An Adaptive Individual Inertia Weight Based on Best, Worst and Individual Particle Performances for the PSO Algorithm

G. Spavieri, R. A. S. Fernandes, G. G. Lage
Electrical Engineering Department
Federal University of São Carlos
São Carlos, Brazil
ricardo.asf@ufscar.br, glage@ufscar.br

R. A. L. Rabêlo
Electrical Engineering Department
Federal University of Piauí
Teresina, Brazil
ricardoalr@ufpi.edu.br

***Abstract*—Due to the growing need for metaheuristics with features that allow their implementation for real-time problems, this paper proposes an adaptive individual inertia weight in each iteration considering global analysis, i.e., the best and worst particles' performance, and individual analysis, i.e., individual particle's performance. As a result, the proposed adaptive individual inertia weight presents a faster convergence for the Particle Swarm Optimization (PSO) algorithm when compared to other inertia mechanisms. The proposed algorithm is also suitable for real-time problems when the actual optimal solution is not necessarily reached, as a feasible solution is optimized in relation to an initial solution. In this sense, the PSO with the proposed adaptive individual inertia weight was tested using six benchmark functions in continuous domain, namely: Sphere, Rosenbrock, Rastrigin, Griewank, Alpine and Schaffer. Through the obtained results it is noteworthy to mention that the adaptive individual inertia weight features a rapid convergence of the PSO algorithm in the first 400 iterations. Furthermore, the proposed PSO was compared to other two algorithms (one with constant inertia weight and other with linearly decreasing inertia weight), where the proposed PSO algorithm was able to reach better optimized results in five out of the six tested benchmark functions at the end of 2000 iterations.**

Keywords—adaptive inertia weight, benchmark functions, particle swarm optimization.

I. INTRODUCTION

The optimization of processes and systems consists in a useful task for enterprises due to the possibility of minimizing operational costs and/or maximizing the use of available resources, and the application of optimization methods to solve such problems may be applied to distinct fields of knowledge. Thus, according to their mathematical modeling, different problems can be solved by a variety of methods, such as the metaheuristics.

In the above-mentioned category, some algorithms that have been widely used for many applications are Genetic Algorithms (GA) [1], Particle Swarm Optimization (PSO) [2], and Ant Colony Optimization (ACO) [3], among others. However, in this paper, it will be highlighted the advances on PSO.

In this sense, the idealization of a PSO algorithm featuring faster and more efficient convergence has been a challenge for the scientific community. Thus, some advances related to the PSO are:

- the use and modifications of the inertia parameter [4]-[10], that will be discussed in Section II;
- hybridizations [11];
- new equations and definitions to update particle's velocity and position [12];
- development of multi-objective algorithms [13];
- development of bi-level algorithms [14].

Following the above context, this paper proposes an adaptive individual inertia weight which is calculated in each iteration considering global analysis (the best and worst particles' performance) and individual analysis (individual particle's performance). Thus, it can be noticed that the proposed adaptive individual inertia weight presents a faster convergence for the PSO algorithm when compared to other inertia mechanisms. For this reason, it can be used for real-time applications. In this way, the PSO with the proposed adaptive individual inertia weight was tested and compared with two other PSO algorithms, namely: constant inertia weight PSO and linearly decreasing inertia weight PSO. In order to guarantee a good comparison, we used six benchmark functions in continuous domain, namely: Sphere, Rosenbrock, Rastrigin, Griewank, Alpine and Schaffer.

The remaining of this paper is organized as follows. Section II presents some literature review on inertia weight mechanisms. In Section III, the proposed adaptive individual inertia weight mechanism is presented. In Section IV, the considered benchmark functions and inertia parameters setup are presented. The obtained results and a comparison between the proposed PSO with adaptive individual inertia weight with the PSO with constant inertia weight and the PSO with linearly decreasing inertia weight are presented in Section V. At last, the main contributions of this paper are highlighted and summarized in Section VI.

