Earth Science
China University of Mining & Technology, Xuzhou, China
xuhui@cumt.edu.cn

Yongguo Yang
School of Mineral Resource and Earth Science
China University of Mining &Technology, Xuzhou, China
ygyang88@cumt.edu.cn

Lei Mao
School of Computer science and Technology
China University of Mining & Technology, Xuzhou, China
ml@cumt.edu.cn

Hongxia Xie
School of Computer science and Technology
China University of Mining & Technology, Xuzhou, China
xiehx@cumt.edu.cn

***Abstract*—Sub-dimension particle swarm optimization(s-dPSO) is proposed based on basic particle swarm optimization (bPSO). Each dimension of particle in s-dPSO is updated in turn. The dimensions with poor diversity would be mutated that is initialized again to improve the diversity of population and get global optimal solution when the algorithm is in the local convergence. Most Benchmark function get good result with s-dPSO which ability of optimization is better than bPSO.**

***Index Terms*—s-dPSO, dimension, diversity, optimizatio**

I. INTRODUCTION

Particle Swarm Optimization (PSO) was proposed based on bird foraging behavior by Dr Eberhart and Kennedy In 1995. PSO which attracts extensive attention from academics in recent years is a highly efficient search algorithms due to the simply concept, easy implementation, fast convergence and less parameters setting [1].

PSO is a heuristic and global optimal algorithm. The individuals in the bird flock adjust its next search direction and size according to the individual in the optimal position in the groups and the optimum position of its own history when the entire groups search for a target. The model of groups foraging behavior was designed to solve function optimization problems through the method of experiments and progressive correction step by step. PSO is easy to trap into the local convergence for the complex function optimization and cannot get the global optimal value.

There have been all kinds of improved algorithms based on different methods to improve the performance of convergence. Some scholars study on the parameters selection and optimization. Shi and Eberthart first introduced inertia weight in the velocity updating equations to extend the search space and improve the ability to explore new areas[2]. Chatterjee and Siarry presented to adjust inertia weight with nonlinear method[3].Clerc presents a PSO with constriction factor algorithm similar to the maximum speed limit which can improve the convergence of the algorithm[4,5]. Monson improved location formula updating the particle location using Kalman filtering, which effectively reducing the number of iterations[6]. Asanga Ratnaweera and Cai presented study strategy to adjust acceleration coefficients respectively [7,8]. Some scholars analyze the influence to the performance of PSO from different topological structure and the correlation between topological structure and optimization problems systematically to describe the basic principle of population structure which provides the theory foundation for the population structure adjustment of PSO in given optimization problems. Suganthan provided a variable neighborhood operator PSO which introduced neighborhood operator to maintain the diversity of population. The neighborhood of particle is its own in the initial phase of algorithm running and the neighborhood would extend to the whole group gradually

doi:10.4304/jsw.8.4.827-833

with the iteration. Krink and Vesterstroem set different target of different stages in the process of solving problem to improve the diversity of population. Some scholars study on the fusion of different algorithms, such as Genetic Algorithm, Artificial Immune Algorithm, Ant Colony Optimization, Artificial Bee Colony[9,10,11,12,13], Simulated Annealing and Chaos Optimization Algorithms[14,15,16]. Other scholars study on the evolution strategy of PSO itself. Sun proposed modified particle swarm optimization with feasibility –based rules as constraint-handling mechanism[17]. A particle is not an integral individual and it can be divided into several parts. Each part could be updated and mutated independently[18,19]. Ji Qiang Zhai and Ke Qi Wang proposed a Baldwin effect based learning strategy utilizes the historical beneficial information to increase the potential search range and retains diversity of the particle population to discourage premature to improve the performance of PSO[20]. Shengli Song and Bing Lu proposed a novel PSO algorithm with adaptive space mutation[21]. Chen proposed particle swarm optimization with adaptive population size and its application[22]. Cai proposed dispersed particle swarm optimization[23].

The important reason of trapping into the local extreme value is that the differences between particles become smaller and all the particles cline to the particle in the best position which may not be the global optimal position in the process of iteration. The diversity of population is a main factor which affecting the performance of convergence[24].

This paper presents sub-dimension PSO(s-dPSO) which updates each dimension of particle instead of all dimensions meanwhile. When standstill occurs in the process of evolution, dimensions of particles in local convergence mutate which avoid blindness in evolution and improve the efficiency of algorithm.

