A Hybrid Particle Swarm Optimization Algorithm Based on Nonlinear Simplex Method and Tabu Search

Zhanchao Li[1,2], Dongjian Zheng[1,2], and Huijing Hou[1]

[1] College of Water Conservancy and Hydropower Engineering, Hohai University, Nanjing 210098, Jiangsu, China
lizhanchao@gmail.com
[2] State Key Laboratory of Hydrology-Water Resources and Hydraulic Engineering, Hohai University, Nanjing 210098, Jiangsu, China

Abstract. Particle swarm optimization (PSO) algorithm is an intelligent search method based on swarm intelligence. It has been widely used in many fields because of its conciseness and easy implementation. But it is also easy to be plunged into local solution and its later convergence speed is very slow. In order to increase its convergence speed, nonlinear simplex method (NSM) is integrated into it, which not only can increase its later convergence speed but also can effectively avoid dependence on initial conditions of NSM. In order to bring particles jump out of local solution regions, tabu search (TS) algorithm is integrated into it to assign tabu attribute to these regions, which make it with global search ability. Thus the hybrid PSO algorithm is an organic composition of the PSO, NSM and TS algorithms. Finally its basic operation process and optimization characteristics are analyzed through some benchmark functions and its effectiveness is also verified.

Keywords: hybrid algorithm, particle swarm optimization, nonlinear simplex method, tabu search.

1 Introduction

The PSO algorithm [1,2] is still in the preliminary stage since it was proposed in 1995, and there are still many theories of it should to be perfected. It is widely used in function optimization, neural network training, fuzzy system control and so on because of its conciseness, easy implementation, needing to adjust little parameters, not requiring gradient information and other excellent features [3]. But how to increase the convergence speed and how to avoid premature convergence have always been the focus of most researchers and they are also the problems faced by all the other random search algorithms [4]. One of the main directions to improve the PSO algorithm is to establish the hybrid PSO algorithm. Hendtlass [5] combined different evolution with the PSO. Parsopoulos [6] etc initialized the PSO using NSM. Mirnada [7] etc put together the best features of evolution strategies with the PSO. Krink [8] etc introduced a hybrid approach called Life Cycle model that simultaneously applied genetic algorithms, PSO and stochastic hill climbing to create a generally well-performing search heuristics. Shi

L. Zhang, J. Kwok, and B.-L. Lu (Eds.): ISNN 2010, Part I, LNCS 6063, pp. 126–135, 2010.

[9] etc proposed two hybrid evolutionary algorithms based on PSO and GA in parallel and series forms respectively. Noel [10] etc introduced a hybrid PSO making use of gradient information to achieve faster convergence without getting trapped in local minima. Wachowiak [11] etc embedded Powell method in the PSO to improve accuracy. Vietoire [12] etc integrated the PSO technique with the sequential quadratic programming technique to solve the economic dispatch problem.

The hybrid PSO algorithms mentioned above are mainly hybrid algorithms of global optimization algorithms with local optimization algorithms or other global ones. Essence of these hybrid algorithms is using advantages of each algorithm, which allows different algorithms to perform their strengths and avoid their shortcomings, so as to reach equilibrium among them. However, existing hybrid PSO algorithms have hardly both studied the increase of convergence speed and the avoidance of premature convergence at the same time. Moreover, it has also not been studied deeply enough to avoid premature convergence, which is not perfect for a good algorithm. Here it is improved from two aspects: (1) using the NSM which has a strong local search ability to enhance its local search ability in the later stage; (2) using the TS algorithm to deal with the local extremum regions so as to bring the particles within them all out, and at the same time to avoid the low efficiency of the TS algorithm caused by its limitation in dealing with plentiful individuals (particles).