II. Inertia Weight Mechanisms

The first inertia weight mechanism related to the PSO algorithm was proposed in [4], and it consists in a positive constant or even a positive linear or nonlinear function of time that multiplies the current velocity of the particles in each iteration of the algorithm. The objective of this classical mechanism was to play the role of balancing the global search and the local search characteristics of convergence, preventing the search for local optimum solutions only. The authors attested that when the inertia weight is small, the PSO works more like a local search algorithm. So, if there is an acceptable solution within the initial search space, the PSO will find the global optimum quickly, otherwise it will not find the global optimum. On the other hand, if the inertia weight is large, the PSO works more like a global search method and will take more iterations to find the global optimum, being more likely to fail. The authors also observed that by using a linearly decreasing function of time instead of a fixed constant, the algorithm tends to initially exploit the search space and, then, perform the exploration of a local area to find the global optimum. The authors in [5] proposed the use of a random value of inertia weight to enable the PSO to track the optimal solution in a dynamic environment, in which it is difficult to predict whether in a given time exploration or exploitation would be desired.

In accordance with the idea of a decreasing inertia weight over iterations, the work developed in [6] proposes a nonlinear decreasing variant of inertia weight mechanism in which a new parameter, the nonlinear modulation exponent, controls the overall variation of the inertia along iterations. This mechanism allows the swarm to employ an aggressive tuning during initial iterations to better search in the solution space and quickly arrive near the optimum solution and, then, gradually employs fine tuning during later iterations so that the optimum solution is approached with a better accuracy. Unlike the widespread used decreasing inertia weight, in [7] the authors propose a linearly increasing inertia weight mechanism. This mechanism was tested for minimizing four different nonlinear functions and the results indicated that the linearly increasing inertia weight, compared to the linearly decreasing inertia weight, greatly improves the accuracy and convergence speed of global search, with almost no additional computational burden.

In addition to the constant, random, linear or nonlinear decreasing or increasing inertia weight mechanism, another category that has been researched is the adaptive inertia weight strategies. In these mechanisms, the inertia weight is adapted based on one or more feedback parameters that monitor the search process of the optimal solution [8]. In [9], the authors propose the use of a Fuzzy System (FS) with nine rules to dynamically adapt the population inertia weight. The proposed FS uses the normalized current best performance evaluation and the current inertia weight as input variables and the output variable is the change of the inertia weight. Experiments with three benchmark functions showed that PSO with a FS for tuning its inertia weight can improve its performance compared to the PSO with a linearly decreasing inertia weight for, at least, these three benchmark functions. In [10], the authors propose an adaptive inertia weight mechanism in which each particle of the swarm has its own inertia weight based on its rank. In each iteration, the particle that presents the best performance is assigned as the highest particle of the rank and receives the maximum inertia weight of the swarm in order to keep moving at the current direction. On the other hand, the particle that presents the worst performance receives the minimum inertia weight, making it easier to this particle change its direction. The proposed approach produced results comparable or better than those obtained by the PSO with constant inertia weight when applied to the economic load dispatch problem.

In order to develop an inertia weight mechanism that takes into account the global and the individual particles fitness, this work proposes an adaptive individual inertia weight mechanism based on the relative performance of each particle compared to the best and the worst current performances, which will be detailed in the following section.

III. Proposed Adaptive Inertia Weight Mechanism

The basic idea of the adaptive individual inertia weight mechanism proposed in this paper is to set the inertia weight of each particle based on the relative fitness compared to the best and the worst global fitness. For the purpose of implementation, for each iteration of the PSO, the inertia weight of each particle is calculated by:

$$\omega_i^{k+1} = \omega_{\min} + \left(\omega_{\max} - \omega_{\min}\right)\frac{\left(f_i^k - fworst^k\right)}{\left(fbest^k - fworst^k\right)}, \qquad (1)$$

where the subscript i denotes the i^{th} particle; the superscript k denotes the k^{th} iteration; $\omega_{máx}$ and $\omega_{mín}$ are constants that represent, respectively, the maximum and the minimum possible inertia weight; $f_{best}{}^k$ and $f_{worst}{}^k$ are, respectively, the best and the worst fitness values out of all the swarm in the k^{th} iteration; and $f_i{}^k$ is the fitness of the i^{th} particle in the k^{th} iteration. Since in the proposed mechanism each particle has its own inertia weight, the velocity update equation is slightly changed to:

$$v_i^{k+1} = \omega_i^k v_i^k + r_1\varphi_1\left(pbest_i^k - x_i^k\right) + r_2\varphi_2\left(gbest^k - x_i^k\right), \qquad (2)$$

where v_i^k and x_i^k are the velocity and the position of particle i in the k^{th} iteration, respectively; $pbest_i^k$ and $gbest^k$ are, respectively, particle i's best position and the global best position up to the k^{th} iteration; φ_1 and φ_2 are positive constants, called the cognitive and social parameter respectively; and, finally, r_1 and r_2 are random numbers uniformly distributed between 0 and 1.

As it can be noticed from (1), the swarm inertia weight distribution is linearly disposed in the range of the swarm worst and best fitness values in the current iteration and, differently from the proposed mechanisms presented in [4]-[7], it is not a constant value, a random value, or a value directly dependent on the iteration number. In this mechanism, the particles that have the best fitness in the current iteration will receive the highest inertia weights in the next iteration, so they will tend to keep moving at the current best search direction. On the other hand, the particles that have the worst fitness in the current iteration will be the ones that receive the smallest inertia weight values in the next iteration, encouraging them to change their direction. Thus, the preferable search directions will be those followed by the particles with the best fitness values.

Despite the fact that the adaptive inertia weight mechanism proposed in this paper is similar to the mechanism proposed in [10], the main difference between these approaches is the method used to rank the particles' performances. While in [10] each particle of the swarm has an inertia weight different from all of the others, even those that showed the same fitness, in the approach presented in this paper, if two particles have the same fitness, then the same inertia weight value will be assigned to both of them.

In the next section, the proposed PSO with adaptive inertia weight will be properly tested and compared to the most popular mechanisms of inertia: the constant inertia weight and the linearly decreasing inertia weight. From this point, the proposed adaptive individual inertia weight PSO will be called AIW-PSO, the constant inertia weight will be called CIW-PSO and the linearly decreasing inertia weight will be called LDIW-PSO.

IV. Experimental Setup

To validate and verify the performance of the proposed inertia weight mechanism, it was tested using four nonlinear functions considered in [15], and two additional benchmark functions.

The first one is the Sphere function, and it consists in a nonlinear function $f_1 : \mathbb{R}^n \to \mathbb{R}$ such that:

$$f_1(x) = \sum_{i=1}^{n} x_i^2 , \tag{3}$$

where $x \in \mathbb{R}^n \,|\, -100 < x_i < 100$, $i = 1, \ldots, n$. The second one is the Rosenbrock function, and it consists in a nonlinear, non-convex and non-separable function $f_2 : \mathbb{R}^n \to \mathbb{R}$ such that:

$$f_2(x) = \sum_{i=1}^{n-1} \left[100\left(x_{i+1} - x_i^2\right)^2 + \left(x_i - 1\right)^2 \right], \tag{4}$$

where $x \in \mathbb{R}^n \,|\, -2.048 < x_i < 2.048$, $i = 1, \ldots, n$.

The third function is the generalized Rastrigin function, and it consists in a nonlinear and non-convex function $f_3 : \mathbb{R}^n \to \mathbb{R}$ such that:

$$f_3(x) = \sum_{i=1}^{n} \left(x_i^2 - 10\cos\left(2\pi x_i\right) + 10 \right), \tag{5}$$

where $x \in \mathbb{R}^n \,|\, -5.12 < x_i < 5.12$, $i = 1, \ldots, n$.

The fourth function is the generalized Griewank function, and it consists in a nonlinear and non-convex function $f_4 : \mathbb{R}^n \to \mathbb{R}$ such that:

$$f_4(x) = \frac{1}{4000} \sum_{i=1}^{n} x_i^2 - \prod_{i=1}^{n} \cos\left(\frac{x_i}{\sqrt{i}} \right) + 1 , \tag{6}$$

where $x \in \mathbb{R}^n \,|\, -600 < x_i < 600$, $i = 1, \ldots, n$.