II. Particle Swarm Optimization Algorithm

A. The Principle of Particle Swarm optimization Algorithm

PSO algorithm simulates the bird foraging behavior which is to achieve the goals through collective collaboration and competition among birds. In PSO, Each alternative solution is called a "particle" and many particles are in coexistence and cooperation for the optimal value similar to birds in search of food. At first PSO algorithm produces initial population, that is, randomly initiates a group of particles in the feasible solution space. Each particle is a feasible solution to optimization problems and fitness value is determined by the objective function. Each particle moves in the solution space and its direction and distance are determined by the speed. Particles usually move following the optimal one and get optimal solutions by iterative search. In each generation, particles trace two extremes; a Pbest is the optimal solution for the particle itself so far and another for the whole population so far to find the optimal solution for Gbest.

Mathematical description in PSO is as following: each single solution in the search space is a "bird" which can be called "particle". There are n particles forming a population in a d-dimensional search space. Each particle i represent a possible solution and have a position vector xi, a velocity vector vi, and the best personal position pi encountered so far by the particle. PSO is initialized with a group of random particles and then searches for optima by updating generations. In each generation, each particle moves in the direction of its own personal best position pi, as well as in the direction of the global best position pg discovered so far by any of the particles in the population. As a result, pi and pg can be used to adjust their own velocities and positions. This means that if a particle discovers a promising new solution, all other particles will move closer to it. At each generation, the velocity and the position of a particle i is updated using Eq 1 and 2:

$$v_{id}^{k'+1} = \omega v_{id}^{k'} + c_1 r_1 (p_{id} - z_{id}^{k'}) + c_2 r_2 (p_{gd} - z_{id}^{k'}) \quad (1)$$

$$z_{id}^{k'+1} = z_{id}^{k'} + v_{id}^{k'+1} \quad (2)$$

Where w is the inertia weight and typically setup to vary linearly from 0.9 to near 0.4 during the course of an iteration run; c1 and c2 are acceleration coefficients; r1 and r2 are uniformly distributed random numbers in the range (0,1)[25]. The velocity vi is limited to the range $[v_{min}, v_{max}]$. Updating velocity in this way enables the particle i to search around its individual best position pi , and the global best position pg. The position zi is limited to the range $[z_{min}, z_{max}]$.

Iterative termination conditions according to the specific problem are maximum number of iterations or the optimum position searched meeting the minimum adaptation threshold.

B. Steps of Algorithm

The steps of bPSO is as follows:

1. Initialize the velocity and position of particles randomly.
2. Calculate the fitness value of particles.
3. The fitness of each particle should be compare with its own personal best position Pbest and if the value is better than Pbest, the new position would be the position as the current Pbest.
4. The fitness of each particle should be compare with global best position Gbest and if the value is better than Gbest, the new position would be the position as the current Gbest.
5. Velocity and position of particles should be adjusted by formula 1 and 2.
6. If it cannot meet the end conditions ,jump to 2.

III. Sub-Dimension PSO

The performance of most stochastic optimization algorithm get worse when dimensions increase. BPSO updates all dimensions of particle and gets a fitness value to determine the fitness of solution[26]. The fitness value could estimate the quality of particle but not each dimension. In a three-dimension function which global optimum is [0,0,0] and the initial value is[1,1,1], the updated value is [1.2,0.5,1.3] after one iteration. If the fitness value is better than before, the value of the first and third dimension get worse. BPSO is difficult to ensure optimal direction of all dimensions in high dimensions function. The search space is divided into several dimensions to update and mutate to independently to resolve this problem.

A. The Principle s-dPSO

One particle is a position vector in s-dPSO presented in this paper. Each dimension of the particle is updated in turn in D-dimension space.

$$v_{id}^{k+1} = \omega v_{id}^{k} + c_1 r_1 (p_{id} - z_{id}^{k}) + c_2 r_2 (p_{gd} - z_{id}^{k}) \quad (3)$$

$$z_i^{k'} = z_{i1}^{k} + z_{i2}^{k} + \cdots (z_{id}^{k} + v_{id}^{k+1}) + \cdots z_{im}^{k} \quad (4)$$

The velocity and position of particle is limited in a range. The value of each dimension is initialized with a given range at first. Once the value of dimension is out of the range when updated, the value would be reset .

