Repulsive Particle Swarm Optimization Based on New Diversity

Guochao Niu[1], Baodi Chen[1,2], Jianchao Zeng[3]
1. Software Engineering Institute, Xidian University, Xi′an 710071,China
E-mail: guochao.niu@gmail.com
2. Complex System and Computational Intelligence Laboratory, Taiyuan University of Science and Technology, Taiyuan 030024,China
E-mail: cbdngc@gmail.com
3. Complex System and Computational Intelligence Laboratory, Taiyuan University of Science and Technology, Taiyuan 030024,China
E-mail: zengjianchao@263.net

Abstract: To avoid the problem of premature convergence, a new diversity-guided Particle Swarm Optimizer (PSO), namely MARPSO is proposed, which is a modification of attractive and repulsive PSO (ARPSO), suggested by Riget and Vesterstorm[1]. A novel measure of population diversity function is presented and a new concept of the particle′s best flight direction is introduced. The simulation test results of four classic functions show that: compared with Standard PSO (BPSO) and ARPSO, MARPSO can effectively increase the diversity of swarm, while maintain a higher convergence speed.

Key Words: Particle Swarm Optimization, Population Diversity, Global Search, Particle′s Best Flight Direction

1 INTRODUCTION

Particle Swarm Optimization [2] technique is a population based stochastic search technique first suggested by Kennedy and Eberhart in 1995. Since then it has been used to solve a variety of optimization problems. Its performance has been compared with many popular stochastic search techniques like Genetic algorithms, Differential Evolution, Simulated Annealing etc. [3], [4]. Although PSO has shown a very good performance in solving many test as well as real life optimization problems, it suffers from the problem of premature convergence like most of stochastic search techniques, particularly in case of multimodal optimization problems. The cure of premature convergence greatly affects the performance of algorithm and many times lead to a sub optimal solution. Aiming at this shortcoming of PSO algorithms, many variations have been developed to improve its performance. Some of these methods include fuzzy PSO [5], hybrid PSO [6], intelligent PSO [7], addition of a queen particle [8] etc.

In this paper, we present a simple and effective PSO called MARPSO. MARPSO is a variation of ARPSO, a diversity-guided PSO developed by Riget and Vesterstorm. Like ARPSO, MARPSO uses diversity as a measure to guide the swarm population. In ARPSO if the diversity is above certain threshold d_{high} then particles attract each other, and if it is below the certain threshold d_{low} then the particles repel each other until they meet the required high threshold d_{high} . In our modified version, a new simple population diversity measure method is presented. The algorithm can develop effectively population diversity, and decrease the opportunity of premature convergence. Then, to balance the similarity and the difference of particles, a new concept of the particle′s best flight direction is proposed. In addition, in order to ensure that local convergence of the algorithm performance, the paper also introduces a kind of mutation strategy of the speed and location of particles. The simulation test results show that the MARPSO can effectively increase the diversity of swarm and maintain a higher convergence speed. The paper is organized as follows: in section 2, the BPSO is briefly described. In section 3, the ARPSO and MARPSO is analyzed in detail. And the experimental results are analyzed in section 4, finally the paper concludes with section 5.

This work is supported by Project 9140A24070106DZ01 National Key Foundation of P.R. China.

2 Basic Particle Swarm Optimization (BPSO)

Particle Swarm Optimization (PSO) is a relatively newer addition to a class of swarm based search technique for solving numerical optimization problems. Its mechanism is inspired from the complex social behavior shown by the natural species like flock of birds, school of fish and even crowd of human beings. The particles or members of the swarm fly through a multidimensional search space looking for a potential solution. Each particle adjusts its position in the search space from time to time according to the flying experience of its own and of its neighbors. For a D-dimensional search space the position of the ith particle is represented as $X_i = (x_{i1}, x_{x2}, ..., x_{iD})$. Each particle maintains a memory of its previous best position $P_i = (p_{i1}, p_{x2}, ..., p_{iD})$and a velocity$V_i = (v_{i1}, v_{x2}, ..., v_{iD})$ along each dimension. In each iteration, the P vector of the particle with best fitness in the local neighborhood, designated, and the vector of the current particle are combined to adjust the velocity along each dimension and a new position of the particle is determined using that velocity. The two basic equations that govern the working of PSO are that of velocity vector and position vector given by:

$$v_{ij}(t+1) = \omega v_{ij}(t) + c_1 r_{1j}(p_{ij} - x_{ij}(t)) + c_2 r_{2j}(p_{gj} - x_{ij}(t)) \tag{1}$$

$$x_{ij}(t+1) = x_{ij}(t) + v_{ij}(t+1) \tag{2}$$

The first part of equation (1) represents the inertia of the previous velocity, the second part is the cognition part and it tells us about the personal thinking of the particle, the third part represents the cooperation among particles and is therefore named as the social component. Acceleration constants c_1, c_2 [9] and inertia weight ω [10] are the predefined by the user and r_1, r_2 are the uniformly generated random numbers in the range of [0, 1].