2 Backgrounds of the PSO, NSM and TS Algorithm

2.1 Particle Swarm Optimization (PSO)

In the PSO algorithm, particle i is expressed as $X_i=(x_{i1}, x_{i2},\ldots, x_{iD})$, which represents a point in the D-dimensional solution space. Each particle saves the best position of itself so far $P_i=(p_{i1}, p_{i2},\ldots, p_{iD})$ and its current flying speed $V_i=(v_{i1}, v_{i2},\ldots, v_{iD})$. The best location so far found by the whole particle swarm is $P_g=(p_{g1}, p_{g2},\ldots, p_{gD})$. In each iteration, the particle's flying speed in the D-dimensional space is updated through P_g, P_i and X_i. Then its position is updated through the updated flying speed. The updating formulae of the PSO algorithm are as flows.

$$\begin{aligned} V_{id}^{New}(t+1) = w \times V_{id}^{old}(t) + c_1 \times rand() \times (p_{id}(t) - x_{id}^{old}(t)) \\ + c_2 \times rand() \times (p_{gd}(t) - x_{id}^{old}(t)) \end{aligned} \tag{1}$$

$$x_{id}^{New}(t+1) = x_{id}^{old}(t) + V_{id}^{New}(t+1) \tag{2}$$

$$v_{\min} \le v_{id} \le v_{\max}, x_{i\min} \le x_i \le x_{i\max} \tag{3}$$

Where c_1 and c_2 are two positive constants called learning factors, which make each particle with the ability of self-summary and learning from the outstanding particles, so as to get close to the best positions of itself and the whole swarm so far; w is the inertia weight factor, which can dynamically adjust the search ability of the particle swarm along with the time; rand()is the random number between 0 and 1, which is used to maintain the diversity of the particle swarm. (1) is the updating formula of the particle speed, which indicates that each particle updates its speed based on its original speed V_{id}, its best location so far P_{id} and the best location of the whole particle swarm P_{gd}. (2) is the updating formula of the particle position and (3) is the constraints of the particle speed and position.

2.2 Non-linear Simplex Method (NSM)

The NSM was developed by Nelder and Mead [13] based on the basic simplex method. It is a widely used direct local search technology for non-linear unconstrained optimization problems because it can complete the optimization directly according to the function value without its derivative information. Its basic principle is that: first construct a convex polyhedron with N+1 vertices in the N-dimensional Euclidean space E_N, calculate the function value of each vertex and determine the maximum, the second largest value and the minimum; then find a better solution to substitute the maximum through reflection, expansion, contraction and reduction; finally approach a better minimum through many iterations.

2.3 Tabu Search (TS)

The TS algorithm is firstly proposed in 1986 by Glover [14] and is a sub-heuristic search technique. It is a reflection of artificial intelligence and an extension of local search. Its most important thought is that it can mark the searched local minima and can avoid them as much as possible so as to ensure an effective search path. Its basic principle is that: determinate a number of candidates of the current solution in its given neighborhood; if the best candidate is better than the best solution so far, its taboo attribute is neglected and it is used to substitute the current solution and the best solution so far, and the tabu table and the tenure are modified; if the candidate mentioned above does not exist, choose the best one in the non-taboo candidates as the new current solution while ignoring its strengths and weaknesses to the current solution and modify the tabu table at the same time; repeat the above iteration until the end criteria is met.

3 The Hybrid PSO Algorithm

3.1 Hybrid Strategy

The PSO algorithm is suitable for non-linear and multi-extremum optimization problems because of its conciseness, easy implementation, fast calculation, not requirement for the objective function's mathematical form and its gradient information. And also it has better portability, robustness and can compute in parallel. The NSM can complete optimization directly according to the function value without its derivative information. So it is a widely used direct local search technology for non-linear unconstrained optimization problems. Because of its needless of the first derivative, Hessen matrix and complex matrix operations, it is particularly adapted to the optimization of complex functions with incomplete information. However, this method is very sensitive to the initial conditions, and can not be guaranteed to converge to the global optimal solution. The TS algorithm has a strong "mountain climbing" ability to jump out of the local optimal solution in the search process so as to search in other regions. So the probability to get a better solution or the global optimal solution is greatly increased.