The fitness of particle would be calculated after updating operation.

$$z_i^k = z_i^{k'}, \; f\left(z_i^{k'}\right) < f\left(z_i^k\right) \quad (5)$$

When all dimensions have been updated, $z_i^{k+1} = z_i^k$ which can ensure the fitness of particle is not worse than the fitness of last iteration that is the particle is the best one of its history. The formula would be modified as follows

$$v_{id}^{k+1} = \omega v_{id}^{k} + c_1 r_1 (p_{gd} - z_{id}^{k}) \quad (6)$$

$$z_i^{k'} = z_{i1}^{k} + z_{i2}^{k} + \cdots (z_{id}^{k} + v_{id}^{k+1}) + \cdots z_{im}^{k} \quad (7)$$

B. Diversity of s-dPSO

All particles pursuit the global best particle in the optimization process. When particles sunk into the local convergence, it's important to jump out the local convergence and look for the global extreme for particles. Local convergence occurs if the change of the fitness value is less than a predetermined threshold after several iterations.

The distribution of each dimension is different. The value of the same dimension would be close with the iterative process, so the value should be mutated to increase difference.

The dimensional diversity is defined by the standard deviation of dimension in particle to determine the aggregation degree of a dimension. S is the number of particles, $x_{i,j}$ is the value of j-dimension of particle i, and $\overline{x}_j$ is the average value of j-dimension of all particles[27]. The diversity of j-dimension is as follows:

$$d(j) = \sqrt{\frac{1}{s}\sum_{i=1}^{s}(x_{i,j} - \overline{x}_j)^2} \quad (8)$$

The value of dimension should be mutated when sunk into the local extreme. Dimension mutating avoids blindness of particle mutating.

The standard deviation of each dimension is calculated and sorted from small to large. The dimension which standard deviation is sorted in the front would be mutated. The mutating operation is to initial the value of dimension instead of making a disturbance to the value of dimension. The particle is easy to trap into the local extreme again if disturbance is slight.

The steps of s-dPSO is as follows and flowchat is shown in figure1:

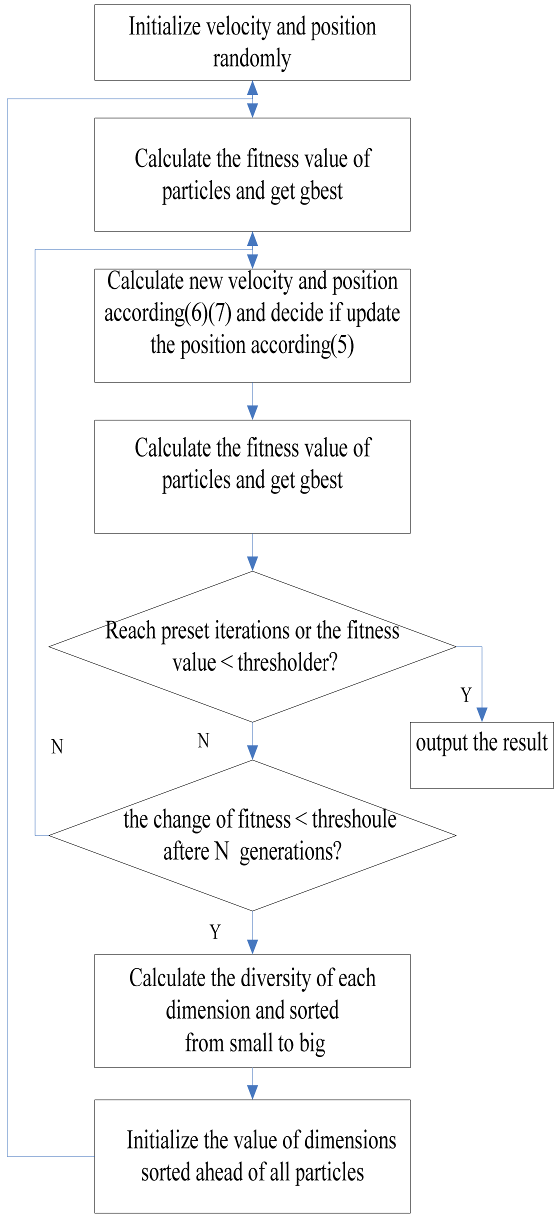


Figure 1. Flowchart of s-dPSO

1. Initialize the position and velocity of particles in swarm.
2. Calculate the fitness of particle and take the particle with highest fitness as gbest.
3. Update each dimension according formula(6)(7) and determine if update the value of dimension in the end.
4. Calculate the fitness of particle after updating all dimensions. If the value of fitness is better than gbest, gbest would be replaced.
5. Calculate the Standard deviation of each dimension and mutate the dimension which diversity is poor if the change of fitness is less than a predetermined threshold after several iterations.
6. Go to(3) if do not reach the end condition(good fitness or a preset maximum iterations).

