

Adaptive modification of the particle swarm method based on dynamic correction of the trajectory of movement of individuals in the population

Yu.V. Minaeva

Senior Lecturer, Department of Computer Aided Design Systems and Information Systems
Voronezh State Technical University
Address: 14, Moscow Avenue, Voronezh, 394026, Russian Federation
E-mail: julia_min@mail.ru

Abstract

Evolutionary search methods are successfully used for different modeling and optimization tasks due to their universality and the relative simplicity of realization in practice. However, a significant problem of using them is related with premature convergence of the computational algorithm due to incomplete exploration of the search space. This happens when all particles come into space of the first found, perhaps local optimum and cannot get out of it. To solve the problem, it is necessary to develop control procedures correcting movements of the individuals in the population.

This paper proposes a particle swarm optimization adaptive modification, permitting dynamic changes to the particles' trajectory to find more promising locations. The method is based on the opportunity to change the displacement vector individually for each particle depending on previous iteration effectiveness. For this purpose, procedures of direction choice and dynamic change of particle movement free parameters are added in the proposed modification. As opposed to the canonic swarm algorithm version, where all individuals converge on one particle with the best value found, in the new modification each particle chooses its displacement direction independently and can change it if the direction will be identified as ineffective. This approach makes it possible to reduce the probability of premature convergence of the algorithm and to explore given search space better, all of which is especially important for the multimodal function with complex landscape. The proposed method was tested on the standard set of test functions for continuous optimization, and it showed high reliability with relatively small use of time and computer resources.

Key words: optimization, evolutionary algorithms, particle swarm optimization, premature convergence, adaptation of the algorithm, hybrid algorithm.

Citation: Minaeva Yu.V. (2016) Adaptive modification of the particle swarm method based on dynamic correction of the trajectory of movement of individuals in the population. *Business Informatics*, no. 4 (38), pp. 52–59. DOI: 10.17323/1998-0663.2016.4.52.59.

Introduction

In the process of design studies of complex technical and economic systems, we often see a task of an optimal choice of those internal characteristics of the system that describe its structure or behavior. These

tasks include formation of the production program of the enterprise, the choice of equipment and production technology, justification of layout diagrams, selection and assessment of risks of investment projects, and so on. To increase the effectiveness and speed of

implementation of procedures of optimal search, one develops a great number of methods, but almost all of them have restrictions related to the nature of the mathematical model of the system under study. From this perspective, the most universal are controlled search algorithms based on the processes of evolutionary development of biological populations. One of such method is the particle swarm optimization (PSO), which uses the model of behavior of complex self-organizing systems with social structure. Such systems consist of simple interacting agents, each of which behaves independently of the others, but as a consequence the behavior of the entire multiagent system turns out to be intelligent [1].

Potential solutions in the PSO are represented as a population of living organisms, each of which occupies a certain position inside the swarm. The reason for the existence of all organisms of the population consists in increasing of the degree of individual utility due to displacement within the location with the best values of the objective function. To achieve this purpose, the particles are constantly updating their coordinates, using both their own knowledge and the experience gained by the rest of organisms of the population [2].

Up to this moment, a great number of modifications of the canonical PSO has been developed, but many of them retain the disadvantages inherent in the original algorithm. One of the most promising directions of research in the field of evolutionary algorithms is the study of adaptive properties of the PSO, improvement of which will increase the efficiency and universality of the search procedures.

The purpose of this scientific paper consists in developing an adaptive modification of the PSO making it possible to carry out dynamic correction of the trajectory of particle movement for the purpose of more effective investigation of the targeted field of search. The proposed method is based on the possibility each particle has to choose the direction, the movement in which increases its usefulness to the population.