3 ARPSO and MARPSO

To void the problem of premature convergence, Riget and Vesterstorm introduce the attractive and repulsive PSO (Attractive and Repulsive Particle Swarm Optimizer, namely ARPSO). The algorithm uses a diversity measure to have the algorithm alternate between exploring and exploiting behavior. Dynamic adjustment exploring and exploiting proportional, so that it could better improve the efficiency of algorithms. The two equations that govern the working of ARPSO are that of velocity vector and position vector given by:

$$\begin{aligned} &V_{ij}(t+1) = \omega V_{ij}(t) + dir \times (c_1 r_{1j}(P_{ij} - X_{ij}(t)) + \\ &c_2 r_{2j}(P_{gj} - X_{ij}(t))) \end{aligned} \tag{3}$$

$$X_{ij}(t+1) = X_{ij}(t) + V_{ij}(t+1) \tag{4}$$

The diversity measure of the swarm is taken as:

$$diversity(S) = 1/(|S| \times |L|) \sum\nolimits_{i=1}^{|S|} \sqrt{\sum\nolimits_{j=1}^{N} (p_{ij} - \bar{p}_j)^2} \tag{5}$$

Where S is the swarm, $|S|$ is the swarm size, $|L|$ is the length of longest the diagonal in the search space, N is the dimensionality of the problem, P_{ij} is the $j'th$ value of the $i'th$ particle and $\bar{p}_j$ is the average of the $j'th$ dimension over all particles. Note that this diversity measure is independent of swarm size, the dimensionality of the problem as well as the search range in each dimension. By measuring the diversity, the swarm autonomously alternates between attraction and repulsion. As long as the diversity is above a certain threshold d_{low}, the particles attract each other. When the diversity declines below d_{low}, the particles change strategy and start to repel each other until the threshold d_{high} is met. The diversity parameters d_{low} and d_{high} were set at 5.0×10^{-6} and 0.25 respectively.

Supposed there is an abstract particle $\bar{p}$, whose position is $(\bar{p}_1, \bar{p}_2, ...\bar{p}_j, \bar{p}_N)$ in the research space, $\bar{p}_j$ is the average of the $j'th$ dimension over all particles. The expression $\sqrt{\sum_{j=1}^{N} (p_{ij} - \bar{p}_j)^2}$ is the distance between the $i'th$ particle and the abstract particle $\bar{p}$. The expression $1/|S| \sum_{i=1}^{|S|} \sqrt{\sum_{j=1}^{N} (p_{ij} - \bar{p}_j)^2}$ is the mean distance for all particles to the abstract particle $\bar{p}$. The expression $1/(|S| \times |L|) \sum_{i=1}^{|S|} \sqrt{\sum_{j=1}^{N} (p_{ij} - \bar{p}_j)^2}$ is the ratio between the swarm′s mean radius and the search space′s longest radius, namely the swarm′s cover degree to the search space. In some cases, the cover degree is generally higher, the particle swarm is more scattered. In other words, the population diversity is better. Otherwise, the particles are more crowded, the swarm diversity is worse. The ARPSO algorithm is diversity guided BPSO in which the behavior of the swarm is controlled as per the variation in diversity. But, there are two shortcomings for this diversity definition, as follows:

$$\begin{aligned} &diversity(S_j) = 1/|S| \sum\nolimits_{i=1}^{|S|} k \\ &k = \begin{cases} 0, if\,|X_{ij} - P_{ij}| < delta \\ 1, therwise \end{cases} \end{aligned} \tag{6}$$

