Bożena BOROWSKA

Technical University of Lodz, Institute of Information Technology

PAPSO algorithm for optimization of the coil arrangement

Abstract. *The PSO is a random algorithm for solving complex optimization problems. However, in order to provide this algorithm to be efficient, it is often necessary to make several numerical computations. To reduce the costs of computations, the new PAPSO algorithm was proposed. The presented algorithm was examined on four test functions then it was applied for determination of the optimal geometry of the coil arrangement.*

Streszczenie. *W artykule zaproponowano algorytm optymalizacji rojem cząstek z zastosowaniem wektora kąta fazowego (PAPSO). Zaproponowany algorytm przetestowano za pomocą czterech funkcji testowych, a następnie zastosowano do rozwiązania zadania polegającego na wyznaczeniu optymalnej geometrii układu cewek. (**Algorytm PAPSO do optymalizacji układu cewek**).*

Keywords: optimization, electromagnetism, particle swarm optimization, magnetic field.
Słowa kluczowe: optymalizacja, elektromagnetyzm, pole magnetyczne.

Introduction

Particle swarm optimization (PSO) is a stochastic, computational technique inspired by the social behavior of a swarm of bees firstly introduced by Kennedy and Eberhart in 1995 [1,2]. It has been applied successfully in many application areas including artificial neural network training, fuzzy system control dynamic web organizing and parameter selection [3, 4, 5, 6, 7]. The PSO algorithms have also found an application for solving several optimization problems in electromagnetism [8, 9].

In the case of traditional optimization methods, the PSO algorithm is characterized by a high efficiency in solving complex global optimization problems. However, in order to provide the algorithm to be efficient, it is often necessary to make several numerical computations of the optimized function, particularly in case of multimodal ones with numerous local minima determined in a multidimensional search space. To obtain a required accuracy in such cases, a considerable number of iterations is needed together with an extended swarm cardinality, and therefore more computations of the optimized function are required. Each determination of the optimization function value needs specific computational expenditures, which are higher when the fitness function is more complex. Due to large computational expenditures, a limitation in the number of the fitness function calculations is therefore required together with no loss of the algorithm efficiency as well as the results accuracy.

In order to reduce the costs of computations, the new Phase Angle Particle Swarm Optimization (PAPSO) algorithm with the phase angle vector was introduced. This algorithm represents the development of the concept proposed by Zhong and Qian [10].

In the standard PSO algorithm [11, 12] a potential solution to a problem represents a swarm of random particles, each of which has its own position and velocity vector. For each particle, in the following iterations, a new velocity vector responsible for the particle movement within the search space is calculated, and this vector takes part in establishing of the particle new position.

In the θ-PSO algorithm [10], determination of the particle velocity was omitted. Instead of the velocity vector, the phase angle vector was introduced. The formula of the phase angle increment update is responsible for the particle movement within the search space. The particle position is determined by the mapping of the corresponding angles. A detailed description of θ-PSO can be found in [10].

In the new PAPSO algorithm a number of significant modifications was introduced. The modifications concern both operation mechanism, updating equations as well as the way the particle move and gain the information.

The Optimization Problem

The proposed PAPSO algorithm was tested on two kinds of optimization problems. First it was examined on benchmark functions [13, 14]. Then it was applied for determination of the optimal geometry of the cylindrical coil arrangement described in [15].

Benchmark Test Functions

The effectiveness of the proposed PAPSO algorithm was examined on four benchmark functions [13, 14] and the results were then compared with performance of θ-PSO and standard PSO algorithm. The fundamental information of the functions used for the investigation is depicted in Table1.

Table 1. Optimization test functions

Function	Dimension	Range of x	Optimal f	Accuracy
Camel	2	(-100,100)	-1,0316	0,0001
Sphere	30	(-100,100)	0	0,0001
Griewank	30	(-600,600)	0	0,1
Rosenbrock	30	(-30,30)	0	100

Coil Arrangement

The optimization problem relies on the determination of the optimal geometry of the coil arrangement generating magnetic field with the specific parameters [15]. The considered set consists of two identical coils that form the cylindrical coil arrangement.