The fifth function is the Alpine function, and it consists in a nonlinear and non-convex function $f_5 : \mathbb{R}^n \to \mathbb{R}$ such that:

$$f_5(x) = \sum_{i=1}^{n} \left| x_i \sin\left(x_i\right) + 0.1 x_i \right| , \tag{7}$$

where $x \in \mathbb{R}^n \,|\, -5.12 < x_i < 5.12$, $i = 1, \ldots, n$.

The sixth and last function is the Schaffer function, and it consists in a nonlinear, non-convex and non-separable function $f_6 : \mathbb{R}^2 \to \mathbb{R}$ such that:

$$f_6(x_1, x_2) = 0.5 + \frac{\sin^2\left(\sqrt{x_1^2 + x_2^2}\right) - 0.5}{\left[1 + 0.001\left(x_1^2 + x_2^2\right)\right]^2} , \tag{8}$$

where $x \in \mathbb{R}^n \,|\, -100 < x_i < 100$, $i = 1, \ldots, n$.

Observe that all functions are nonlinear, and that functions f_2 to f_6 are non-convex, which characterize possible multiple minima depending on their domain.

In this paper, the dimensions of the domains of the considered functions are: n = 5 for f_1; n = 10 for f_2; n = 30 for f_3; n = 50 for f_4; and, at last, n = 10 for f_5. Notice that the Schaffer function is the only function that is exclusively defined in the $\mathbb{R}^2$.

The global optimal solutions and the corresponding function values for the considered benchmark functions are listed in Table I.

TABLE I. Global Optima and Global Minima.

Function	Global Optimal Solution	Global Minimum
Sphere	(0,0,…,0)	0
Rosenbrock	(1,1,..,1)	0
Rastrigin	(0,0,…,0)	0
Griewank	(0,0,…,0)	0
Alpine	(0,0,…,0)	0
Schaffer	(0,0,…,0)	0

In all experiments, the swarm size was 50 and it was initialized exactly the same way. The cognitive and social factors were both set to 1.25, the maximum velocity was set to the right bound of the hypercube space for in each function, and every function was tested 20 times independently, each time with 2000 iterations. As follows, Table II lists the inertia weight parameters considered for the three mechanisms for each benchmark function.

TABLE II. INERTIA PARAMETERS SETUP.

Function	CIW-PSO	LDIW-PSO		AIW-PSO	
	ω	$\omega_{mín}$	$\omega_{máx}$	$\omega_{mín}$	$\omega_{máx}$
Sphere	0.9	0.4	0.9	0.4	0.9
Rosenbrock	0.9	0.4	0.9	0.4	0.9
Rastrigin	0.9	0.7	0.9	0.7	0.9
Griewank	0.9	0.4	0.9	0.4	0.9
Alpine	0.9	0.4	0.9	0.4	0.9
Schaffer (F6)	0.9	0.7	0.9	0.7	0.9

V. EXPERIMENTAL RESULTS AND DISCUSSION

The metric used to evaluate the accuracy that each algorithm can reach up to a given number of iterations was the Mean Optimum Fitness (MOF), i.e., the mean of the global best fitness obtained in the 20 runs of the algorithm. Thus, Table III presents the obtained MOF for the six benchmark functions. The MOF indicated that the adaptive inertia weight mechanism proposed in this paper can get better optimum fitness results for five out the six benchmark functions. These results indicate that the proposed mechanism has better global search performance when compared to the other two inertia weight PSO variants. The results obtained for the Rosenbrock and the Griewank functions highlight the performance increase accomplished by the proposed adaptive individual inertia weight PSO. The only exception noticed was for the sphere function, in which all three algorithms reached an excellent MOF, but the best value was reached by the LDIW-PSO.