Generally, it is difficult to realize efficient optimization only depending on search algorithms with a single neighborhood structure, and it is an effective means to broaden application scope and improve performance of an algorithm that make the algorithm with a hybrid neighborhood structure [15]. At present, the hybrid structure of an algorithm can be divided into serial, mosaic, parallel and mixed structures. Hybrid algorithm with a serial structure can absorb advantages of different algorithms. For example, it can use results of an algorithm as the starting point of another algorithm to optimize the problem sequentially and its purpose is to improve the optimization efficiency on the premise of a certain good quality. Hybrid algorithm with a mosaic structure performs that an algorithm is an optimization operation of another algorithm or an evaluating device of the search performance. Hybrid of these algorithms is in view of their complementarities so as to overcome the premature convergence and (or) the plunge into local minimum of a single algorithm. And mosaic structure allows the aided algorithm to be executed repeatedly and the information between the sub-algorithms can exchange bidirectional. Therefore, the mosaic structure is applied to the hybrid of the PSO, NSM and TS algorithm.

For global optimization problems, a good algorithm should have a strong global exploration ability to obtain suitable seeds in the earlier stage while a strong local search ability to increase the convergence speed in the later stage, and also can jump out of the local minimum to search in other regions when the obtained solution does not meet accuracy requirements. The PSO algorithm has a faster convergence speed and a better global exploration ability in the earlier stage but a slower convergence speed in the later stage. The main reason is that in the later stage most particles have congregated in the vicinity of the optimal particle and almost stopped moving. But the approximate area of the minimum has been well located which provides good initial conditions for the NSM. Therefore, application of the NSM at this stage can not only strengthen the local search ability of the PSO algorithm, but also avoid the easy plunge of the NSM into the local minimum because of its very dependence on initial conditions. This hybrid algorithm performs well on functions with few local minimums, while has a limited improvement on multimodal functions with lots of local minima and global extremums because even in the simple low-dimensional cases the computational complexity of these functions is quite high and is a typical NP-hard problem. Parospoulos [16,17] etc used "function stretching" technology to make particles jump out of local minima. But transformation of the objective functions may generate pseudo local minima and misleading gradient information. Although the PSO algorithm does not directly use the gradient information, it is convinced to be used in some indirect way [4]. Here the TS algorithm is used to make particles jump out of the local minimum, that is, when the particles are plunged into it, the local minimum region is dealt with by the TS algorithm and thus the particles can keep away from this region to search globally.

3.2 Basic Steps of the Hybrid PSO Algorithm

The main idea of the hybrid PSO algorithm (PSO_NSM_TS) is that: First the particle swarm searches globally in the whole solution space in view of its fast pre-convergence speed and overall exploration ability. Then the first N+ 1 excellent particles are isolated from the swarm in the later stage and the NSM is used to strengthen its local search abilities so as to find the local minimum. When the particle swarm is plunged into a local minimum, the TS algorithm is used to deal with the local minimum region, and particles in this region are re-initialized in the whole solution space. So particles can keep away from this region to search globally in the subsequent search process. The basic steps are as follows:

(1) Initialize the particle swarm: the initial position and velocity of each particle are randomly initialized in the solution space for the given particle swarm with 3N+1 particles;

(2) Calculate fitness of each particle and sort: the fitness of each particle is calculated according to the objective function and all particles are sorted from small to big according to the fitness;

(3) Calculate the average particle distance [18] of the fist N+1 excellent particles $D(t) = \frac{1}{SgL}\sum_{i=1}^{s}\sqrt{\sum_{d=1}^{N}(p_{id} - \overline{p}_d)^2}$, where L is the maximal diagonal length of the search space; S is the swarm size, here taken to be N+1; N is the dimension of the solution space;