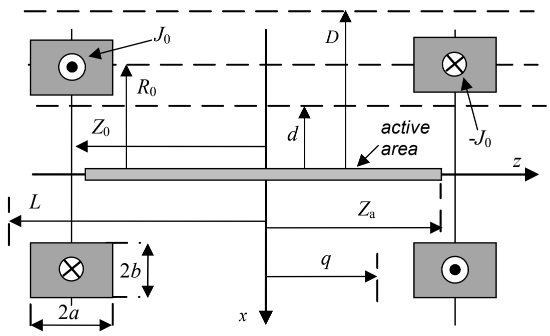

Fig.1. The cross section of the coil arrangement generating magnetic field with the controlled gradient

We consider the distribution of the magnetic field acting along the z-axis of the Cartesian coordinate system. The xy-plane represents a symmetry plane, and the distance between those coils is $2Z_0$. The average radius of the coil is R_0, and the sides of the coil cross section are $2a$ and $2b$ respectively (Fig.1). The J_0 parameter represents density of the current flowing within the coil. If the currents of the opposite directions flow through the coils, the magnetic field in the centerline of the coil arrangement symmetry is $H(0,0,0) = 0$. According to [16], the magnetic field along the z-axis is as follows:

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

For the purpose of the investigation, we assumed that J_0 = 250 A/m^2, Z_a=0.7m, D=0.6 m, d=0.25m, q=0.4m, L=1m and additionally introduced geometrical constraints as follows: $R_0+b\leq D$, $R_0-b\geq d$, $Z_0-a\geq q$, $Z_0\leq L$, and $a\cdot b\cdot R_0\leq 0.006$m^3. For these assumptions, the a, b, R_0, Z_0 parameters are determined in such a manner to obtain the largest possible gradient of the magnetic field in an active area Z_a and to maintain simultaneously the maximal possible linearity of this gradient. The fitness function is defined as follows:

(2)
$$F=\frac{100\left[\left(\frac{4}{3}\left|H(z=0.75Z_a)\right|-\left|H(z=Z_a)\right|\right)^2+2\left(2\left|H(z=Z_a/4)\right|-\left|H(z=Z_a/2)\right|\right)^2\right]^{\frac{1}{2}}}{\left|H(z=Z_a/2)\right|^k\left|H(z=Z_a)\right|}$$

The factor k=0.15 determines the priority of the field gradient magnitude with reference to its linearity.

Phase Angle Particle Swarm Optimization algorithm

In this method, a number of significant modifications and extensions in comparison to the original version was introduced. The operation mechanism and updating equations were rebuilt. The introduced changes concern both the way of the particle move and the way of the information collecting by the swarm during searching for the optimal solution. This helps the algorithm to explore the search space more efficiently. The PAPSO algorithm operation proceeds as follows:

1. Initialization, which relies on random attribution of the φ angle to each particles.
2. Establishing the particle position according to the formula [10]:

(3)
$$x_j^i=f(\varphi_j^i)=\frac{x_{max}-x_{min}}{2}\cdot\sin(\varphi_j^i)+\frac{x_{max}+x_{min}}{2}$$

3. Evaluation of the particle position by means of the fitness function
4. Reduction of the swarm cardinality by comparing the successive particles and the selection of better fitted particle (between two neighboured ones).
5. Establishing the best phase angle φ_{pb}, for each particle, by which the particle has managed to achieve its best position.
6. The choice of the particle of the φ_{gb} angle, by which the particle has achieved its best position within the whole swarm.
7. Updating the phase angle increment vector of each particle within the swarm according to the formula:

(4)
$$\begin{aligned}\Delta\varphi_{j+1}^i&=w\Delta\varphi_j^i+\sin(r_1)c_1(\varphi_{pb}^i-\varphi_j^i)\\&+\sin(r_2)c_2(\varphi_{gb}-\varphi_j^i)+c_3(\sqrt{3}(\varphi_{gb}+\varphi_{pb}^i)/4-\varphi_j^i)\end{aligned}$$

8. Assigning two new phase angles to each particle, ($u\in[-1;1]$) according to the formulae:

(5)
$$\varphi_{j+1}^{i1}=\varphi_j^i+\Delta\varphi_{j+1}^i$$

(6)
$$\varphi_{j+1}^{i2}=\varphi_j^i+u(-\Delta\varphi_{j+1}^i)$$

In this way, each particle with two new phase angles will obtain two proposals of its position.

9. Updating the particle location using (3) equation.
10. Evaluation of the particle position by means of the fitness function.
11. Comparison and selection of the new phase angle φ for each particle.
12. Repetition of 5-11 steps until the algorithm stopping criterion is met.