TABLE III. MEAN OPTIMUM FITNESS VALUES

Function	CIW-PSO	LDIW-PSO	AIW-PSO
Sphere	2.123E-12	2.613E-264	4.892E-30
Rosenbrock	71.79	25.15	2.378
Rastrigin	217.9	122.5	103.7
Griewank	112.0	31.84	4.611
Alpine	1.689E-3	1.321E-15	1.271E-15
Schaffer (F6)	2.776E-18	0.000	0.000

To analyze and compare the convergence speed of the proposed inertia weight mechanism, Figures 1 to 6 indicate the average fitness of the particles along the iterations.

Figures 1 to 6 clearly illustrate that the proposed AIW-PSO convergence is faster than the other two tested algorithms for the six analyzed benchmark functions, even for the Sphere function, in which the MOF of the proposed mechanism was outperformed by the linearly decreasing inertia weight mechanism. It can be argued, nevertheless, that the fast convergence characteristic of the AIW-PSO at the early iterations indicates that the proposed inertia weight mechanism is more efficient than the constant and the linearly decreasing inertia weight mechanism for applications that, for some reason, the maximum number of function evaluations is restricted.

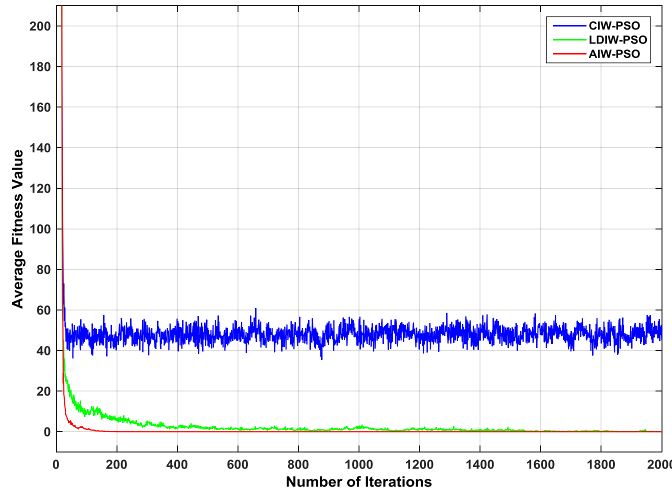

Fig. 1. Sphere function minimization convergence characteristic.

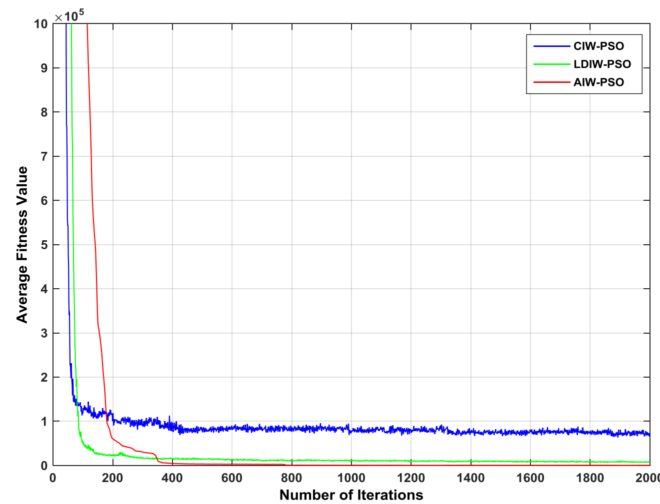

Fig. 2. Rosenbrock function minimization convergence characteristic.

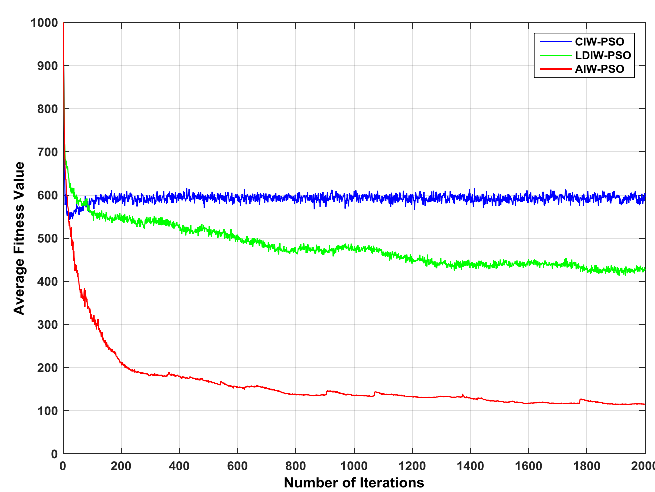

Fig. 3. Rastrigin function minimization convergence characteristic.

