Bożena BOROWSKA

EFFECTIVENESS OF THE MPSO ALGORITHM IN OPTIMIZATION OF THE COIL ARRANGEMENT

ABSTRACT *One of the most important problems in designing of various constructions is optimization of technical facilities. The optimization process leads to find the best solution of a considered problem, and the solution should meet established criteria. Evolutionary algorithms have been found to be effective in solving such optimization problems.*

In the following paper, a modification of the PSO algorithm has been proposed in order to determine an optimal geometry of the coil arrangement evoking, in a defined active area, magnetic field of the largest possible gradient, and simultaneously keep this gradient relatively stable. The computations confirmed high efficiency of the proposed method. The results were also compared with the achievements of other evolutionary algorithms.

Keywords: *Electromagnetism, optimization, Particle Swarm Optimization, magnetic field*

1. INTRODUCTION

Optimization of technical devices is one of the most important problems in designing of various constructions. Its objective is to find the best solution

dr inż. Bożena BOROWSKA
e-mail: bozena@ics.p.lodz.pl

Instytut Informatyki,
Politechnika Łódzka

of the considered problem, and this solution should meet the established criteria. In fact, in many cases these requirements are at variance with each other, which brings about a necessity to achieve some specific compromise.

The above-mentioned problems are solved using among other things evolutionary algorithms such as genetic algorithms (GA) [6], genetic programming (GP) [14] or algorithms based of swarm intelligence [11]. In the following article, a particle swarm optimization algorithm (PSO) has been considered. It has gaining an increasing popularity due to its simplicity, and effectiveness in performing difficult optimization tasks. It has been used to solve a wide range of optimization problems [2, 7, 12, 13, 22]. Applications of the PSO method in electromagnetism are also known [15, 17, 23]. In comparison with other optimization algorithms, PSO is taken into account as a powerful technique in solving various kinds of complex, nonlinear optimization problems. Distinguished from genetic algorithms (GA), PSO does not need complex encoding and decoding processes and special genetic operators like crossover and mutation. The advantages of particle swarm optimization are that PSO is easy to implement and there are fewer parameters to adjust. Moreover PSO not only has a better response but also converges very quickly in comparison with ordinary evolutionary methods [19]. More information about advantages and drawbacks of PSO compared with other evolutionary algorithms can be found in [20, 21].

However, in order to find the optimal solution it is often necessary to make several numerical computations. The more complex the fitness function is, the longer computation time needs to be applied.

In order to improve the efficiency of the algorithm, and simultaneously to reduce the number of computations, in the following article a modification of the PSO algorithm was applied. The effectiveness of the proposed method was then confirmed by solving the problem that relied on determination of the optimal geometry of the coil arrangement generating magnetic field with the specific parameters. The results were then compared with performances of other evolutionary algorithms.

The rest of this paper is organized as follows. Section 2 defines problem used for comparison of the algorithms. Section 3 describes the standard PSO. Section 4 describes the MPSO algorithm. Section 5 presents the results followed by conclusions in section 6.

2. THE OPTIMIZATION PROBLEM

The coil arrangement evoking magnetic field with the defined distribution described in [1] has been considered. The coils form a cylindrical symmetry configuration. The xy-plane represents a symmetry plane, whereas the z-axis is a symmetry axis of the coils. The magnetic field along the z-axis is described [3] as follows:

$$H\ (0,0,z) = \frac{J_0}{2} \left\langle \begin{array}{l} \left\{ \begin{array}{l} (a-z-Z_0)\ln\left[\dfrac{(R_0+b)+((R_0+b)^2+(a-z-Z_0)^2)^{1/2}}{(R_0-b)+((R_0-b)^2+(a-z-Z_0)^2)^{1/2}}\right] + \\ (a+z+Z_0)\ln\left[\dfrac{(R_0+b)+((R_0+b)^2+(a+z+Z_0)^2)^{1/2}}{(R_0-b)+((R_0-b)^2+(a+z+Z_0)^2)^{1/2}}\right] \end{array} \right\} - \\ \left\{ \begin{array}{l} (a-z+Z_0)\ln\left[\dfrac{(R_0+b)+((R_0+b)^2+(a-z+Z_0)^2)^{1/2}}{(R_0-b)+((R_0-b)^2+(a-z+Z_0)^2)^{1/2}}\right] + \\ (a+z-Z_0)\ln\left[\dfrac{(R_0+b)+((R_0+b)^2+(a+z-Z_0)^2)^{1/2}}{(R_0-b)+((R_0-b)^2+(a+z-Z_0)^2)^{1/2}}\right] \end{array} \right\} \end{array} \right\rangle \tag{1}$$

The computation methods applied for magnetic field evoked by cylindrical symmetry coils have been presented in [3, 5, 16].

In the considered arrangement, the coil cross section sides are 2a and 2b respectively. The J_0 parameter represents density of the current flowing within the coil. For the purpose of the following research, it was established as J_0 = 250 A/m^2. The R_0 parameter in the arrangement describes the average radius of the coils. Furthermore, it was assumed that the distance between the coils and the x-axis is Z_0. The cross section of the considered arrangement is shown in Figure 1.

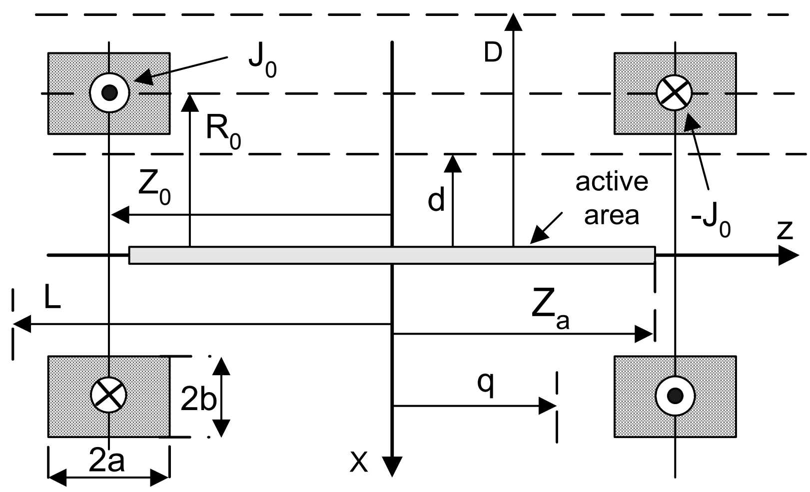

Fig. 1. The cross section of the coil arrangement generating magnetic field with the controlled gradient

For such the determined system, and on the assumption that the currents of the opposite direction flow through the coils, the magnetic field in the centerline of the coil arrangement symmetry is 0. The following geometrical constrains were introduced: $R_0 + b \leq D$, $R_0 - d \geq b$, $Z_0 - q \geq a$, and $L \geq Z_0$. The inequality $abR_0 \leq 0.006$ m^3 determines the maximal amount of material needed to make the coils. Moreover, for the purpose of the investigation, the following values of other parameters were used: $D = 0.6$ m, $d = 0.25$ m, $Z_a = 0.7$ m, $q = 0.4$ m, and $L < 1$ m. We wish to find the values of a, b, R_0, and Z_0 in such a manner to obtain the largest possible gradient of the magnetic field in an active area $2Z_a$, and to keep simultaneously the maximal possible linearity of this gradient. For such the defined problem, we define the fitness function Fun as described below:

$$Fun = \frac{1000\ G}{\left|H(z = Z_a/2)\right|^k} = \frac{1000\left[\left(\frac{4}{3}\left|H(z=0.75Z_a)\right| - \left|H(z=Z_a)\right|\right)^2 + 2\left(2\left|H(z=Z_a/4)\right| - \left|H(z=Z_a/2)\right|\right)^2\right]^{1\backslash 2}}{\left|H(z=Z_a/2)\right|^k \left|H(z=Z_a)\right|} \tag{2}$$

The factor G represents the field gradient stability, which is equalled to zero when the field gradient in the active area is constant. The fitness function Fun denominator is proportional to the field gradient in the active area. A minimization of the fitness function relies on decreasing of the numerator and increasing of the denominator simultaneously. The factor $k = 0.15$ determines the priority of the field gradient magnitude with reference to its linearity.

