Yu. A. Benderuk; V. I. Mesyura, Cand. Sc. (Eng.), Assist. Prof.

CONSTANT PARAMETER SELECTION OF PARTICLE SWARM OPTIMIZATION METHOD IN PRODUCTION LOAD DISPATCH PROBLEM USING SIMULATED ANNEALING METHOD

This paper shows the solution of production load dispatch problem using particle swarm optimization method. It suggests to use the simulated annealing method for constant parameter selection of particle swarm optimization (socialization and personalization coefficients) during the solution of the production load dispatch problem. The use of the mentioned above approach allowed to achieve better results in comparison with classical particle swarm optimization method.

Keywords: *production load dispatch problem, particle swarm optimization, simulated annealing method***.**

Introduction

The present stage of science development gave rise to tasks of an applied character for which there are no efficient method for optimal solutions so far. For these tasks it is expedient to use the heuristic intelligent methods, which allow to find a solution, close to the optimal one within reasonable time.

The objective of the paper is to demonstrate the feasibility of using the particle swarm optimization method for the solution of the production load dispatch task as well as to demonstrate the advantages, given by the selection of the constant parameters of this intelligent method on the base of the other intelligent method – method of simulated annealing.

Formulation of the production load dispatch problem

The production load dispatch problem shall be formulated as follows: N factories are given, each of which produces a certain amount of some abstract material. The price of producing p_i number of material pieces by the i^{th} factory shall be determined by the formula (1):

$$f_i = A_i p_i^2 + B_i p_i + C_i; i = 1,2...N. \tag{1}$$

Production of material by each of the factory must satisfy the restrictions, presented in formula (2):

$$pMin_i \le p_i \le pMax_i; i = 1,2...N. \tag{2}$$

It is necessary to produce s units of material, minimizing the total spendings [1].

In the above formulas A_i, B_i, C_i, $pMin_i$, $pMax_i$ – some constants which characterize the i^{th} factory.

Particle swarm optimization method and its application to the task of production load dispatch problem

Particle swarm method is simulation intelligent method based on the imitation of birds or fish behavior during self-teaching [2].

At the beginning of the algorithm operation there shall be randomly generated the population of particles, each of which has speed, position in the space of solutions as well as the fitness function [3]. The speed and the position of the particle are vectors, the length of which coincides with the length of space for searching solutions.

The iteration process starts after the elementary initialization. At each iteration for each particle its speed will be recalculated. The formula for speed recalculation looks as follows:

$$v_i^{(j+1)} = v_i^{(j)} + c_1 * rand() * (pbest - current) + c_2 * rand() * (gbest - current); j = 1,2...t-1; i = 1,2...N \tag{3},$$

where $v_i^{(j)}$ is i^{th} coordinate value of a speed vector in j^{th} iteration, $v_i^{(j+1)}$ – same value at $(j+1)^{th}$ iteration, c_1 – significance of personal component, *rand()* – random value, equally distributed on the section [0;1], *pbest* – the best for this particle fitness function, which was achieved during the iteration process, *current* – its current fitness function, *gbest* – the best fitness function achieved during the iteration process among all population particles, c_2 – the value of social component, t – number of iterations. Constants c_1 and c_2 show how the particles orient on the own and global results achieved.

The position of the particle shall also be recalculated on the base of its changed speed. The rule of the position recalculation is the following:

$$p^{(j+1)} = p^{(j)} + v; j = 1,2...t-1 \tag{4}$$

The fitness function shall be recalculated after the estimation of the new position.

The result of the algorithm operation is the best (in terms of appropriate problem) fitness function achieved during the iteration process [4].

During the use of the particle swarm optimization method in the task of the distribution of manufacturing load, as the position of the particle it is appropriate to use vector, which contains information on the number of material, produced by each factory. The fitness function is the sum total for the production of the necessary quantity of the material.

Selection of the constant parameters of the particle swarm optimization method on the base of simulated annealing method

The significant drawback of the above-mentioned method is that the selection of the main constant parameters (namely – parameters of socialization and personalization) is not a trivial task for it. It is an obvious fact that the results of algorithm operation significantly depend on their selection.

Zhi-hui Zhan, Jun Zhang, Yun Li and H. S. H. Chung in [5] suggest to assign the personalization factor some high value and the socialization factor – some low value. Further on, they suggest to exponentially decrease the factor of personalization and increase the socialization factor correspondingly.