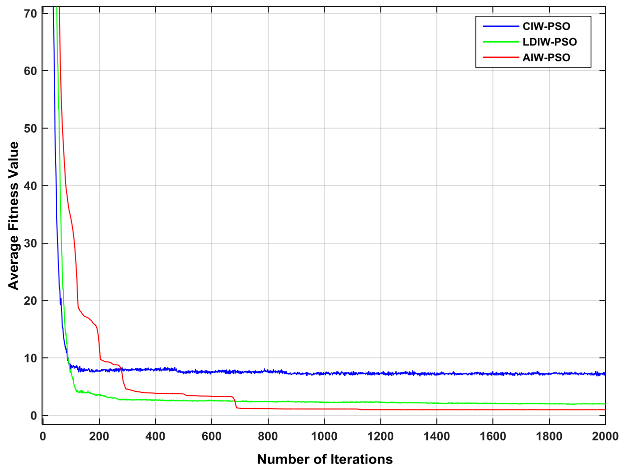

Fig. 4. Griewank function minimization convergence characteristic.

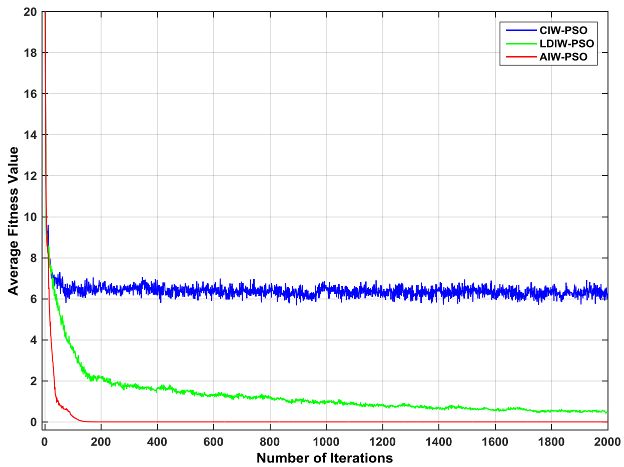

Fig. 5. Alpine function minimization convergence characteristic.

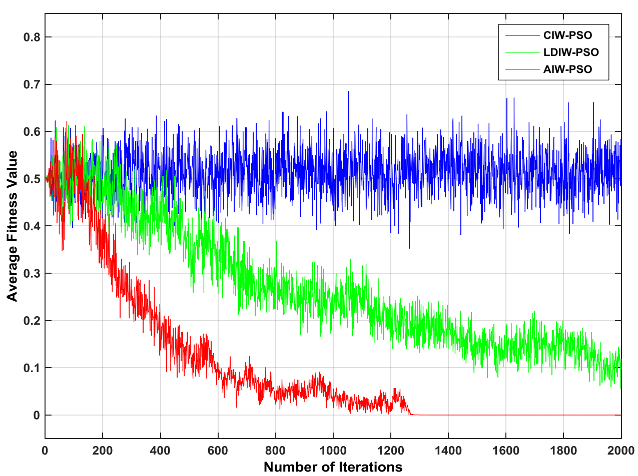

Fig. 6. Schaffer (F6) function minimization convergence characteristic.