IV. Performance Analysis And Evaluation Of S-Dpso

In order to balance the local and global search during the evolutionary process, a linearly varying inertia weight (ω) is over the course of generations and value is in the range [0.4,0.9]. When ω changes from 0.9 to 0.4, the search space of particle changes from a wide space to a small space gradually. A large number of experiments proved that if ω linear decrease in the iterations of the algorithm, the performance of convergence would be improved greatly. c1 is a learning factors usually c1 =2.

Four typical Benchmark functions are selected to be test function[28].

f1:Spherical $f_1(x) = \sum_{i=1}^{n} x_i^2$

Spherical is a simple unimodal function. The optimum solution of spherical is easy to be achieved by most value optimization algorithms. Its simplicity is conductive to the research on dimensions of question in optimization algorithm. The variables in this function do not influence each other and the gradient information is always pointing to the global optimum. Its optimal state and value is min(f(x))=f(0,…,0)=0. Two dimensional Spherical function is shown in figure 2.

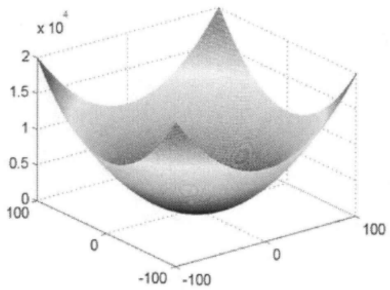
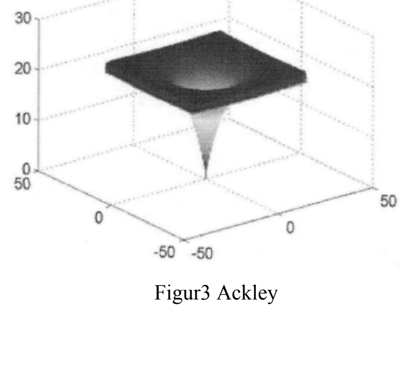


Figure 2 Spherical

f2:Ackley

$$f_2(x) = -20\exp\left(-0.2\sqrt{\frac{1}{n}\sum_{i=1}^{n} x_i^2}\right) - \exp\left(\frac{1}{n}\sum_{i=1}^{n}\cos(2\pi x_i)\right) + 20 + e$$

Ackley is a multimodal function with a lot of local optimal points and a global optimal point. No correlation between variables. Optimization algorithm is easy to trap into the local point in the global optimum paths. Its optimal state and value is min(f(x))=f(0,…,0)=0. Two dimensional Ackley function is shown in figure 3.



Figur3 Ackley

f3:Rastrigin

$f_3(x) = \sum_{i=1}^{n}(x_i^2 - 10\cos(2\pi x_i) + 10)$

Rastrigin is a multimodal version in Sphere functions. There are a large number of sine inflection point arrangement and deep local optimal point. No correlation between variables and the local optimal points wave with sine. Optimization algorithm is easy to trap into the local point in the global optimum paths. Traditional optimization algorithms are very difficult to find the optimal solution. Its optimal state and value is min(f(x))=f(0,…,0)=0. Two dimensional Rastrigin function is shown in figure 4.

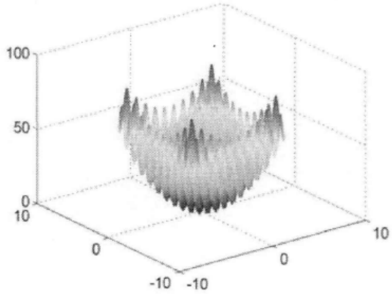


Figure 4 Rastrigin

f4:Griewank

$f_4(x) = \sum_{i=1}^{n}\frac{1}{4000}(x_i)^2 + \prod_{i=1}^{n}\cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$

Griewank is a multimodal function and variables in it affect each other. There are a lot of local optimal points. Its optimal state and value is min(f(x))=f(0,…,0)=0. Two dimensional Griewank function is shown in figure 5.