Where S is the swarm, $|S|$ is the swarm size, $diversity(S_j)$ is the diversity of $j'th$ dimension, X_{ij} is the $j'th$ value of the $i'th$ particle, P_{gj} is the $j'th$ value of the global best particle, $delta$ is the radius of the neighbor field of P_g. Apparently, we can conclude that:

$$0 \leq diversity(S_j) \leq 1 \tag{7}$$

In the evolution process of the swarm, if the value is too low, the swarm will do the repulsive movement, namely the swarm disperses from the global best particle. If the value is too high, the swarm will do the attractive movement, namely the swarm concentrates to the global best particle. The generation equations of swam is taken as follows:

$$\begin{aligned} &v_{ij}(t+1) = \omega v_{ij}(t) d_{ij}(t+1) + dir(t+1) \\ &\times (c_1 r_{1j}(p_{ij} - x_{ij}(t)) + c_2 r_{2j}(p_{gj} - x_{ij}(t))) \end{aligned} \tag{8}$$

$$x_{ij}(t+1) = x_{ij}(t) + v_{ij}(t+1) \tag{9}$$

$$dir(t+1) = \begin{cases} -1, if(dir(t) > 0)\&\&(diversity < d_{low}) \\ 1, if(dir(t) < 0)\&\&(diversity > d_{high}) \\ dir(t), otherwise \end{cases} \tag{10}$$

$$d_i(t+1) = \begin{cases} v\,(t)/|v\,(t)|, if(f(x(t)) < f(p_i)) \\ d_i(t), otherwise \end{cases} \tag{11}$$

Where $dir(t)$ expresses the flight direction of $t'th$ generation, $d_i(t)$ is the $i'th$ particle′s flight direction of $t'th$ generation. The expression $dir(t) = 1$ says that the swarm does the attractive movement. The expression $dir(t) = -1$ says that the swarm does repulsive movement. The d_{high} and d_{low} separately expresses the high threshold and low threshold of the diversity of species respectively. The values of d_{high} and d_{low} will influence the efficiency of MARPSO. For the higher d_{high}, the swarm will maintain higher diversity, so the convergence speed will be lower. For lower d_{low}, the population diversity will be lower, but the convergence speed will be higher. So the values of d_{high} and d_{low} should be neither too low nor too high, and we can choose the experiential values. The test results show that we can set $d_{high} = 0.8$, and set $d_{low} = 0.2$. The expression $d_i(t) = 1$ expresses that the inertia part of the formula is beneficial to the search. The expression $d_i(t) = -1$ expresses that the inertia part of the formula is beneficial to the search, too. The flight factor of each particle is not necessarily the same as the swarms flying direction, therefore, there are common characters among particles, and there are differences, too. The convergence rate is faster when the common character is bigger. If the differences are bigger, the capability of the global search is good. The algorithm is more able to jump out of the best local region. Under the guidance of the population diversity, the MARPSO makes the similarities and differences of particles achieve a balance, so it is beneficial to do the search.

Neither the MARPSO nor the ARPSO can guarantee the local convergence and the global convergence of the algorithm. Therefore, in order to improve the convergence of the algorithm, the paper introduces the speed and location of the mutation strategy, which enlightened in the literature [11], but they are given new meaning, the definition are as follows:

$$v(t+1) = \begin{cases} V_{\max} \times r_3^t, if(|v(t)| < V_{\min})\&\&(r_4 < 0.5) \\ -V_{\max} \times r_3^t, if(|v(t)| < V_{\min})\&\&(r_4 \geq 0.5) \end{cases} \tag{12}$$

$$x(t+1) = \begin{cases} p_g + r_3^t, if(|v(t)| < V_{\min}).and.(r_4 < 0.5) \\ p_g - r_3^t, if(|v(t)| < V_{\min}).and.(r_4 \geq 0.5) \end{cases} \tag{13}$$

Where $V_{\max}$ and $V_{\min}$ respectively express high threshold and low threshold of the speed of the particles, r_3 and r_4 are real numbers chosen uniformly and at random in a given interval, usually [0, 1]. When the speed of the particles is less than$V_{\min}$, we have the speed and location of the particles to carry out mutation, particles are distributed on both side of the P_g by 50% probability. With the unlimited increasing of the evolution iterations, particles that are on both side of the P_g gradually approach the best global particle. In the formulas (12) and (13), the value of the $V_{\min}$ will affect the effect of the algorithm implementation. If the threshold is too big, swarms will lead to chaos, so that the algorithm cannot be an effective local search. If the threshold is too small to make the speed of the particles decline in a relative short time, the speed of the search of the algorithm cannot be improved effectively. By way of the experimental test, the value of the $V_{\min}$ will not seriously affect the efficiency of the algorithm, so we can get the basic experiential value.