Where:

x^i_j -the position vector of the i–th particle in the j-th iteration,

$\Delta\varphi^i_{j+1}$ -the phase angle increment of the i–th particle,

φ^i_{pb} -phase angle of the personal best solution of the i-th particle,

φ_{gb} -the phase angle of the global best solution,

w -the inertial weight that determines the deviation of the particle original movement direction,

c_1, c_2, c_3 -the acceleration factor that determines how much the particle is influenced by its best phase angle and how much the particle is influenced by the whole swarm,

r_1, r_2 -the randomly generated angle in the range (0^o,90^o) for each iteration and for each particle.

The $\Delta\varphi^i_{j+1}$ value is limited by the ($\Delta\varphi_{min}$; $\Delta\varphi_{max}$) interval. As in case of $\Delta\varphi^i_{j+1}$ the value of the φ^i_j angle is limited and belongs to the (φ_{min}; φ_{max}) interval.

Results

Benchmark Test Functions

The first simulation tests of the new PAPSO algorithm were carried out on the benchmark function. The parameters of optimized functions are listed in Table 1. The obtained results were then compared with the achievements of the standard PSO and θ-PSO algorithms described in [10]. The computations were executed with inertia weight w = 0,6 and acceleration coefficients c_1 = c_2 = 1,7. The maximum number of iterations was established to 1000. The exemplary results of the tests performed for 40 particles of the swarm are depicted in Table 2. The presented values for PAPSO were averaged over 30 trials.

Table 2. PSO, θ-PSO and PAPSO algorithm performance

Function	Algorithm	Number of iterations		Number of achieved solution(%)
		Minimum	Average	
Camel	PAPSO	25	41	100
	θ-PSO	26	40	100
	PSO	32	60	100
Sphere	PAPSO	334	383	100
	θ-PSO	352	406	100
	PSO	577	716	100
Griewank	PAPSO	219	317	100
	θ-PSO	231	334	100
	PSO	375	461	100
Rosenbrock	PAPSO	180	279	100
	θ-PSO	194	283	100
	PSO	569	2268	55

The investigation of benchmark function confirmed the effectiveness of the proposed PAPSO algorithm. In almost all cases the new algorithm turned out to be more effective than the other algorithms used for the tests. For the tested functions (except Camel) the minimum and average numbers of iterations were lower than those of θ-PSO and PSO. In order to find the optimum of this function with required accuracy fewer number of iterations was therefore needed. For Camel function the results of PAPSO and θ-PSO were almost the same but they were much better than the standard PSO.

Coil Arrangement

The study on an 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 computations were executed with acceleration coefficients c_1 =0.4, c_2 = c_3 =0.6. The parameter w was used in the range 0.8 to 0.3 with a linearly decreasing, whereas the maximum number of iterations was fixed to 1000. The results were then compared with the achievements of the standard PSO and the θ-PSO algorithms.

The exemplary results of the tests performed for 10, 30, 50 and 80 particles in the initial population of the swarm are depicted in Fig. 2. All the values were averaged over 50 trials.

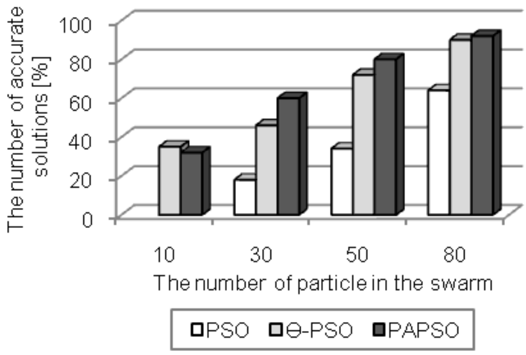

Fig.2. The number of accurate solutions versus the swarm cardinality for standard PSO, θ-PSO and PAPSO

The proposed algorithm turned out to be effective with respect to the number of accurate results and the number of iterations needed to achieve them. In comparison to the standard PSO (Table 3), the new algorithm was able to find more accurate solutions within a few times lower iteration number.

Table 3. The relationship between the swarm cardinality and the number of iterations to achieve the accurate solutions for the PSO, θ-PSO and PAPSO

Algorithm	The number of particles in the swarm			
	10	30	50	80
PAPSO	107	82	64	57
θ-PSO	131	112	76	64
PSO	-	642	567	531

Moreover, the algorithm also achieved to be efficient for small populations of particles. In case of PSO for 10 particles in the swarm, no accurate solution was managed to be obtain when the number of iterations was as high as 1000, whereas the PAPSO could find 32% optimal solutions within only 107 iterations (on average). In most cases, the results were also more accurate than for the θ-PSO algorithm. Only for the population size of 10 particles, the PAPSO algorithm found 3% fewer accurate solutions than θ-PSO, but for lower number of iterations of even 18%.