3. THE STANDARD PSO ALGORITHM

Particle swarm optimization is a search method whose mechanics was inspired by the social behavior of a bird flocking and swarm of bees. It was firstly introduced by Kennedy and Eberhart in 1995 [9, 10]. The PSO conducts searches using a population (swarm) of individuals named particles. Each particle represents a possible solution to the optimization task. For each particle the position vector and the velocity vector (v) are randomly generated for every dimension.

During every iteration, each particle is accelerated in the direction of its own personal best position, named *pbest*, found so far as well as in the direction of the global best position, named *gbest*, discovered so far by any of the particles in the whole swarm. The fitness of each particle is evaluated according to the fitness function of the optimization problem [18]. At every iteration, the velocity vector (v) of each particle in the swarm is updated using the following equation [11]:

$$v^i_{j+1} = wv^i_j + c_1 r_1 (pbest^i - x^i_j) + c_2 r_2 (gbest - x^i_j) \tag{3}$$

where *w* is called inertia weight that determines the impact of previous velocity of particle on its current one, c_1 and c_2 are acceleration factors that determine how much the particle is influenced by the memory of its best location and by the rest of the swarm respectively, whereas r_1 and r_2 represent randomly generated numbers in the range (0,1).

The new particle location is a function of a newly determined velocity and its previous position according to the following formula:

$$x^i_{j+1} = x^i_j + v^i_{j+1} \tag{4}$$

where x^i_{j+1} and x^i_j represent the current and previous positions of particle i respectively. According to the update equations, at every iteration particles will gradually move closer and closer to the global best position.

4. MODIFICATION OF THE PSO ALGORITHM

In order to improve the effectiveness of the PSO algorithm, a number of modifications towards the standard version were introduced. Those improvements concern both the way of the search space exploration and the way by which the swarm collects information. The essential modification relies on a rebuilt of the velocity vector updating equation to which an additional element was introduced.

In most publications, a global version of the PSO algorithm has been applied in which the last element of the updating equation (3) contains information about the position of the best fitted particle within the whole swarm. In contrast to the global version, the last element of the equation in the local neighbourhood version contains information about neighbours of a given particle [8]. In the paper [4] both elements mentioned above are taken into account.

In the following study, the global version of the velocity increment updating equation (3) was extended with an additional element that includes information of both a considered particle and the best fitted particle.

$$v_{j+1}^{i} = wv_{j}^{i} + c_1 r_1 (pbest^{i} - x_{j}^{i}) + c_2 r_2 (gbest - x_{j}^{i}) + c_3 r_3 (\alpha(gbest + pbest) - x_{j}^{i}) \quad (5)$$

The factor α determines the priority of the introduced information. r_3 similarly to r_1 and r_2 (see above) represent randomly generated number in the range (0,1). c_3 similarly to c_1 and c_2 (see above) is an acceleration factor.

5. RESULTS

The study on the effectiveness of the proposed method used to determine an optimal geometry of the coil arrangement was undertaken by means of a computer program written in Mathematica. The searching for optimal solutions was performed with the accuracy up to 0.001. The inertial weight value was fixed to w = 0.9 with a linear decrease to 0. The value of α was experimentally established to α = 0.5. The results were then compared with the achievements of the standard PSO and the θ-PSO algorithms described in [24]. The computations were made for 1000 iterations. In each case, the algorithms started from the areas of allowable solutions. The particles that did not meet those constraints were not evaluated.