Step1: Carry on the randomization settings to all particles$'$ positions and speeds;
Step2: Set $j = 0$, D is the dimension of search space;
Step3: Calculate $diversity_j(t)$ that is the diversity of the $t'th$ generation of the $j'th$ dimension;
Step4: According to $diversity_j(t)$, set the swarm$'$s flight direction $dir_j(t)$ in the $t'th$ generation of the $j'th$ dimension;
Step5: Set $j = j + 1$;
Step6: If $j < D$, then return to step3;
Step7: Set $i = 0$, N is the swarm size;
Step8: Updating the velocity an position of the $i'th$ with equations (8)-(13);
Step9: Calculate the $i'th$ particle$'$s fitness;
Step10: If the $i'th$ particle$'$s fitness is better than itself best position $P_i's$, takes it as the new itself best position;
Step11: If the $i'th$ particle$'$s fitness is better than global best position, takes it as the new global best position, and updates $d_j(t)$ which is the best flight direction of individual particle;
Step12: Make $i = i + 1$;
Step13: If $i < N$, return to step3;
Step14: Loop to step2 until a stop criterion is met, usually a sufficiently good fitness value or a predefined maximum number of generations $G_{\max}$.
Step15: The algorithm end.

4 EXPERIMENT RESULTS

We have tested the MARPSO model on four standard functions. All four are widely known benchmark functions for testing the performance of different evolutionary optimization strategies such as evolutionary programming, simulated annealing, genetic algorithms and PSO. The four test functions are:

- Griewank, n-dimensional

$$f_1(x) = \sum_{i=1}^n x_i^n/4000 - \prod_{i=1}^n \cos(x_i/\sqrt{i}) + 1 \quad where -600 \leq x_i \leq 600 \tag{14}$$

- AckleyF1, n-dimensional

$$f_2(x) = e + 20 - 20\exp\left(-0.2\sqrt{1/n\sum_{i=1}^n x_i^2}\right) - \exp\left(1/n\sum_{i=1}^n \cos(2\pi x_i)\right) \quad where -32 \leq x_i \leq 32 \tag{15}$$

- Rosenbrock, n-dimensional

$$f_3(x) = \sum_{i=1}^n \left(100\left(x_{i+1} - x_i^2\right) + (x_i - 1)^2\right) \quad where -30 \leq x_i \leq 30 \tag{16}$$

- Rastrigin, n-dimensional

$$f_4(x) = \sum_{i=1}^n \left(x_i^2 - 10\cos(2\pi x_i) + 10\right) \quad where -5.12 \leq x_i \leq 5.12 \tag{17}$$

In our selection, several aspects of multi-modal optimization are considered. Each test function has been chosen to have unique characteristics to extract as much information about the MARPSO as possible. The Griewank test function is Sphere-like with added noise. In low dimensions this is a highly multi-modal function, whereas in higher dimensions the Griewank function resembles the plain Sphere-function, because of the added noise dimension. The Rosenbrock function is extremely steep when the optimum is being approached from far and banana-shaped close to the optimum. Additionally, the Rosenbrock function, as the only one, has a strong dependency between variables x_i. The AckeyF1 and Rastrigin objective functions are both highly multi-modal in all dimensions, but they have different steepness. All test functions have a global minimum at (0, 0,. . . ,0) with a fitness value of 0.

We made experiments with each test function in 20, 50, and 100 dimensions. The performance of the modified model is directly compared to the BPSO and the ARPSO. The number of evaluations in the different dimension is 20d =

40000 evaluations, 50d = 100000 evaluations, and 100d = 200000 evaluations. A linearly decreasing inertia weight is used which starts at 0.9 and ends at 0.4, with the user defined parameters $c_1 = 2.0$ and $c_2 = 2.0$, error is set to be 1.0×10^{-10}. Furthermore, the value of $delta$ affects the size of neighbor field of the P_g, and affects the calculation of swarm diversity. If $delta$ is too high or too low, the diversity′s reliability will be declined. So, we should carefully choose the value of $delta$ according to convergence accuracy, here is $delta = 1.0 \times 10^{-10}$.
The performance of the modified model (MARPSO) is directly compared to the BPSO and the ARPSO. The experimental data of the ARPSO algorithm is from the literature [1]. The results of the benchmark problems $f_1 - f_4$ are separately shown in Tables 1-4 in terms of mean best fitness.