1. Classical particle swarm optimization

Let us suppose that our swarm consists of n particles. Each of the swarm particles at any moment can be described by its coordinates $x_i = \{x_{i1}, x_{i2}, \dots, x_{id}\}$ and velocity $v_i = \{v_{i1}, v_{i2}, \dots, v_{id}\}$, while i – number of particle ($i = 1, \dots, n$), and d – the search space dimension. In this case, the whole swarm of particles at the k -th instant

of time is characterized by the vector of coordinates $x_k = \{x_1, x_2, \dots, x_n\}$ and vector of velocities of all the particles $v_k = \{v_1, v_2, \dots, v_n\}$. In accordance with the canonical particle swarm optimization developed by Kennedy and Eberhart [1], iterations are performed in accordance with the following pattern:

$$v_{k+1} = \alpha v_k + \beta r_1 (p_k - x_k) + \gamma r_2 (g_k - x_k),$$

$$x_{k+1} = x_k + v_{k+1}.$$

Moreover, p_k and g_k are the coordinates of the best solution found by the particle itself and by all the swarm, respectively, α, β, γ are free parameters of the algorithm, and r_1 and r_2 are random numbers within the range $[0, 1]$. Coefficients α, β and γ determine the degree of influence of each of the three components on the particle velocity. Value α is responsible for the persistence of the particle movement. If value α is close to 1, then the particle continues its path, thus exploring all the search space. Otherwise, the particle tends to the best value (its own or the social one), and will be staying in the area around it. Value β reflects the influence of cognitive component, that is determination of the particle to go back to the “best” value, in terms of the objective function, found by it earlier. Value γ expresses the social component of velocity, that is the tendency of the particle displacement to the current best solution found by the remaining particles [2].

The disadvantages of the canonical method are as follows [2]:

- ◆ the possibility of the particle coordinates leaving the function tolerance range limits;
- ◆ premature convergence of the algorithm to the first extreme (generally local one) and impossibility of any further search.

2. Modifications of particle swarm optimization

To remedy the shortcomings of the method, a great number of modifications have been developed, some of which are aimed at improving the work of the entire algorithm as a whole, while the others are designed to solve problems of a particular class.

All the modifications developed can be assigned to one of the following groups:

- ◆ modification of the cognitive component;
- ◆ modification of the social component;
- ◆ selection of free parameters of the algorithm;
- ◆ hybridization of algorithms.

The most significant modifications of the cognitive component of the canonical algorithm comprise consideration in the velocity formula of not only the “positive” experience of the particle, but also the negative experience, that is, the desire to move away from the “bad” values of the objective function [3], as well as the possibility of forced displacement of the particle during prolonged stagnation of its coordinates [4].

Modifications of the social component suppose consideration of influence of not only the best solution at the given moment, but also the current values of the remaining particles. To this group can be assigned such algorithms as the fully informed PSO, in which the greatest influence on the particle movement is exerted by the particles with “good” values [5], and the PSO based on the ratio of “value-distance”, where the degree of influence of each particle depends on the proximity of its location [2].

Influence of the social component upon the effectiveness of the optimal search procedures is largely determined by the topological structure of the population, because the size of the sub-aggregate of particles with which each individual particle can share its experiences depends on precisely this characteristic [2]. Research studies on the effectiveness and convergence of the PSO and its modifications under various topological structures show that the topologies of weak cohesion of the particles, that is those with a few number of neighbors, allow more effective research of the search space, and reduce the likelihood of premature convergence of the algorithm [2, 5, 6].

The effectiveness and reliability of the PSO is largely dependent on observance of the correct balance between the stages of the search space research and location of the extremum. To regulate the interrelationships between these stages, one uses such free parameters of the algorithm as α , β and γ , for which various scientific studies suggest the use of both constant values and time-dependent ones. For example, for coefficient α there are developed the following patterns of increments of the coefficient values [7, 8]:

◇ linear:

$$\alpha_k = \alpha_{max} - \frac{(\alpha_{max} - \alpha_{min})}{T_{max}} k,$$

where α_{max} and α_{min} – permissible maximum and minimum values of the coefficient;

T_{max} – the maximum possible number of iterations;

◇ non-linear:

$$\alpha_k = \alpha_{min} + \frac{(\alpha_{max} - \alpha_{min})}{T_{max}^2} (T_{max} - k)^2.$$

◇ exponential:

$$\alpha_k = \alpha_{min} + (1 - \alpha_{min}) e^{-\lambda k},$$

where λ is a defined constant.

When using the linear and nonlinear patterns, pre-assignment of T_{max} is mandatory.

For coefficients β and γ it is also recommended to use both constant values [1, 2, 9] and time-dependent ones [9, 10].