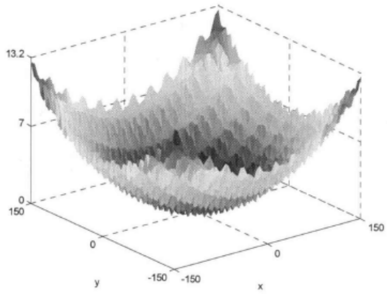


Figure 5 Griewank

Parameters setting of two algorithms are the same. The number of iterations is 1000. Four functions are set up for the 10 dimensions, the value of each dimension in F1 ranges from [-100,100], the value in F2, F3 and F4 ranges from [-10,10] and the target minimum value is 0. Parameter settings are shown in table 1.

TABLE I.

TEST FUNCTION AND PARAMETERS SETTING

function	Value range	Population size	Convergence precision
F1	[-100,100]	100	1E-10
F2	[-10,10]	100	1E-6
F3	[-10,10]	100	1E-6
F4	[-10,10]	100	1E-6

The test on 4 Benchmark functions are executed 50 times respectively using two different algorithms. The comparison of algorithms performance is based on optimal solution, average optimal solution, the standard deviation of solution and convergence rate The results are shown in table II to table V.

TABLE II.

TEST RESULTS ON OPTIMAL SOLUTION

function	algorithm	Optimal solution	Optimal iteration
F1	bPSO	7.997582E-11	360
	s-dPSO	5.801163E-11	588
F2	bPSO	--	--
	s-dPSO	2.038835E-07	410
F3	bPSO	--	--
	s-dPSO	2.160583E-07	839
F4	bPSO	--	--
	s-dPSO	4.623512E-07	660

Optimal solution is to measure the ability of exploring the solution. The result of optimal solution is shown in table II. When benchmark functions are optimized by bPSO, three functions F2, F3 and F4 could not find the optimal solution. Even if function F1 find the optimal solution, the number of iteration is larger than the one in s-dPSO. When Benchmark functions are optimized by s-dPSO, all functions could find the optimal solution. S-dPSO is effective to four benchmark functions.

TABLE III.

TEST RESULTS ON AVERAGE OPTIMAL SOLUTION

function	algorithm	Average optimal solution	Average optimal iteration
F1	bPSO	8.453675E-11	875
	s-dPSO	7.143528E-11	435
F2	bPSO	--	--
	s-dPSO	6.224587E-07	660
F3	bPSO	--	--
	s-dPSO	5.372487E-07	1078
F4	bPSO	--	--
	s-dPSO	7.186932E-07	784

Average optimal solution is to measure the overall quality of solution. The result of average optimal solution is shown in table III. The average optimal solution in s-dPSO is better than the one is Bpso.

TABLE IV.

TEST RESULTS ON STANDARD DEVIATION

function	algorithm	Standard deviation
F1	bPSO	26453.12
	s-dPSO	68.31582
F2	bPSO	46.43586
	s-dPSO	3.649246
F3	bPSO	918.29831
	s-dPSO	182.06821
F4	bPSO	156.24713
	s-dPSO	1.261871

Standard deviation is to measure the stability of solution in the given numbers of iteration. The result of the standard deviation is shown in table IV. If the value of standard deviation is small, the stability of solution is strong. The stability in s-dPSO is better than bPSO in four functions.

TABLE V.

TEST RESULTS ON OPTIMAL SOLUTION

function	algorithm	Convergence rate
F1	bPSO	32%
	s-dPSO	100%
F2	bPSO	0
	s-dPSO	83%
F3	bPSO	0
	s-dPSO	38%
F4	bPSO	0
	s-dPSO	51%

Convergence rate is to measure the convergence performance of algorithm. The result of the standard deviation is shown in table V. F1 could find the solution but the rate of convergence is only 32% in bPSO. When Benchmark functions are optimized by s-dPSO, all functions could find the optimal solution. The rate of convergence in F1 is 100% and other three function are 83%, 38% and 51%.