Table 1: Test result of Griewank

Dim	Algorithms	Mean best fitness
20	BPSO	1.74E-2
	ARPSO	2.50E-2
	MARPSO	4.03E-3(86%)
50	BPSO	1.35E-2
	ARPSO	3.05E-2
	MARPSO	1.97E-4(97%)
100	BPSO	1.25E-2
	ARPSO	9.84E-2
	MARPSO	0(100%)

Table 2: Test result of AckleyF1

Dim	Algorithms	Mean best fitness
20	BPSO	0.018
	ARPSO	0.33E-7
	MARPSO	0
50	BPSO	0.668
	ARPSO	0.027
	MARPSO	2.39E-10
100	BPSO	0.830
	ARPSO	0.218
	MARPSO	3.99E-9

Tables 1- 4 show the results of the conducted experiments. The results on performance clearly show that the MARPSO is a better than the BPSO and the ARPSO on all the test functions, expect for on the Griewank function. For the Griewank function, as the dimension increases, the convergence precision of MARPSO gradually exceed over the BPSO and the ARPSO. However, on the AckleyF1, the Rosenbrock, and the Rastrigin test functions, the BPSO algorithm cannot easily find the global optimum. The MARPSO algorithm has more opportunities to find the global optimum, and the obtained fitness value is on average much better than that of the BPSO and the ARPSO. The greatest improvements are found for the AckeyF1, the Rosenbrock and the Rastrigin benchmark functions, especially in high-dimensional situations. So it can be seen that the diversity definition we proposed is better than the definition Riget proposed.

Table 3: Test result of Rosenbrock

Dim	Algorithms	Mean best fitness
20	BPSO	11.16
	ARPSO	2.34
	MARPSO	0.13
50	BPSO	30.8
	ARPSO	10.43
	MARPSO	1.28
100	BPSO	122.14
	ARPSO	103.46
	MARPSO	16.93

Table 4: Test result of Rastrigin

Dim	Algorithms	Mean best fitness
20	BPSO	9.71
	ARPSO	0
	MARPSO	0
50	BPSO	47.14
	ARPSO	0.02
	MARPSO	0
100	BPSO	96.59
	ARPSO	0.438
	MARPSO	0

In order to compare the MARPSO,the ARPSO and the BPSO in the convergence processes, figures 1 to 4 show the performance over time for some of our experiments (Note: In order to facilitate the evolution of the show and observation, the paper on the function of fitness for admission to 10 at the end of the few). The experimental environment: dimension is 50, the swarm size is 20 and the largest iteration is 10000. It can be seen from the figures that: compared with the BPSO and ARPSO, the MARPSO algorithm can rapidly converge to the global or near the optimal solution (about after 6,000 generations evolution).

5 CONCLUSION

It is a well-known fact that maintaining a high diversity while preserving fast convergence are two contradicting features. The ARPSO algorithm is diversity guided BPSO in which the behavior of the swarm is controlled as per the variation in diversity. Through the swarm′s attractive movement and repulsive movement, the population diversity and convergence rate are balanced. The ARPSO guarantees swarm′s high diversity, while avoids falling into the local extreme point, and maintains the high convergence rate, is one kind of good optimized strategy, but its diversity function′s operations are relatively complex. This article proposed a new simpler diversity function whose operation is very simple, the experiment proves it can instruct swarm *the attractive movement* and *the repulsive movement* effectively. At the same time, this paper proposed a new concept of the particle′s best flight direction and a kind of new mutation strategy, which greatly improved the