Such binding of changes of the coefficients to the time of the algorithm execution can lead to insufficiently effective search for solutions, because it is impossible to precisely predict at which exact iteration the optimum will be detected and localized. To remedy this shortcoming, scientific papers [8, 9] propose adaptive modifications of the particle swarm optimization, making possible more objective control over the optimization process.

Scientific paper [9] proposes to divide the solution-achieving process into four stages, depending on the range of particles straggling: research investigation of the search area, optimum localization, stagnation, and leaving the state of stagnation. Each stage is characterized by a certain strategy for changing the coefficients. At the stage of research investigation of the search area, the coefficient β increases, and γ decreases; during the optimum localization β and γ are changing insignificantly; in case of stagnation, the coefficients are slightly increasing; and while leaving the state of stagnation β decreases, and γ increases. The authors suggest changing the persistence coefficient depending on the rate of change in the value of the objective function.

The particle swarm optimization, as well as other evolutionary algorithms, can be easily used as a part of hybrid-type circuits. For example, for determination of the new particle velocity, scientific papers [11–13] propose to use selection, cross-breeding and mutation operations taken from the genetic algorithm. Apart from that, to improve the quality of this method one can use local search and differential evolution [14]. Particle swarm optimization can be also used in combination with non-evolutionary algorithms or their component parts (for example, scientific paper [14] shows the use of operations of stretching, reflection and displacement from the Nelder-Mead method of deformed polygon [15]). Multi-swarm algorithms [16], consisting of

several swarms, each of which, in the general case, possesses its own set of parameters, also represent a special case of hybridization.

3. Particle swarm optimization with adaptation of the movement trajectory

Canonical particle swarm optimization assumes the rushing of all the particles to one center having currently the best value of the objective function. However, this process can lead, in the first place, to premature stagnation of the algorithm, and, in the second place, to departure of the particles from the advantageous locations and loss of values of functions therein.

To remedy the mentioned shortcomings, it is suggested to carry out a dynamic correction of the trajectory of movement of each particle by means of adaptive selection of direction and changes in the degree of influence of both its own and social experiences. For this purpose, the following procedures are added to the classic version of this method:

- ◆ procedure of selection by each particle x_i of its own "social leader" $x_j, j = 1, \dots, n$, with the help of tournament selection method borrowed from the genetic algorithm;
- ◆ procedure of correction of values of coefficients α, β, γ .

The algorithm for the dynamic adaptation of the particle swarm optimization includes the following steps:

1. Initialization of values of velocity, coordinates of the particles and the free coefficients

$$v_i^0 = \text{rnd}(D_v), x_i^0 = \text{rnd}(D_x),$$

$$p_i^0 = x_i^0, y_i = i,$$

$$\alpha_i^1 = \alpha_max, \beta_i^1 = \beta_max, \gamma_i^1 = \gamma_min,$$

where D_v, D_x – tolerance ranges of the velocities and coordinates of the particles;

y – an array containing numbers of those particles, towards which the social component of the particle velocity vector is directed.

There shall be preset the number of iteration $k = 1$.

2. Calculation of values of the objective function $f(x_i^{k-1})$ and updating of the list of the best values found for all particles

$$p_i^k = \begin{cases} x_i^{k-1} - \text{when } f(x_i^{k-1}) < f(p_i^{k-1}). \\ \text{otherwise, } p_i^{k-1} \end{cases}$$

3. In the event $k < 2$, then transfer to sub-paragraph 4; otherwise one performs the procedure of dynamic

adaptation of parameters of the particle movement, including the following two stages:

a) Selection of the direction of movement of the particle depending on changes in its usefulness by means of the selection tournament method:

$$y_i = \begin{cases} \arg \min(f(\tilde{X}_i)) - \text{when } f(x_i^k) > f(x_i^{k-1}) + \delta f, \\ y_i, - \text{otherwise} \end{cases}$$

where \tilde{X}_i – a random sub-aggregate of particles to be selected for particle $x_i, \tilde{X}_i \subseteq X$;

b) Correction of those coefficient that are responsible for the movement persistence and the degree of influence of the cognitive and social components of velocity:

$$\alpha_i^{k+1} = \alpha_i^k - \delta\alpha,$$

$$\beta_i^{k+1} = \beta_i^k - \delta\alpha,$$

$$\gamma_i^{k+1} = \gamma_i^k + \delta\alpha,$$

where $\delta\alpha$ – a correction to the values of coefficients α, β, γ , to be defined as follows:

$$\delta\alpha = \frac{f(x_i^{k-1}) - f(x_i^k)}{f(x_i^{k-1})},$$

To ensure that the algorithm works correctly, after recalculation of the coefficients one should carry out verification of the new values belonging to the permissible intervals:

$$z_j = \begin{cases} z_min_j, & \text{when } z_j < z_min_j, \\ z_j, & \text{when } z_min_j \leq z_j \leq z_max_j, \\ z_max_j, & \text{when } z_j > z_max_j, \end{cases}$$

where $z = \{\beta_i^k, \gamma_i^k, \alpha_i^k\}$ – an array of current values of the coefficients;

$$z_min = \{\alpha_min, \beta_min, \gamma_min\},$$

$z_max = \{\alpha_max, \beta_max, \gamma_max\}$ – arrays with predetermined minimum and maximum values for each coefficient, $j = 1, \dots, 3$.

4. Recalculation of velocity and coordinates of the particles:

$$v_i^k = \alpha_i^k v_i^{k-1} + \beta_i^k r_1 (p_i^{k-1} - x_i^{k-1}) + \gamma_i^k r_2 (x_{y_i}^{k-1} - x_i^{k-1}),$$

$$x_i^k = x_i^{k-1} + v_i^k$$

5. If the condition of halting algorithm $|f(x_i^k) - f(x_i^{k-1})| \leq \varepsilon$, is satisfied and ε – permissible error of calculations, then we see the end of work of the algorithm, otherwise there should be set the number of iteration number $k = k + 1$ and transfer to sub-paragraph 2.

Table 1.

A set of test functions to check the algorithm

Name	Formula	Interval
Spherical function	$F_1(x) = \sum_{i=1}^n x_i^2$	$x \in (-5, 5)$
Rosenbrock's function	$F_2(x) = \sum_{i=1}^{n-1} (100(x_{i+1} - x_i^2)^2 + (x_i - 1))$	$x \in (-2.5, 2.5)$
De Jong's function 2	$F_3(x) = \frac{-100}{100 \cdot (x_1^2 - x_2) + (1 - x_1)^2 + 1}$	$x \in (-5, 5)$
Rastrigin's function	$F_4(x) = 10n + \sum_{i=1}^n (x_i^2 - 10 \cos(2\pi x_i))$	$x \in (-5, 5)$

To assess the effectiveness of the proposed algorithm, let us compare its work with the canonical PSO, the fully informed particle swarm (FIPS) [5], and the adaptive PSO (APSO), proposed in [9]. Let us denominate the developed modification with the dynamic correction of the trajectory of the particle movement as TPSO. Experimental research study of the methods was carried out at the following parameters:

- ◆ for all the methods, there is set the size of population of 30 particles and a time restriction of 2000 iterations;

- ◆ for PSO, coefficient α decreases in a linear pattern from $\alpha_{max} = 0.9$ to $\alpha_{min} = 0.4$, at coefficients $\beta = \gamma = 1.49618$ [2];

- ◆ for FIPS $\alpha = 0.7298$, $\beta = \gamma = 2.05$ [5];

- ◆ for methods APSO and TPSO there are set boundaries $\alpha_{max} = 0.9$, $\alpha_{min} = 0.4$, $\beta_{max} = 2.5$, $\beta_{min} = 1.5$, $\gamma_{max} = 2.5$, $\gamma_{min} = 1.5$ [9].

The test functions used (Table 1) were taken from the recommended standard set of test tasks of continuous optimization [2], and they allow us to check the quality of searching the extrema for functions with a different topography of the search space. Two test functions (Rosenbrock and spherical) are unimodal, and the rest are complex multimodal ones. For all the functions, the dimension of the coordinate space $n = 10$.

To compare various versions of the PSO, one uses such criteria as the efficiency of search procedures, number of iterations spent on the search for an optimum, and solution achieving time. As a solution time, one takes the difference between the system time of the computer measured before the start of the calculations and after

their completion. Table 2 shows the average values of the given criteria. The best value in each line is highlighted in semi-bold.