The exemplary results of the test performed for 20, and 40 particles in the initial population are shown in Fig 2 and Fig 3. All the values were averaged over 50 trials for each combination of the parameters.

The average number of iterations needed to achieve the accurate solutions in relation to the number of particles and the algorithm used is depicted in table 1.

The results confirmed the effectiveness of the proposed MPSO algorithm. Compared with the other algorithms used for the investigation, the new MPSO algorithm turned out to be the most efficient since it could find more accurate solutions within a significantly lower number of iterations needed to achieve that. The best results were obtained for the swarms comprising of 40 particles.

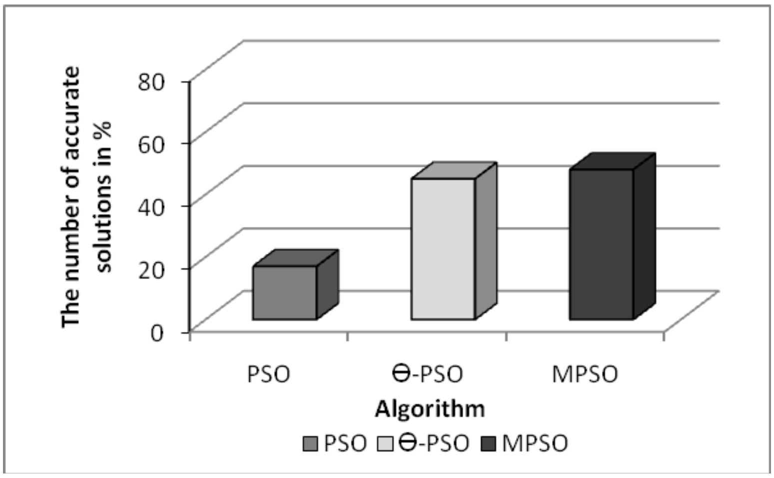

Fig. 2. The number of accurate solutions for 20 particles in the swarm, and for various algorithms

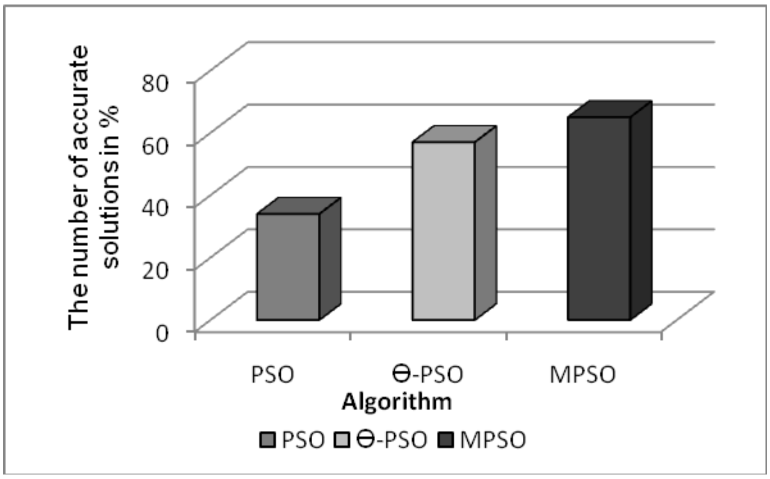

Fig. 3. The number of accurate solutions for 40 particles in the swarm, and for various algorithms

TABLE 1
The relationship between the population cardinality and the number of iterations to achieve the accurate solutions for various algorithms

The number of particles in the swarm	**Algorithm**		
	PSO	***ϴ*-PSO**	**MPSO**
20	607	123	104
40	576	94	79

In comparison with standard PSO, the new algorithm was able to find more accurate solutions within a few times lower iteration number (table 1). For the population comprising 40 particles, the number of accurate solution found by means of MPSO was half as much again than the number of accurate solutions obtained by PSO, and of 12% larger that the number of accurate solutions found by means of θ-PSO.