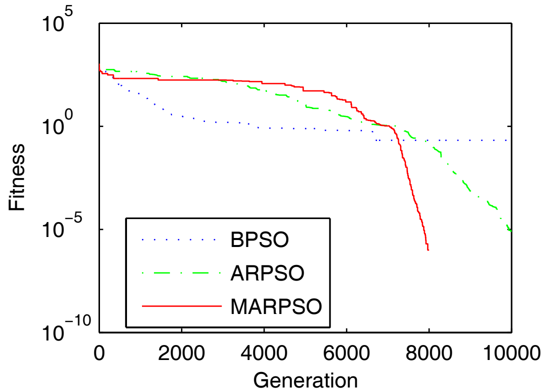

Figure 1: Evolution curve of Griewank

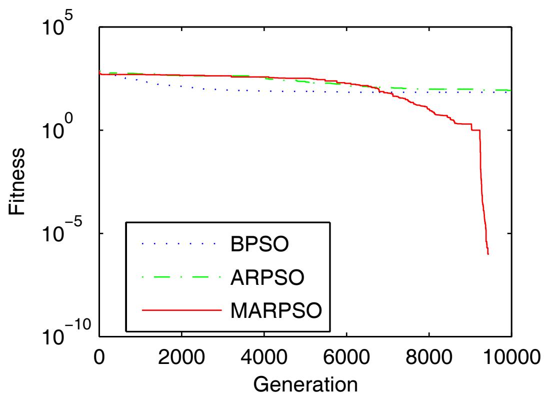

Figure 2: Evolution curve of AckleyF1

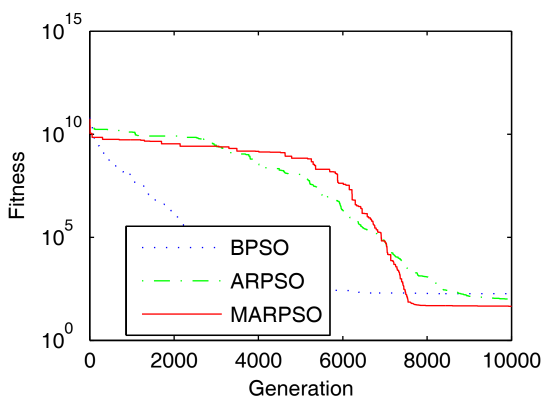

Figure 3: Evolution curve of Rosenbrock

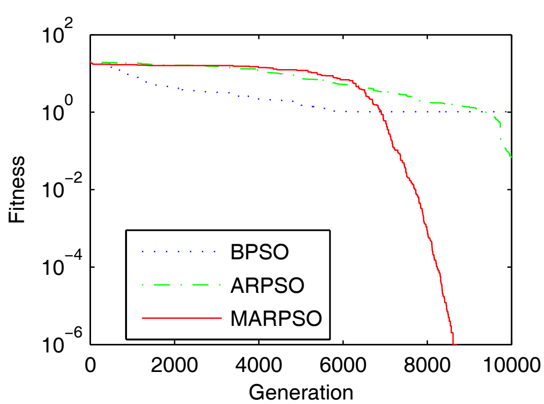

Figure 4: Evolution curve of Rastrigin

convergence of the algorithm. However, in certain extreme cases, population diversity of these two methods cannot really measure the group of discrete degree (i.e. population diversity). Therefore, it is necessary to explore more rational and efficient method of population diversity measure, to take more advantage of population diversity in swarm evolution.

REFERENCES

[1] J Riget, J S Vestterstrom. A Diversity-guided particle swarm optimizer ˇ the ARPSO. Technical report, EVAlife, Dept. of Computer Science, University of Aarhus, Denmark, 2002.

[2] J. Kennedy, R. Eberhart. Particle Swarm Optimization. IEEE International Conference on Neural Networks (Perth Australia), IEEE Service Center, pg. IV, 1942-1948, 1995.

[3] J. C. Zeng, J. Jie, Z. H. Cui. Particle Swarm Optimization. Beijing:Science Press, Chap. 3, 2004.

[4] J. Vesterstrom, R. Thomsen. A Comparative study of Differential Evolution, Particle Swarm Optimization, and Evolutionary Algorithms on Numerical Benchmark Problems. In Proc. IEEE Conger. Evolutionary Computation, Portland, OR, Jun. 20-23, 1980-1987, 2004.

[5] Y. Shi, R.C. Eberhart. Fuzzy Adaptive particle Swarm Optimization. Proc. Congress on Evolutionary Computation. seoul korea, IEEE Service Center, 2001.

[6] G. Ciuprina, D. Loan and I. Munteanu. Use of Intelligence Particle Swarm Optimization in Electromagnetics. IEEE Transactions on MAGNETICS, Vol.38, No.2, 1037-1040, 2002.

[7] I C Trelea. The particle swarm optimization algorithm: Convergence analysis and parameter selection. Information Processing Letters, Vol.95, No.6, 317-325, 2003.

[8] J. Kennedy. The Particle Swarm: Social Adaptation of Knowledge. IEEE International Conference on Evolutionary Computation (Indianapolis, Indiana), IEEE Service Center, 3030-308, 1997.

[9] R C. Eberhart, Y. Shi. Particle Swarm Optimization: developments, Applications and Resources. IEEE Int. Conference on Evolutionary Computation, 1-86, 2001.

[10] Y. H. Shi, R. C. Eberhart. A Modified Particle Swarm Optimizer. IEEE International Conference on Evolutionary Computation, Alaska, 69-73, 1998.

[11] Ratnaweera, S. K. Halgamuge, H.C. Watson. Self-organizing hierarchical particle swarm optimizer with time varying accelebration coefficients. IEEE Trans. On Evolutionary Computation, Vol.8, No.3, 240-255, 2004.