Based on the results of the computing experiment, it can be concluded that the proposed algorithm has demonstrated on the set of test functions used high effectiveness for both unimodal and complex multimodal tasks. Moreover, to search for an extremum it took fewer iterations and consumed less time for the processor operation than the rest of the PSO modifications (on three of the four test functions). In accordance with its parameters, the proposed method is close to the APSO, however its advantage consists in a simpler practical implementation and less time required to search for solutions.

Conclusion

This scientific paper proposes a hybrid modification of PSO that makes it possible to carry out adaptive correction of the particle movement trajectory depending on the effectiveness of performance of the optimal search procedure at the previous iteration. Such a pattern of achieving a solution allows particles to move quickly into the most advantageous locations due to the dynamic changes in the degree of influence of the cognitive and social components of their velocity. The modification of the canonical method considered was tested on standard test tasks of continuous optimization. Based on the test results, it can be concluded that the implementation of dynamic correction of the particles' movement trajectory allows us to increase the effectiveness of the global optimum search process and to reduce the evolution time as compared with the existing modifications of the PSO. ■

Table 2.

Criteria of effectiveness of various versions of the PSO

Function	Criteria	Methods of solution			
		PSO	FIPS	APSO	TPSO
$F_1(x)$	Effectiveness	100%	100%	100%	100%
	Number of iterations	523.4	492.1	478.5	475.4
	Solution time	0.311	0.302	0.324	0.305
$F_2(x)$	Effectiveness	100%	100%	100%	100%
	Number of iterations	591.2	524.7	489.1	498.3
	Solution time	0.408	0.397	0.415	0.380
$F_3(x)$	Effectiveness	90.1%	96.5%	100%	100%
	Number of iterations	754.5	685.4	621.2	613.8
	Solution time	0.634	0.605	0.615	0.594
$F_4(x)$	Effectiveness	92.5%	97.1%	100%	100%
	Number of iterations	712.9	647.5	597.2	584.1
	Solution time	0.612	0.594	0.618	0.590
Average value	Effectiveness	95.7%	98.4%	100%	100%
	Number of iterations	645.4	587.4	546.5	542.9
	Solution time	0.491	0.474	0.493	0.467

References

- Kennedy J., Eberhart R.C. (1995) Particle swarm optimization. Proceedings of the *IEEE International Conference on Neural Networks (ICNN'95)*, 27 November – 01 December 1995, Perth, Australia, vol. 4, pp. 1942–1948.
- Karpenko A.P., Seliverstov A.P. (2010) Globalnaya bezuslovnaya optimizacia roem chastits na graficheskikh processorah arhitektury CUDA [Global unconstrained particle swarm optimization on graphic processors with CUDA architecture]. *Science and Education: Electronic Scientific and Technical Edition*, no. 4. Available at: <http://technomag.edu.ru/doc/142202.html> (accessed 01 July 2016) (in Russian).
- Yang C., Simon D. (2005) A new particle swarm optimization technique. Proceedings of the *18th International Conference on Systems Engineering (ICSEng'05)*, 16–18 August 2005, Las Vegas, USA, pp. 164–169.
- Xie X., Zhang W., Yang Z. (2002) Adaptive particle swarm optimization on individual level. Proceedings of the *6th International Conference on Signal Processing (ICSP'02)*, 26–30 August 2002, Beijing, China, pp. 1215–1218.
- Mendes R., Kennedy J., Neves J. (2004) The fully informed particle swarm: Simpler, maybe better. *IEEE Transactions on Evolutionary Computation*, vol. 8, no. 3, pp. 204–210.
- Kennedy J., Mendes R. (2002) Population structure and particle swarm performance. Proceedings of the *2002 Congress on Evolutionary Computation (CEC'02)*, 12–17 May 2002, Washington, USA, pp. 1671–1676.
- Parsopoulos K.E., Vrahatis M.N., Eds. (2010) *Particle swarm optimization and intelligence: Advances and applications*. N.Y.: IGI Global.
- Clerc M. *Particle swarm optimization*. London: ISTE, 2006.
- Zhan Z., Zhang J., Li Y., Chung H.S.H. (2009) *Adaptive particle swarm optimization*. *IEEE Transactions on Systems, Man, and Cybernetics. Part B*, vol. 39, no. 6, pp. 1362–1381.
- Ratnaweera A., Halgamuge S.K., Watson H.C. (2004) Self-organizing hierarchical particle swarm optimizer with time-varying acceleration coefficients. *IEEE Transactions on Evolutionary Computation*, vol. 8, no. 3, pp. 240–255.
- Angeline P.J. (1998) Using selection to improve particle swarm optimization. Proceedings of the *1998 IEEE International Conference on Evolutionary Computation (ICEC'98)*. 4–9 May 1998, Anchorage, Alaska, USA, pp. 84–89.
- Chen Y.P., Peng W.C., Jian M.C. (2007) Particle swarm optimization with recombination and dynamic linkage discovery. *IEEE Transactions on Systems, Man, and Cybernetics. Part B*, vol. 37, no. 6, pp. 1460–1470.