Moreover, for the population comprising 20 particles in the swarm, the number of accurate solutions found by PSO was lower than in case of the number of accurate solutions found by the new MPSO algorithm. In relation to θ-PSO, the MPSO algorithm was able to find more accurate solutions of only 2%, but the number of iterations was of even 24% lower.

With respect to the accuracy of the results, the PSO algorithm turned out to be the least effective, in comparison to the other algorithms. For the population comprising 20 particles in the swarm, the PSO algorithm, in over 85% cases, was not able to find accurate solutions with the established accuracy (up to 0.001) within 1000 iterations. An increase in the number of particles up to 40 resulted in a slight improvement of the PSO algorithm performance. However, in over 70% cases, the results with the required accuracy were not found. The θ-PSO algorithm was giving more results with the desired accuracy than standard PSO, but up to 18% more iterations needed to be performed than in case of the MPSO algorithm.

6. SUMMARY

In the following study, the new MPSO algorithm for optimization of constructions was proposed. This algorithm makes up a modification of standard PSO. The improvements include both the way of the search space exploration and the way by which the swarm gains information. The algorithm was applied for optimization of the coil arrangement geometry evoking magnetic field with specific parameters. The results have proved high efficiency of the proposed algorithm both in terms of the convergence velocity and the capability of finding global optimum.

LITERATURE

1. Borowska B., Nadolski S.: Particle Swarm Optimization: The Gradient Correction, Journal of Applied Computer Science, vol. 18, no. 2, pp. 7-15, 2009.

2. Cui Z.H., Zeng J.C., Sun G.J.: A fast particle swarm optimization, International Journal of Innovative Computing, Information and Control, vol. 6, pp. 1365-1380, 2006.
3. Forbes L.K., Crozier S., Doddrell M.: *Rapid* Computation of static Fields Produced by Thick circular solenoids, IEEE Transaction on Magnetics, vol. 33, pp. 4405-4410, 1997.
4. Foryś P.: Numerical optimization with particle swarms, Proceedings of the 10th International Conference on Numerical Methods in Continuum Mechanics, Zilina, 2005.
5. Garrett M. W.: Axially symmetric systems for generating and measuring magnetic fields. Part 1, Journal of Applied Physics, vol. 22, no. 9, pp. 1091-1107, 1951.
6. Goldberg D.E.: Algorytmy genetyczne i ich zastosowania, Wydawnictwo Naukowo-Techniczne, Warszawa, 1998.
7. Hassas S.: Using swarm intelligence for dynamic web content organizing, IEEE Swarm Intelligence Symposium, Indianapolis, USA, pp. 19-25, 2003.
8. Hu X., Eberhart R.C, Shi Y.: Engineering optimization with Particle Swarm, Proceedings of the 2003 IEEE on Swarm Intelligence Symposium, pp. 53 - 57, 2003.
9. Kennedy J. Eberhart R.C.: Particle swarm optimization, IEEE Internat. Conf. on Neural Networks, Perth, Australia, pp.1942-1948, 1995.
10. Kennedy J., Eberhart R.C.: A new optimizer using particle swarm theory, 6th International Symposium on Micromachine and Human Science, Nagoya, Japan, pp. 39-43, 1995.
11. Kennedy J, Eberhart R. C.: Swarm Intelligence*,* Morgan Kaufmann Publishers, Inc., San Francisco, 2001.
12. Mendes R., Cortez P., Rocha M., Neves J.: Particle swarm for feedforward neural network training, International Joint Conference on Neural Networks, pp.1895-1899, 2002.
13. Parsepolous K.E., Papageorgiou E.I., Groumpos P.P., Vrahatis M.N.: A first study of fuzzy cognitivemaps learning using particle swarm optimization, IEEE Congress on Evolutionary Computation 2003, pp. 1440-1447, Australia, 2003.
14. Riolo R., Soule T., Worzel B.: Genetic programming theory and practice, Springer-Verlag, USA, 2007.
15. Robinson J., Rahmat-Samii Y., Particle Swarm Optimization in Electromagnetics, IEEE Trans. Anten. and Propag., vol.52, pp. 397-407, 2004.
16. Romeo F., Hoult D. I.: Magnet field profiling: Analysis and correcting coil design, Magnetic Resonance in Medicine, vol. 1, no. 1, pp. 44 - 65, 1984.
17. Sato S., Sakaguchi S., Futamata K. and Katou K.: Coil Optimization for Homogeneous Magnetic Field with Small Leakage Field, IEEE Trans. Magn., vol.36, pp. 649-653, 2000.
18. Shi Y., Eberhart R.C.: A modified particle swarm optimizer, IEEE World Congress on Computational Intelligence, pp. 69-73, 1998.
19. Shi Y., Eberhart R.C.: Particle Swarm Optimization: Developments, Applications and Resources, Proceedings of the 2001 Congress on Evolutionary Computation, vol. 1, pp. 81-86, 2001.
20. Shi, Y.; Eberhart, R.C.; Empirical Study of Particle Swarm Optimization, Proceedings of the Congress on Evolutionary Computation, vol. 3, pp. 1945-1950, 1999.
21. Spears W.M., Kennedy J.: Matching algorithms to problems: an experimental test of the particle swarm and some genetic algorithms on the multimodal problem generator, Proceedings of the IEEE International Conference on Evolutionary Computation, pp. 78-83, Anchorage, AK, 1998.
22. Venayagamoorthy G.K., Doctor S.: Navigation of mobile sensor using PSO and embedded PSO in a fuzzy logic controller, 39th IEEE IAS Annual Meeting on Indus. Applic., pp.1200-1206, Seattle, USA, 2004.