As far as the optimal solutions are concerned, the best results were obtained for large swarm. For the populations comprising 80 particles, PAPSO algorithm could find over 80% accurate solutions. It was also found, that the proposed new method gave the solutions of the same accuracy, and faster exploration of the search space.

Summary

In the following study, the new PAPSO optimization algorithm with the phase angle vector was proposed. In the proposed method, determination of the velocity vector was omitted. Instead of the velocity vector, the phase angle vector was introduced. The equation of the phase angle increment update is responsible for the particle movement within the search space. The particle position is determined by the mapping of the corresponding angles.

The proposed algorithm was tested on benchmark test functions and was applied to determine the optimal geometry of the coil arrangement generating magnetic field with the specific parameters. The results were then compared with performances of the standard PSO and the θ-PSO algorithms. The calculations obtained in this study confirmed efficiency of the proposed algorithm.

REFERENCES

[1] Kennedy J., Eberhart R.C., *Particle Swarm Optimization*, IEEE Intern. Conf. on Neural Networks, Australia (1995), 1942-1948

[2] Kennedy J., Eberhart R.C., *A New Optimizer Using Particle Swarm Theory*, In: 6th International Symposium on Micromachine and Human Science, Japan (1995), 39-43

[3] Mendes R., Cortez P., Rocha M., Neves J., *Particle Swarm for Feedforward Neural Network Training*, International Joint Conference on Neural Networks, (2002), 1895-1899

[4] Venayagamoorthy G.K., Doctor S., *Navigation of Mobile Sensor Using PSO and Embedded PSO in a Fuzzy Logic Controller*, In: 39th IEEE IAS Annual Meeting on Indus. Applic., Seattle, USA (2004) 1200-1206

[5] 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, Australia (2003), 1440-1447

[6] Cui Z.H., Zeng J.C., Sun G.J., *A Fast Particle Swarm Optimization*, International Journal of Innovative Computing, Information and Control, 6 (2006), 1365-1380

[7] Hassas S., *Using Swarm Intelligence for Dynamic Web Content Organizing*, IEEE Swarm Intelligence Symp.,USA (2003), 19-25

[8] Sato S., Sakaguchi S., Futamata K., Katou K., *Coil Optimization for Homogeneous Magnetic Field with Small Leakage Field*, IEEE Trans. Magn., 36 (2000), 649-653

[9] Robinson J., Rahmat-Samii Y., *Particle Swarm Optimization in Electromagnetics*, IEEE Trans. Anten. Propag., 52 (2004), 397-407

[10] Zhong W., Li S., Qian F., *θ-PSO: a New Strategy of Particle Swarm Optimization*, Journal of Zhejiang University-Science A, 9 (2008), 786-790

[11] Shi Y., Eberhart R.C., *A Modified Particle Swarm Optimizer*, IEEE World Congress on Comput. Intelligence, (1998), 69-73

[12] Tripathi P.K, Bandyopadhyay S., Pal S., *Multi-objective Particle Swarm Optimization with Time Variant Inertia and Acceleration Coefficients*, Information Sciences, 177 (2007), 5033-5049

[13] Trelea I.C., *The Particie Swarm Optimization Algorithm: Convergence Analysis and Parameter Selection*, Information Processing Letters, 85 (2003), 317-325

[14] Bergh F., Engelbrecht A.P., *A Study of Particle Swarm Optimization Particle Trajectories*, Information Sciences, 176 (2006), 937-971

[15] Borowska B., Nadolski S., *Particle Swarm Optimization: The Gradient Correction*, Journal of Applied Computer Science, 17 (2009), 7-15

[16] Forbes L.K., Crozier S., Doddrell M., *Rapid Computation of Static Fields Produced by Thick Circular Solenoids*, IEEE Trans. Magn., 33 (1997), 4405-4410

***Author**: dr inż. Bożena Borowska, Politechnika Łódzka, Instytut Informatyki, ul. Wólczańska 215, 93-005 Łódź, E-mail: bozena.borowska@p.lodz.pl*