13. Andrews P.S. (2006) An investigation into mutation operators for particle swarm optimization. Proceedings of the *2006 IEEE Congress on Evolutionary Computation*, 16–21 July 2006, Vancouver, BC, Canada, pp. 1044–1051.
14. Liang J.J., Suganthan P.N. (2005) Dynamic multi-swarm particle swarm optimizer with local search. Proceedings of the *2005 IEEE Congress on Evolutionary Computation (CEC'05)*, 2–5 September 2005, Edinburgh, Scotland, pp. 522–528.
15. Gimmler J., Stützle T., Exner T.E. (2006) Hybrid particle swarm optimization: An examination of the influence of iterative improvement algorithms on performance. Proceedings of the *5th International Workshop "Ant Colony Optimization and Swarm Intelligence" (ANTS 2006)*, 4–7 September 2006, Brussels, Belgium, pp. 436–443.
16. Jordan J., Helwig S., Wanka R. (2008) Social interaction in particle swarm optimization, the ranked FIPS, and adaptive multi-swarms. Proceedings of the *10th Annual Conference on Genetic and Evolutionary Computation (GECCO'08)*, 12–16 July 2008, Atlanta, USA, pp. 49–56.

Адаптивная модификация метода роя частиц на основе динамической коррекции траектории движения особей в популяции

Ю.В. Минаева

*старший преподаватель кафедры систем автоматизированного проектирования и информационных систем
Воронежский государственный технический университет
Адрес: 394026, г. Воронеж, Московский пр-т, д. 14
E-mail: julia_min@mail.ru*

Аннотация

Методы эволюционного поиска успешно применяются для решения разнообразных задач оптимизации и моделирования ввиду своей универсальности и относительной простоты практической реализации. Однако большой проблемой при их использовании является преждевременная сходимость вычислительного алгоритма вследствие неполного исследования пространства поиска. Это происходит в том случае, когда все частицы попадают в область первого обнаруженного, возможно, локального оптимума, и не могут из нее выбраться. Для решения этой проблемы необходима разработка управляющих процедур, корректирующих перемещение особей в популяции.

В статье предлагается адаптивная модификация метода роя частиц, позволяющая осуществлять динамическое изменение траектории движения частиц для нахождения наиболее перспективных локаций. В основе метода лежит возможность изменения вектора перемещения индивидуально для каждой частицы, в зависимости от результативности выполнения предыдущей итерации. Для этого в предлагаемую модификацию канонического метода добавлены процедуры выбора направления и динамического изменения свободных параметров движения частицы. В отличие от канонической версии роевого алгоритма, в котором все особи популяции стремятся приблизиться к одной частице с наилучшим найденным значением, в новой модификации каждая частица самостоятельно выбирает направление движения и может изменить его в случае, если оно будет признано неэффективным. Такой подход позволяет снизить вероятность преждевременной сходимости алгоритма и лучше исследовать заданную область поиска, что особенно важно для многоэкстремальных функций со сложным рельефом. Предложенный метод был проверен на стандартном наборе тестовых функций непрерывной оптимизации и показал высокую эффективность при относительно небольших затратах времени и вычислительных ресурсов.

Ключевые слова: оптимизация, эволюционные алгоритмы, метод роя частиц, преждевременная сходимость, адаптация алгоритма, гибридизация алгоритмов.