The other idea for changing the values of these factors is suggested in [6] by Asanga Ratnaweera, Saman K. Halgamug and Harry C. Watson. It implies for the change in each factor depending on the iteration number, total number of iterations and values the other two previously determined constants.

Supiya Ujjin and Peter. J. Bentley in [7] suggest another way for adaptation process of particle swarm optimization parameters. In the beginning of algorithm's runtime random values are assigned to socialization and personalization factors. During iterative process values of this factors converge to some constant value exponentially.

It is possible to conclude that the ideas of selection of values of socialization and personalization factors on the base of other intellectual methods is not given adequate attention in topical papers. The objective of this paper is to show the possibility and appropriateness of using such a combination of the two intelligent methods.

It is suggested to select the constant parameters for the particle swarm optimization by the simulated annealing method.

The simulated annealing method is an intelligent method, which imitates the physical process of metal crystallization during its cooling. At the beginning of algorithm operation there shall randomly be generated the initial solution and set the value of the initial temperature of the process. Further on, the iteration process takes place, during which the current solution shall be randomly changed. With the set probability there takes place transition from the current state to the generated state. This probability shall be calculated as follows:

$$P_j = \min(1, e^{-\frac{g(X_j)-f(X_j)}{T_j}}); j = 1,2..t-1, \tag{5}$$

where P_j – target value of probability at j^{th} iteration, $g(X_j)$ – value of the efficiency function in changed solution at j^{th} iteration, $f(X_j)$ – value of the efficiency function in the initial solution at j^{th} iteration, T_j – current temperature value.

Change of the temperature takes place upon each iteration. Let p – some constant, which belongs to the interval from zero to one. Then the temperature change shall be described by the law:

$$T_{j+1} = T_j p; j = 1,2..t-1 \tag{6}$$

The results of the algorithm operation– the best achieved value of target function.

It shall be suggested to select the values of the main constant parameters on the base of the simulated annealing method for each particle. The fitness function shall be the current value of the target function (in the task of distribution of the manufacturing load – the sum total of the material production). These parameters are initially equal the evenly distributed random value from the section [1; 2]. The transition from one state into the other occurs by adding the random evenly distributed value from the section [-0.05; 0.05] to the current value.

Comparison of the classical method of particle swarm optimization and the method for particle swarm optimization with the selection of constant parameters on the base of simulated annealing method

The comparison of genetic algorithm, classical partical swarm optimization, particle swarm optimization with the selection of constant parameters on the base of simulated annealing method and other three algorithms, in base of which lies adaptation of constant factors of particle swarm optimization (suggested by Zhi-hui Zhan and others, Asanga Ratnaweera and others, and Supiya Ujjin and others correspondingly). Was carried out the minber of itreration for all algorithms was equal to 200, and population size was equal to 100.

Let M be the size of population. Then complexity of genetic algorithm is $O(NtM^2)$. Other's algorithm complexity is $O(NtM)$.

Table 1

The results of the comparison of the classical method for the particle swarm optimization with the selection of the constant parameters on the base of the simulated annealing method

Sample size	Genetic algorithm results	Classic particle swarm method result	Result of particles swarm method with adaptation of parameters suggested by Zhi-hui Zhan and others	Result of particles swarm method with adaptation of parameters suggested by Asanga Ratnaweera and others	Result of particles swarm method, with adaptation of parameters, suggested by Supiya Ujjin and othes	Results of the particles swarm method with the selection of constant parameters on the base of simulation and annealing method
10	$1.0699*10^{13}$	$1.0679*10^{13}$	$1.0662*10^{13}$	$1.0661*10^{13}$	$1.0662*10^{13}$	$1.0658*10^{13}$
20	$1.0313*10^{13}$	$1.0195*10^{13}$	$1.0193*10^{13}$	$1.0195*10^{13}$	$1.0191*10^{13}$	$1.0189*10^{13}$
30	$9.7242*10^{12}$	$9.6936*10^{12}$	$9.6828*10^{13}$	$9.6826*10^{13}$	$9.6823*10^{13}$	$9.6823*10^{13}$
40	$1.1294*10^{13}$	$1.1102*10^{13}$	$1.1099*10^{13}$	$1.1109*10^{13}$	$1.1095*10^{13}$	$1.1093*10^{13}$
50	$1.1428*10^{13}$	$1.1419*10^{13}$	$1.1421*10^{13}$	$1.1425*10^{13}$	$1.1417*10^{13}$	$1.1413*10^{13}$
60	$1.1829*10^{13}$	$1.1557*10^{13}$	$1.1556*10^{13}$	$1.1552*10^{13}$	$1.1552*10^{13}$	$1.1551*10^{13}$
70	$1.1836*10^{13}$	$1.1757*10^{13}$	$1.1754*10^{13}$	$1.1753*10^{13}$	$1.1752*10^{13}$	$1.1751*10^{13}$
80	$1.1574*10^{13}$	$1.1497*10^{13}$	$1.1497*10^{13}$	$1.1498*10^{13}$	$1.1497*10^{13}$	$1.1491*10^{13}$
90	$1.1383*10^{13}$	$1.1271*10^{13}$	$1.1269*10^{13}$	$1.1273*10^{13}$	$1.1269*10^{13}$	$1.1265*10^{13}$
100	$1.0834*10^{13}$	$1,0832*10^{13}$	$1.0828*10^{13}$	$1.0829*10^{13}$	$1.0828*10^{13}$	$1.0827*10^{13}$