(4) Judge the average particle distance: if D(t)≤[D], call the NSM and go to step (5); if D(t)>[D], use the PSO algorithm to update the whole particle swarm and go to step (8);

(5) Call the NSM: the NSM is used among the first N+1 excellent particles to find the local solution;

(6) Judge the local solution: if the local solution searched by the NSM meets accuracy requirement, the whole optimization is completed and exits; if not, call the TS algorithm;

(7) TS algorithm updates the tabu table: add the current local solution region to the tabu table and update the tenure; Because the current speed of the particles in the local minimum region is very slow and in order to bring them to quickly jump out of this region, the particles within the region are re-initialized. Meanwhile, in order to avoid the particle swarm to be plunged into the local minimum region again, the best positions of these particles are also re-initialized and the best position of the swarm is updated correspondingly; In order to organically combine with the PSO algorithm and also not increase the complexity of the hybrid algorithm, poor fitness is given to the particles within the tabu regions during the PSO updating process and the particles are kept away from the tabu regions due to the attraction of the outstanding particles.

(8) Update the particle swarm: update the particle swarm according to the speed and location updating formula, and go to step (2).

The basic flowchart of the hybrid PSO algorithm is shown in Fig. 1.

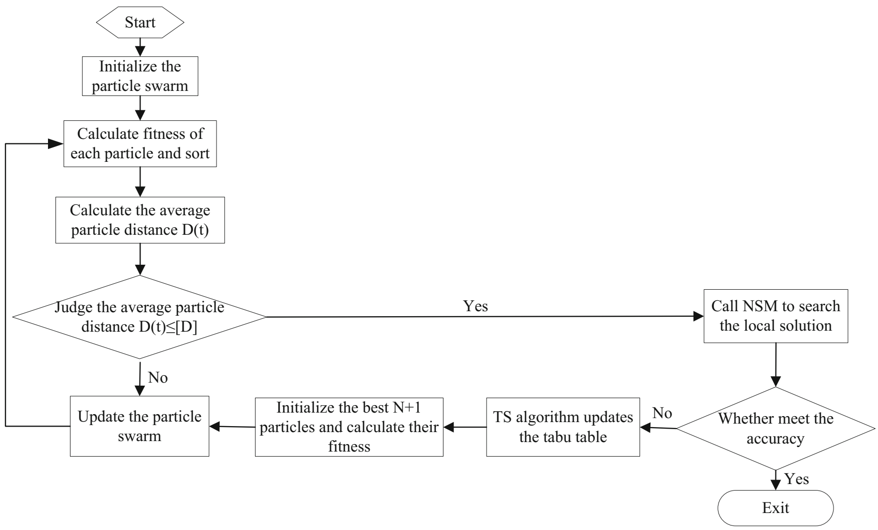

Fig. 1. Flowchart of the hybrid PSO algorithm

4 Optimization Analysis of the Hybrid PSO Algorithm

In order to analyze the basic operation process of the hybrid PSO algorithm, the DeJong function [3] is taken as an example to be optimized. Its expression is $f(x)=\sum_{i=1}^{N} x_i^2$, $-100 \le x_i \le 100$. Its global minimum is min(f)=f(0,…,0)=0. The change process of the best fitness so far along with the iteration of the hybrid PSO algorithm is shown in Fig. 2.