Цитирование: Minaeva Yu.V. Adaptive modification of the particle swarm method based on dynamic correction of the trajectory of movement of individuals in the population // *Business Informatics*. 2016. No. 4 (38). P. 52–59. DOI: 10.17323/1998-0663.2016.4.52.59.

Литература

1. Kennedy J., Eberhart R.C. Particle swarm optimization // Proceedings of the IEEE International Conference on Neural Networks (ICNN'95), 27 November – 01 December 1995, Perth, Australia. Vol. 4. P. 1942–1948.
2. Карпенко А.П., Селиверстов А.П. Глобальная безусловная оптимизация роиом частиц на графических процессорах архитектуры CUDA // Наука и образование: Электронное научно-техническое издание. 2010. №4. [Электронный ресурс]: <http://technomag.edu.ru/doc/142202.html> (дата обращения: 01.07.2016).
3. Yang C., Simon D. A new particle swarm optimization technique // Proceedings of the 18th International Conference on Systems Engineering (ICSEng'05), 16–18 August 2005, Las Vegas, USA. P. 164–169.
4. Xie X., Zhang W., Yang Z. Adaptive particle swarm optimization on individual level // Proceedings of the 6th International Conference on Signal Processing (ICSP'02), 26–30 August 2002, Beijing, China. P. 1215–1218.
5. Mendes R., Kennedy J., Neves J. The fully informed particle swarm: Simpler, maybe better // IEEE Transactions on Evolutionary Computation. 2004. Vol. 8. No. 3. P. 204–210.
6. Kennedy J., Mendes R. Population structure and particle swarm performance // Proceedings of the 2002 Congress on Evolutionary Computation (CEC'02), 12–17 May 2002, Washington, USA. P. 1671–1676.
7. Parsopoulos K.E., Vrahatis M.N. (Eds.) Particle swarm optimization and intelligence: Advances and applications. N.Y.: IGI Global, 2010.
8. Clerc M. Particle swarm optimization. London: ISTE, 2006.
9. Zhan Z., Zhang J., Li Y., Chung H.S.H. Adaptive particle swarm optimization // IEEE Transactions on Systems, Man, and Cybernetics. Part B. 2009. Vol. 39. No. 6. P. 1362–1381.
10. Ratnaweera A., Halgamuge S.K., Watson H.C. Self-organizing hierarchical particle swarm optimizer with time-varying acceleration coefficients // IEEE Transactions on Evolutionary Computation. 2004. Vol. 8. No. 3. P. 240–255.
11. Angeline P.J. Using selection to improve particle swarm optimization // Proceedings of the 1998 IEEE International Conference on Evolutionary Computation (ICEC'98). 4–9 May 1998, Anchorage, Alaska, USA. P. 84–89.
12. Chen Y.P., Peng W.C., Jian M.C. Particle swarm optimization with recombination and dynamic linkage discovery // IEEE Transactions on Systems, Man, and Cybernetics. Part B. 2007. Vol. 37. No. 6. P. 1460–1470.
13. Andrews P.S. An investigation into mutation operators for particle swarm optimization // Proceedings of the 2006 IEEE Congress on Evolutionary Computation, 16–21 July 2006, Vancouver, BC, Canada. P. 1044–1051.
14. Liang J.J., Suganthan P.N. Dynamic multi-swarm particle swarm optimizer with local search // Proceedings of the 2005 IEEE Congress on Evolutionary Computation (CEC'05), 2–5 September 2005, Edinburgh, Scotland. P. 522–528.
15. Gimmmler J., Stützle T., Exner T.E. Hybrid particle swarm optimization: An examination of the influence of iterative improvement algorithms on performance // Proceedings of the 5th International Workshop “Ant Colony Optimization and Swarm Intelligence” (ANTS 2006), 4–7 September 2006, Brussels, Belgium. P. 436–443.
16. Jordan J., Helwig S., Wanka R. Social interaction in particle swarm optimization, the ranked FIPS, and adaptive multi-swarms // Proceedings of the 10th Annual Conference on Genetic and Evolutionary Computation (GECCO'08), 12–16 July 2008, Atlanta, USA. P. 49–56.