For the comparison there had been used the random sets of input data, each parameter of which – equally distributed random value. The number of factories, producing the material, belong to the section [0; 1000]. The parameters of each factory belonged to the section [0; 10000]. The total volume of the material had also been chosen randomly and in a way to preserve the correctness of input data. The algorithms were compared on samples with size 10, 20, 30 … 100 of the testing sets. Standard of comparison – the average value of the target function for all the tests of the samples. The results of the comparison are presented in table 1.

The table shows that the results of the particle swarm optimization method with the selection of constant parameters on the base of annealing simulation method are not worse than the results of competing algorithms on all the ten sets of the testing data. Such a result may be explained by the fact that the constant parameters are determined not randomly. With the help of simulated annealing method there had been conducted the ordering search for the values of such socialization and personalization factors, which approach the result of the algorithm operation to the minimum.

Conclusions

The paper showed the possibility to apply the particle swarm optimization method to the solution of tasks on the distribution of the manufacturing load. There had been suggested the method for the selection of constant parameters to the particle swarm optimization method on the base of the simulated annealing method. There had also been presented the obtained results. The obtained results testify that the suggested method is more efficient than the other concurrent methods.

REFERENCES

1. Leandro dos Santos Coelho. Solving production load dispatch problems in power systems using chaotic and Gaussian particle swarm optimization approaches / Leandro dos Santos Coelho, Chu-Sheng Lee // Electrical Power and Energy Systems. – December 2008. – № 30. – P. 297 – 307.

2. Девятков В. Системи штучного інтелекту / Володимир Девятков. – М.: Видавництво МГТУ ім. Баумана, 2001. – 352 с.

3. Trelea Ioan Cristian. The particle swarm optimization algorithm: convergence analysis and parameter selection / Ioan Cristian Trelea // Information Processing Letters. – March 2003. – № 85. – P. 317 – 325.

4. Рассел Стюарт. Штучний інтелект. Сучасний підхід / Стюарт Рассел, Пітер Норівг; пер. з англ. К. А. Птіцина. – М.: «Вільямс», 2006. – 1408 с.

5. Zhi-hui Zhan. Adaptive Particle Swarm Optimization / Zhi-hui Zhan, Jun Zhang // Systems, Man, and Cybernetics, Part B: Cybernetics. – December 2009. – Vol. 39. – № 6. – P. 1362 – 1381.

6. Ratnaweera A. Self-Organizing Hierarchical Particle Swarm Optimizer With Time-Varying Acceleration Coefficients / A. Ratnaweera, S. Halgamuge, H. C. Watson // Evolutionary Computation, IEEE Transactions. – June 2004. – Vol. 8. – № 3. – P. 240 – 255.

7. Supiya Ujjin. Particle Swarm Optimization Recommender System / Supiya Ujjin, Peter. J. Bentley // Swarm Intelligence Symposium, 2003. SIS '03. Proceedings of the 2003 IEEE. – April 2003. – P. 124 – 131.

Benderuk Yulia – Student, Institute of Information Technologies and Computing Engineering.

Mesyura Volodymyr – Cand. Sc. (Eng.), Assist. Professor, Professor Department of computer sciences.
Vinnytsia National Technical University.