The curve of optimal value in bPSO and s-dPSO are shown in figure 6 to figure 9. Much important information could be learned from these figures. The functions of F3 and F4 are multimodal functions and bPSO could not find the global optimal value once trap into the local convergence. Because bPSO just trends to pbest and gbest particles in the process of evolution and has no strategies, it is difficult to jump out the local minimum value.

Dimensions with poor diversity in the process of evolution would be mutated in s-dPSO which ensure s-dPSO could jump out the local convergence and continue to look for the global optimal value. Figure 8 and figure 9 show the trend of curve. We can learn that the solution is achieved after several variations in s-dPSO.

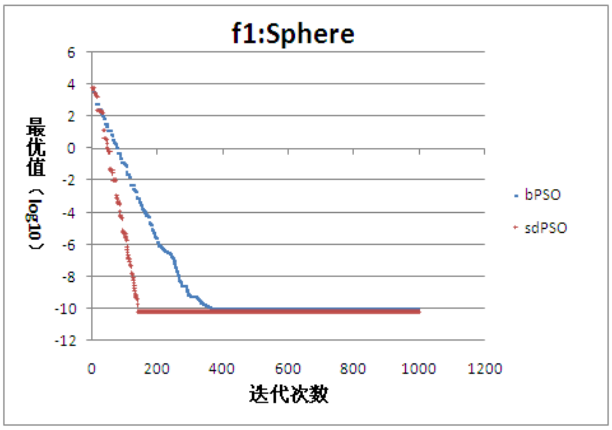


Figure 6. Curve of optimal value in bPSO and s-dPSO of F1

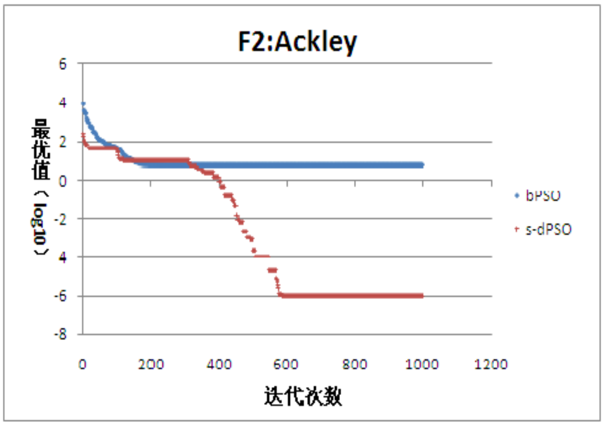


Figure 7. Curve of optimal value in bPSO and s-dPSO of F2

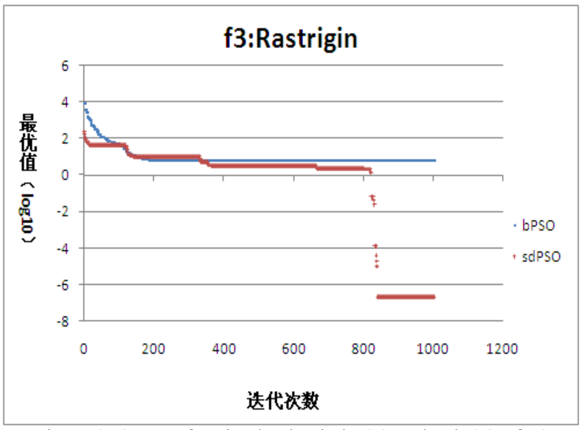


Figure 8. Curve of optimal value in bPSO and s-dPSO of F3

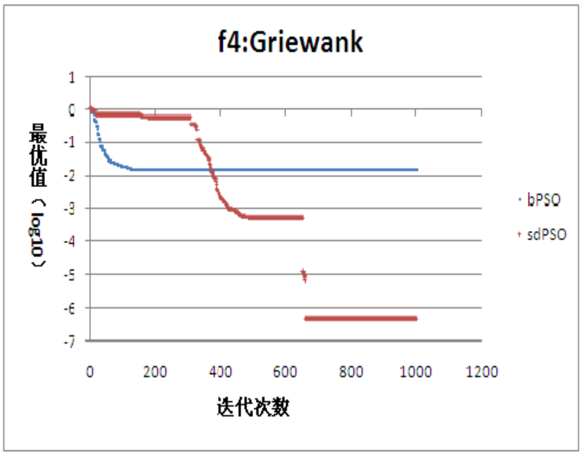


Figure 9. Curve of optimal value in bPSO and s-dPSO of F4

V. Conclusion

BPSO is a randomized global optimization mechanism. In each generation, each particle moves in the direction of its own personal best position Pbest, as well as in the direction of the global best position Gbest discovered so far by any of the particles in the population. The population of particles tends to cluster together and decrease the diversity of population after a certain number of generations. BPSO is easy to be in the local convergence. S-dPSO is proposed in this paper. The dimension of a particle is updated in turn in s-dPSO. When particles are all in the local convergence, the dimensions in poor diversity would be mutated which avoid blindness of mutating and improve the efficiency of optimization. Through simulation the ability of optimization of s-dPSO is significantly better than the one of bPSO, especially in the case of multiple local extremes.

Acknowledge

This work was supported by the National Natural Science Foundation of China (No. 40972207), the National Science and Technology Major Projects (No. 2011ZX05034-005) and the PAPD of Jiangsu Higher Education Institutions. These supports are gratefully acknowledged.