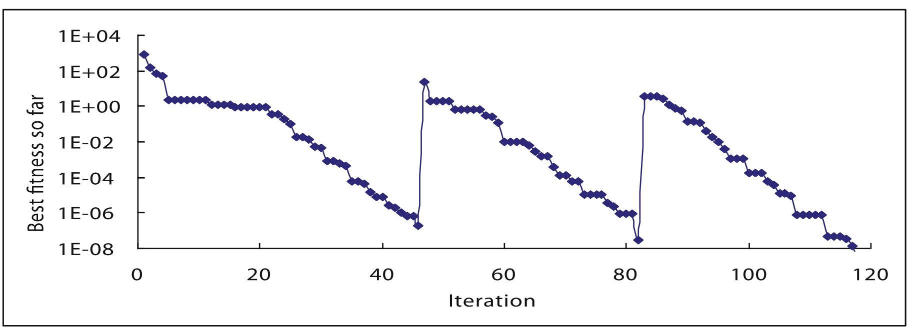

Fig. 2. Change process of the best fitness so far along with the iteration

As can be seen from Fig. 2, in the earlier stage the hybrid PSO algorithm searches in the solution space according to the basic PSO algorithm. And the NSM is called when the first N+1 excellent particles satisfy the condition of D(t)≤[D]. When the solution found by the NSM within the allowed iteration does not meet accuracy requirement, it is considered to be plunged into the local minimum and the TS algorithm is used to bring the particles jump out of this region.

In order to analyze the characteristics of the hybrid PSO algorithm in dealing with multi-minimum problems, the Rastrigin function is taken as an example. Its expression [3] is $f(x)=\sum_{i=1}^{N}[x_i^2-10\cos(2\pi x_i)+10]$, $-10\le x_i \le 10$. Its global minimum is min(f)=f(0,…,0)=0. Fig. 3 shows the two-dimensional Rastrigin function. This function is tested 50 times using the basic PSO algorithm, the PSO_NSM hybrid algorithm, the PSO_TS hybrid algorithm and the PSO_NSM_TS hybrid algorithm respectively, and the average iterations, the convergence success rate and the average number of calling the TS algorithm of these algorithms are shown in table 1. Fig. 4 shows the change process of the best fitness so far along with the iteration of the basic PSO algorithm and the PSO_NSM hybrid algorithm under convergence and the PSO_NSM_TS hybrid algorithm and the PSO_TS hybrid algorithm under convergence when calling TS.

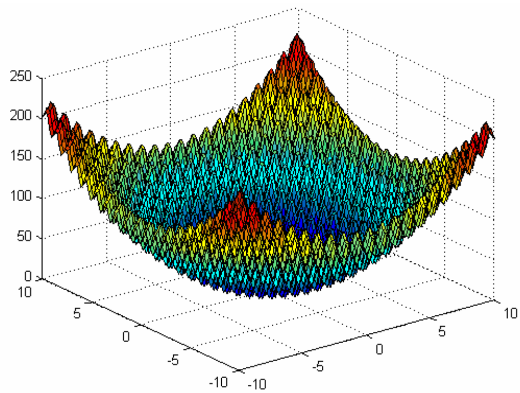

Fig. 3. The two-dimensional Rastrigin function

Table 1. Result of these PSO algorithms

Type	Average iterations[1]	Convergence success rate (%)	Average number of calling TS
PSO_NSM_TS	228.76	100	0.64
PSO_TS	358.28	100	1.18
PSO_NSM	102.66	68	/
PSO	128.26	38	/

[1] The average iterations of the PSO_NSM hybrid algorithm and the basic PSO algorithm are only iterations under convergence.

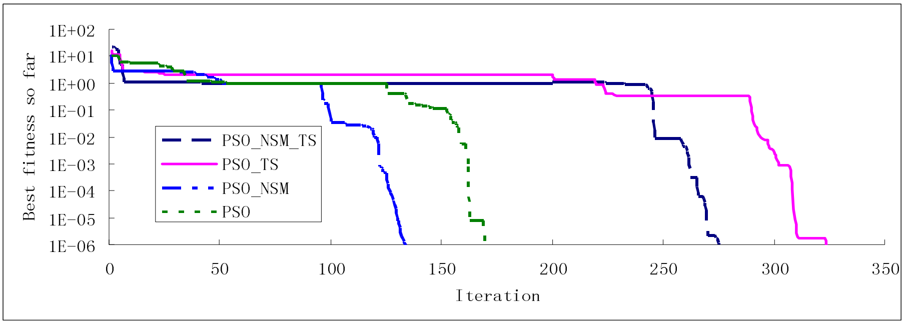

Fig. 4. The best fitness so far along with the iteration of the PSO_NSM_TS, PSO_TS, PSO_NSM and PSO