VI. Conclusions

This paper presented a proposal for adaptive individual inertia weight, where the inertia parameter of each particle is updated as a function of the best and worst particle's fitness and the maximum and minimum values of inertia obtained for the swarm in each iteration. The proposed PSO algorithm was compared with two other algorithms with inertia mechanisms, being one of them with constant inertia weight and the other with linearly decreasing inertia weight. The obtained results allowed the conclusion that the proposed adaptive inertia weight provides a faster convergence in the first 400 iterations for all of the considered benchmark functions. This characteristic makes the proposed PSO algorithm quite suitable for real-time applications. After 2000 iterations, the proposed PSO algorithm achieved feasible results better than those obtained by the other two PSO algorithms for five out of the six nonlinear benchmark functions. However, it is noteworthy to mention that just for the Sphere benchmark function, the proposed algorithm presented a worse outcome ($4.892e^{-30}$) when compared to the PSO with linearly decreasing inertia weight ($2.613e^{-264}$). However, both results are very close to zero, that is, the optimal solution. Therefore, the proposed algorithm presented characteristics that could be better explored in future research as well as practical applications.

References

[1] D. Goldberg. *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wesley Professional, 1989.

[2] J. Kennedy and R. Eberhart, "Particle Swarm Optimization," *Proc. of the IEEE Int. Conf. on Neural Networks*, vol.4, pp.1942-1948, 1995.

[3] M. Dorigo and L. M. Gambardella, "Ant Colony System: A Cooperative Learning Approach to the Traveling Salesman Problem," *IEEE Trans. on Evolutionary Computation*, vol. 1, no. 1, pp. 53-66, 1997.

[4] Y. Shi and R. C. Eberhart, "A Modified Particle Swarm Optimizer," *Proc. of the IEEE Int. Conf. on Evolutionary Comp.*, pp. 69-73, 1998.

[5] R. C. Eberhart and Y. Shi, "Tracking and Optimizing Dynamic Systems with Particle Swarms," *Proc. of the Congress on Evolutionary Comp.*, vol.1, pp. 94-100, 2001.

[6] A. Chatterjee and P. Siarry, "Nonlinear Intertia Weight Variation for Dynamic Adaptation in Particle Swarm Optimization," *Computers & Operations Research*, vol. 33, no. 3, pp. 859-871, 2006.

[7] Y. Zheng, L. Ma, L. Zhang, and J. Qian, "Empirical Study of Particle Swarm Optimizer with an Increasing Inertia Weight," *IEEE Congress on Evolutionary Computation*, vol. 1, pp. 221-226, 2003.

[8] A. Nickabadi, M. M. Ebadzadeh, and R. Safabakhsh, "A Novel Particle Swarm Optimization Algorithm with Adaptive Inertia Weight," *Applied Soft Computing*, vol. 11, no. 4, pp. 3658-3670, 2011.

[9] Y. Shi and R. C. Eberhart, "Fuzzy Adaptive Particle Swarm Optimization," *Proc. of the Congress on Evolutionary Computation*, vol. 1, pp. 101-106, 2001.

[10] B. K. Panigrahi, V. R. Pandi, and S. Das, "Adaptive Particle Swarm Optimization Approach for Static and Dynamic Economic Load Dispatch," *Energy Conversion and Management*, vol. 49, no. 6, pp. 1407-1415, 2008.

[11] T. Niknam and A. Babak, "An Efficient Hybrid Approach Based on PSO, ACO and k-Means for Cluster Analysis," *Applied Soft Computing*, vol. 10, pp. 183-197, 2010.

[12] Z. Zhan, J. Zhang, Y. Li, and Y. Shi, "Orthogonal Learning Particle Swarm Optimization," *IEEE Trans. on Evolutionary Comp.*, vol. 15, no. 6, pp. 832-847, 2011.

[13] X. Hu and R. Eberhart, "Multiobjective Optimization Using Dynamic Neighborhood Particle Swarm Optimization," *IEEE WCCI*, 2002.

[14] R. J. Kuo and C. C. Huang, "Application of Particle Swarm Optimization Algorithm for Solving Bi-level Linear Programming Problem," *Computers & Mathematics with Applications*, vol. 58, no. 4, pp. 678-685, 2009.

[15] Y. Shi, and R. C. Eberhart, "Empirical study of Particle Swarm Optimization," *Proc. of the Congress on Evolutionary Comp.*, vol. 3, pp. 1945-1950, 1999.