References

[1] Zhen Ji, Jiarui Zhou, Huilian Liao and Qinghua Wu, "A novel intelligent single particle optimizer", *Chinese journal of computers*, vol.33, No.3, 2010, pp.556-561.

[2] Shi Y, Eberhart R C, "A modified particle swarm optimizer", *Proceeding of the IEEE Congress on Evolutionary Computation*, 1998, pp.69-73.

[3] Chatterjee A., Siarry P., "Nonlinear Inertia Weight Variation for Dynamic Adaptation in Particle Swarm Optimization", *Computer and Operations Research*, no.33, 2006, pp.859-871.

[4] Clerc M, "The swarm and the queen:towards a deterministic and adaptive particle swarm optimization", *Proceeding of the 1999 Congress on Evolutionary Computation*, 1999, pp.1927-1930.

[5] Clerc M., Kennedy J., "The Particle Swarm: Explosion, Stability and Convergence in a Multidimensional Comples Space", *IEEE Transaction on Evolutionary Computation*, no.6, 2002, pp.58-73.

[6] Monson C K, Sepp K D, "The Kalman Swarm-A New App roach to Particle Motion in Swarm Optimization", *Proceedings of the Genetic and Evolutionary Computation Conference*, 2004, pp.140-150.

[7] Ratnaweera A., Halgamuge S.K., Watson H.C., "Self-Organizing Hierarchical Particle Swarm Optimizer with Time-Varying Acceleration Coefficients", *IEEE Transaction on Evolutionary Computation*, Vol.8, no.3, 2004, pp.240-255.

[8] Cai X.J., Cui Z.H., Zeng J.C.,et al., "Particle Swarm Optimization with Self-adjusting Cognitive Selection Strategy", *International Journal of Innovative Computing, Information and Control*, vol.14, no.4, 2008, pp.943-952.

[9] Karaboga D, "An Idea Based on Honey Bee Swarm for Numerical Optimization",2005, *Technical Report-06*.

[10] Karabogy D, Akay B, "A Comparative Study of Artificial Bee Colony Algorithm", *Applied Mathematics and Computation*, vol.14, no.2, 2009, pp.108-132.

[11] Akay B, Karaboga D, "A Modified Artificial Bee Colony Algorithm for Real-parameter Optimization", 2010, *Information Science*.

[12] Shelokar P S, Siarry P, Jayaraman V K, Kulkarni B D, "Particle swarm and ant colony algorithms hybridized for improved continuous optimization", *Applied Mathematics and Computation*, vol.188, no.1, 2007, pp.129-142.

[13] Chen H.C., Zhu Y.L., "Optimization based on symbiotic multi-specied coevolution", *Applied Mathematics and Computation 205*, 2008, pp.47-60