As can be seen from Table 1 and Fig. 4, for the multi-minimum problems the PSO_NSM hybrid algorithm is the fastest, the basic PSO algorithm second, while the PSO_TS hybrid algorithm is the slowest which indicates that the convergence speed can be increased through the introduction of the NSM. The convergence success rates of the basic PSO algorithm and the PSO_NSM hybrid algorithm are only 38% and 68%, while the convergence success rate can be increased to 100% through the introduction of the TS algorithm. The average number of calling the TS algorithm of the PSO_NSM_TS hybrid algorithm is only 0.64 which is far lower than that of the PSO_TS hybrid algorithm. Furthermore, other benchmark functions are also used to test the proposed hybrid algorithm such as the generalized Rosenbrock function, Griewank function and so on. The result shows that the proposed hybrid algorithm in this paper not only can increase the convergence speed but also can avoid premature convergence, which is the starting point of this paper.

In a word, the key point of the hybrid PSO algorithm (PSO_NSM_TS) is the introduction of the NSM and the TS algorithm. The introduction of the TS algorithm prevents the re-plunge of the particles into the known local minimum regions to a great extent. Strength of the preventability is embodied by the tabu length in the TS algorithm. When the tabu length is set to zero, that is, just initialize the first N+1 excellent particles in the current local minimum region and not memorize the previously searched regions. At the moment, the hybrid algorithm is similar to a completely random algorithm. That is, when the particle swarm is plunged into the local minimum region the first N+1 excellent particles are re-initialized to search in the global solution space. When the tabu length is set to infinite, that is, all the preciously searched local minimum regions are memorized and dealt with by the TS algorithm, so the particles search in other un-searched regions and eventually can be able to find the global minimum solution. Therefore, setting of the tabu length not only can harmonize the global search ability and the search speed, but also can harmonize the possibility to treat the local minimum solution as the global minimum solution.

5 Conclusion

The basic PSO algorithm is improved through accelerating the convergence speed and avoiding premature convergence. The improved hybrid PSO algorithm (PSO_NSM_TS) is an organic composition of the basic PSO, NSM and TS algorithm. In the hybrid algorithm, the NSM is introduced to strengthen the local search ability of the PSO algorithm in the later stage, and the TS algorithm is introduced to bring particles jump out of the local minimum regions when the particle swarm is plunged into them. So the hybrid PSO algorithm can increase the convergence speed and avoid premature convergence to a great extent. And the benchmark functions also verify its correctness and high efficiency. The pseudo-minima tend to be generated when the tabu regions are dealt with poor fitness. But they only survive within the tabu length. When the local minimum regions which generate them are released, they are automatically eliminated. So using the TS algorithm can solve the pseudo-minima problem to a certain extent. And it can increase the convergence speed and reduce the calculation work when the pseudo-minimum problem is effectively solved. At the same time, there are still some problems should to be deeply studied. For example, the hybrid algorithm only considers the region within the average particle distance [D]. If all regions converging to the local minimum can be found and treated, the re-plunge into the local minimum can be completely avoided, which not only increase the global convergence speed but also can improve the search efficiency.

Acknowledgement

This paper is supported by National Natural Science Foundation of China (Grant Nos. 50879024, 50809025, 50539010), National Science and Technology Supporting Plan (Grant No. 2006BAC14B03), Specialized Research Fund for the Doctoral Program of Higher Education (Grant No. 20070294023) and Graduate Innovation Program of University in Jiangsu Province (Grant No. CX09B_163Z).