23. Zainun-Deen S. H., Mahmoud K. R., El-Adawy M., Ibrahem S. M. M.: Design of Yagi-Uda antenna and electromagnetically coupled curl antenna using particle swarm optimization algorithm, Twenty Second National Radio Science Conference, Egypt, 2005.

24. Zhong W., Li S., Qian F.: θ-PSO: a new strategy of particle swarm optimization, Journal of Zhejiang University-Science A, vol. 9, pp. 786-790, 2008.

Manuscript submitted 11.05.2010
Reviewed by Prof. Jacek Sosnowski

SKUTECZNOŚĆ ALGORYTMU MPSO W OPTYMALIZACJI UKŁADU CEWEK

Bożena BOROWSKA

STRESZCZENIE *Jednym z najważniejszych zagadnień w projektowaniu różnych konstrukcji jest optymalizacja urządzeń technicznych. Jej celem jest znalezienie najlepszego rozwiązania rozpatrywanego zagadnienia o najlepszych w sensie przyjętych kryteriów parametrach. Do rozwiązywania tego typu zadań m.in. stosuje się algorytmy ewolucyjne. Aby algorytm był skuteczny często niezbędne jest jednak przeprowadzenie bardzo dużej liczby obliczeń.*

W celu redukcji kosztów obliczeń w artykule zaproponowano algorytm MPSO będący modyfikacją algorytmu PSO do problemu wyznaczenia optymalnej konstrukcji. Zadaniem zaproponowanego algorytmu było wyznaczenie optymalnej geometrii układu cewek generujących w zdefiniowanym obszarze aktywnym pola magnetycznego o możliwie dużym gradiencie przy zachowaniu jak największej stałości tego gradientu. Na podstawie przeprowadzonych badań, dokonano porównania efektywności zaproponowanej metody MPSO z osiągnięciami standardowego algorytmu optymalizacji cząsteczkowej PSO oraz algorytmu θ-PSO zaproponowanego przez Zhong i innych [24]. Przeprowadzone obliczenia potwierdziły skuteczność algorytmu MPSO.

Dr inż. Bożena BOROWSKA, ukończyła studia informatyczne na Politechnice Łódzkiej. Doktorat uzyskała w 2010 roku. Obecnie jest pracownikiem Instytutu Informatyki Politechniki Łódzkiej.