[14] Chuanwen J., Bompard E., “A hybrid Method of Chaotic Particle Swarm Optimization and Linear Interior for Reactive Power Optimization”, *Mathematics and Computers in Simulation*, 2005, pp.57-65.
[15] Dos Santos Coelho L., Herrera B.M., “Fuzzy Identification based on a Chaotic Particle Swarm Optimization Approach Applied to a Nonlinear Yo-yo motion System”, *IEEE Transactions on Industrial Electronics*, vol.54, no.6, 2007, pp.3234-3245.
[16] Liu B., Wang L., Jin Y.H., et al., “Improved Particle Swarm Optimization Combined with Chaos”, *Chaos, Solitions & Fractals*, vol.25, no.5,2005,pp.1261-1271.
[17] Chaoli Sun, Ying Tan, Jianchao Zeng, et al., “The structure optimization of main beam for bridge crane based on an improved PSO”, Journal of Computers,vol.6, no.8, 2011, pp.1585-1990.
[18] Peram T, Veeramachaneni K, Mohan C.K., “Fitness-distance -ratio Based Particle Swarm Optimization”, *Proceedings of the Swarm Intelligence Symposium*, 2003, pp.174-181.
[19] Mallipeddi R., Suganthan P.N., Pan Q.k., et al, “Differential evolution algorithm with ensemble of parameters and mutation strategies”, *Applied Soft Computing*, vol.11, 2011, pp.1679-1696.
[20] Ji Qiang Zhai and Ke Qi Wang, “Baldwin effect based particle swarm optimizer for multimodal optimization”, *Journal of Computers*, vol.7, no.9, 2012,pp.2114-2119
[21] Shengli Song, Bing Lu, Li Kong and Jiangjing Cheng, “A novel PSO algorithm model based on population migration strategy and its application”, *Journal of Computers*, vol.6, no.2, 2011, pp.280-287.
[22] Chen D B, Zhao C X, “Particle swarm optimization with adaptive population size and its application”, *Applied Soft Computing*, vol.9,no.1, 2009, pp.39-48.
[23] Cai X J, Cui Z H, Zeng J, Tan Y, “Dispersed particle swarm optimization”, *Information Processing Letters*, vol.105, no.6, 2008,pp.231-235.
[24] Sabat S.L., Ali L. and Udgata S.K., “Integrated learning particles swarm optimizer for global optimization”, *Applied Soft Computing*, vol.11, no.1, 2011, pp.574-584
[25] Jiangbo Yu, Lifeng Xi, Shijin Wang, “An Imoroved Particle Swarm Optimization for Evolving Feedforward Artificial Neural Networks”, *Neural Process Lett*, vol.26, 2007, pp.217-231.
[26] Guojiang Fu, Shaomei Wang, Shuyan Liu, “A PSO with dimension mutation operator”, *Journal of Wuhan University of Hydraulic and Electric Engineering*, vol.38, No.4, 2005, pp.79-83.
[27]Garcia-Villoria A, Pastor R, “Introducing dynamic diversity into a discrete particle swarm optimization”-, .*Computers and Operations Research*, vol.36, 2009, pp.951-966.
[28] Liang JJ, QinAK, Suganthan Pnetal, “Comprehensive learning particle swarm optimizer for global optimization of multimodal functions”, *IEEE Transaction on Ebolutionary Computation*, vol.10, No.3, 2006, pp.281-295.

Xu Hui is current a Ph.D .candidate at China University of Mining and Technology(CUMT), China. She received her MS degree in Computer Application Technology from CUMT in 2005, and her BS degree in Computer Science from CUMT in 2002. She is currently a lecture at school of Computer Science and Technology, CUMT. Her research interest is computation intelligence and coalbed methane et al.

Yongguo Yang is born in 1962, Ph.D. He is a professor at school of Mineral Resource and Earth Science and a director of Geo-Information Science Institute in CUMT. His research interests are mathematical geology and GIS applications. He has published 3 books, and more than 40 research papers in journals and international conferences. Now he preside the National Natural Science Foundation of China (No. 40972207), the National Science and Technology Major Projects (No. 2011ZX05034-005) and the PAPD of Jiangsu Higher Education Institutions.

Mao Lei is currently a Ph.D.candidate at China University of Mining and Technology, China. He received his MS degree in Computer application Technology from China University of Mining and Technology in 2004, and his BS degree in Computer Science from China University of Mining and Technology in 2000. He is currently a lecturer at School of Computer Science and Technology, China University of Mining and Technology. His research interests include cloud computing, workflow system and parallel processing et al.

Hongxia xie is currently a Ph.D.candidate at China University of Mining and Technology, China. She received her MS degree in Computer application Technology from China University of Mining and Technology in 2005, and her BS degree in Computer Science from China University of Mining and Technology in 2002. She is currently a lecturer at School of Computer Science and Technology, China University of Mining and Technology. Her research interests are in distributed parallel processin g, data mining, neural network.