References

1. Eberhart, R.C., Kennedy, J.: A New Optimizer Using Particle Swarm Theory. In: Proceedings of the Sixth International Symposium on Micro Machine and Human Science, Nagoya, Japan, pp. 39–43 (1995)
2. Kennedy, J., Eberhart, R.C.: Particle Swarm Optimization. In: Proceedings of the IEEE International Conference on Neural Networks, Piscataway, NJ, USA, pp. 1942–1948 (1995)
3. Ji, Z., Liao, H.L., Wu, Q.H.: Particle Swarm Optimization and Its Application. Science Press, Beijing (2009) (in Chinese)
4. Wang, F.: Research on Particle Swarm Algorithm. South West University, Chongqing (2006) (in Chinese)
5. Hendtlass, T.: A Combined Swarm Differential Evolution Algorithm for Optimization Problems. In: Monostori, L., Váncza, J., Ali, M. (eds.) IEA/AIE 2001. LNCS (LNAI), vol. 2070, pp. 11–18. Springer, Heidelberg (2001)
6. Parsopoulos, K.E., Vrahatis, M.N.: Initializing the Particle Swarm Optimizer Using the Nonlinear Simplex Method. In: Advances in Intelligent Systems, Fuzzy Systems, Evolutionary Computation, pp. 216–221. WSEAS Press (2002)
7. Miranda, V., Fonseca, N.: EPSO-Best-of-Two-Worlds Meta-Heuristic Applied to Power System Problems. In: Proceedings of the IEEE Congress on Evolutionary Computation, Honolulu, Hawaii, USA, pp. 1080–1085 (2002)

8. Krink, T., Lovbjerg, M.: The Life Cycle Model: Combining Particle Swarm Optimization, Genetic Algorithms and Hill Climbers. In: Proceedings of Parallel Problem Solving from Nature VII, pp. 621–630 (2002)
9. Shi, X., Lu, Y., Zhou, C., Lee, H., Lin, W., Liang, Y.: Hybrid Evolutionary Algorithms Based on PSO and GA. In: Proceedings of IEEE Congress on Evolutionary Computation (CEC), Canbella, Australia, pp. 2393–2399 (2003)
10. Noel, M.M., Jannett, T.C.: Simulation of a New Hybrid Particles Swarm Optimization Algorithm. In: Proceedings of the Thirty-Sixth Southeastern Symposium on System Theory, pp. 150–153 (2004)
11. Wachowiak, M.P., Smolfkova, R., Zheng, Y., Zurada, J.M., Elmaghraby, A.S.: An Approach to Multimodal Biomedical Image Registration Utilizing Particle Swarm Optimization. IEEE Transactions on Evolutionary Computation 8(3), 289–301 (2004)
12. Victoire, T.A.A., Jeyakumar, A.E.: Hybrid PSO-SQP for Economic Dispatch with Valve-point Effect. Electric Power Systems Research 71(1), 51–59 (2004)
13. Nelder, J.A., Mead, R.: A Simplex Method for Function Minimization. Computer Journal 7, 308–313 (1965)
14. Glover, F.: Future Paths for Integer Programming and Links to Artificial Intelligence. Computers and Operations Research 13, 533–549 (1986)
15. Wang, L.: Intelligence Optimization Algorithm and Its Application. Tsinghua University Press, Beijing (2001) (in Chinese)
16. Parsopoulos, K.E., Magoulas, V.P.G., Vrahatis, M.: Stretching Technique for Obtaining Global Minimizes through Particle Swarm Optimization. In: Proceedings of the workshop on particle swarm optimization, Indianapolis, IN (2001)
17. Parsopoulos, K.E., Vrahatis, M.N.: Modification of the Particle Swarm Optimizer for Locating All the Global Minima. In: Proceeding of the International Conference on Artificial Neural Networks and Genetic Algorithms, Prague, Czech Republic, pp. 324–327 (2001)
18. Riget, J., Vesterstroem, J.S.: A Diversity-guided Particle Swarm Optimizer – the ARPSO. Technical Report, Dept. of Computer Science, University of Aarhus, EVALife No.2002-02 (2002)