International Journal on
"Technical and Physical Problems of Engineering"
(IJTPE)
Published by International Organization of IOTPE

ISSN 2077-3528
IJTPE Journal
www.iotpe.com
ijtpe@iotpe.com

June 2012 | Issue 11 | Volume 4 | Number 2 | Pages 24-33

OPTIMAL POWER SYSTEM STABILIZER DESIGN TO REDUCE LOW FREQUENCY OSCILLATIONS VIA AN IMPROVED SWARM OPTIMIZATION ALGORITHM

A. Alfi [1] M. Khosravi [2]

1. Faculty of Electrical and Robotic Engineering, Shahrood University of Technology, Shahrood, Iran
a_alfi@shahroodut.ac.ir
2. Faculty of Electrical and Computer Engineering, Shahrood Branch, Islamic Azad University, Iran
mohsen.khosravi.sh@gmail.com

Abstract- The power system stabilizer (PSS) has been one the most active area of research in power systems. The positive effect of PSS on Low Frequency Oscillations (LFO) damping is obviously clear. Proper designing of PSS can increase the positive effect. So, to enhance of the effectiveness, this paper presents a novel method to reduce LFO. Since the problem of PSS design can be considered as a multi-objective optimization problem, this paper proposes an improved Particle Swarm Optimization (IPSO) algorithm, which is a novel heuristic optimization algorithm, to improve the searching space and convergence speed of the Conventional PSO (CPSO) algorithm. A suitable and comprehensive fitness function is also introduced to cover the wide operating conditions. Thereby, this algorithm is employed to identify the optimal parameters of PSS for Single Machine connected to Infinite Bus (SMIB) system by minimizing the fitness function. Simulation results indicate the superiority of the proposed algorithm.

Keywords: Damping Torque, Stability, Swarm Intelligence, LFO, PSO, PSS, SMIB.

I. INTRODUCTION

The power system stability is considered as one of the most significant concepts of power systems quality. Nonlinearity characteristics, high complexity and time varying behavior of power systems have constructed widespread challenges to stability of the power systems. There are different problems in maintaining the stability of power systems. Power System Stabilizer (PSS) have been broadly utilized to improve the stability of power systems. It has been considered as a key technique to enhance the damping of electromechanical oscillations in power systems.

Over four decades extensive studies have been devoted to the power system stabilizers and the way of designing them. Studying Low frequency Oscillations (LFOs) and its effect on mechanical sections showed that these oscillations can cause boredom especially in rotor section and finally their destruction [1, 2]. Because of these, many approaches have been proposed to control and regulate PSS to reduce LFO such as root locus, sensitivity analysis and pole placement [3, 4]. Heffron and Phillips were the first to present the small perturbation model in terms of constant parameters of a Single Machine connected to Infinite Bus (SMIB) system [5, 6]. Their investigation revealed that the use of modern continuously acting regulators greatly increased the steady-state stability limit of turbine generators in the under-excited region.

DeMello and Concordio developed insights into the effects of thyristor-type excitation system and established understanding of the stabilizing requirement for such system. These stabilizing requirements included the voltage regulator gain parameters as well as the PSS parameters [7]. Kundur et al. described the details of a Delta-P-Omega PSS design for generating unit in Ontario Hydro [8]. Yu and Siggers presented the application of state-feedback optimal PSS while Moussa and Yu proposed an eigenvalue shifting technique for determining the weighing matrix in the performance index [9, 10]. Fleming et al. proposed a sequential eigenvalue assignment algorithm for selecting the parameters of stabilizers in a multi-machine power system [11]. Doi and Abe proposed the coordinate design/tuning of PSS in multi-machine system by combining eigenvalue sensitivity analysis and linear programming [12]. The PSS parameters are determined by minimizing a performance index, which is the sum of all PSS gains.

Since the power system is a highly nonlinear and complex system, these existing approaches have some disadvantage as the parameters of control are adjusted to the certain nominal operating conditions and the control law depends on a linearized machine model. The validation of the controller parameters is also limited while the system conditions changes [13, 14]. Therefore, for better operating performance as than conventional stabilizers, two main approaches to stabilize a power system over a wide range of operating conditions, namely adaptive control [15, 16] and robust control [17-19] have been proposed. These stabilizers have also the certain drawbacks such as they require a comprehensive information about the power system, high computational time for online parameter identification and large implementation costs [20].

Although adaptive control can change the controller parameters online based on the changes in system operating conditions, but if the adaptive controllers are not properly initialized, they have generally poor performance during the learning phase. It is also noticeable that however robust control provides an efficient technique to deal with the uncertainties caused by variations of operating conditions, but it requires a comprehensive search and results in a high order controller. Furthermore, Kharitonov theorem, which is the well-known robust approach, leads to conservative design for PSS [21, 22]. To overcome these shortages, nowadays Artificial Intelligence (AI) techniques have been used to enhance the stabilization of the power systems. Artificial Intelligence (AI) techniques such as fuzzy logic [23] applied to resolve many power system problems and that they could be more effective when properly joined together with conventional mathematical approaches. There are some advantages of the fuzzy logic controllers over the classical controllers. In addition they can be easily implemented in a large scale nonlinear system; they are not so sensitive to the variation of system structure, parameters and operation points as well. Nevertheless, the major shortage of them is the lack of systematic methods to define fuzzy rules and fuzzy membership functions. Most fuzzy rules are also based on human knowledge and differ among persons despite the same system performance. The fuzzy controller requires previous data about a complex system as well. Thus, the implementation of fuzzy logic in the power industry has been limited.

Based on aforementioned before, a novel technique is very essential for the stabilization of power systems. Inspired algorithms such as Genetic Algorithm (GA) [24, 25], Bacterial Foraging Optimization (BFO) [26] Artificial Bee Colony (ABC) algorithm [27] and Tabu Search (TS) [28] algorithm especially with stochastic search techniques have been employed to find optimal value of PSS stabilizer parameters. Modifying damping ratio and damping factor of the lightly damped or un-damped electro-mechanical modes are two objectives. It has been observed that taking just one of the objectives into account may yield to an unacceptable result for other objective.

Recently, PSO algorithm has become available and a promising technique for real world optimization problems. PSO as a population (swarm) based algorithms can solve a variety of difficulties associated with optimization problems. The PSO is initialized with a population of random solutions, which is called particles and searches for optima by updating generations. When it is used for solving the optimization problem, each particle flies through the solution space with a certain trajectory. The trajectory of each individual in the search space is adjusted dynamically altering the velocity of each particle according to flight experiences of its own and the other particles in the search space. Due to this, the algorithm does not require the gradient of objective function to determine the search direction. Because of this, PSO applied to better stability of the power system [29, 30, 31, 32]. To minimize the maximum overshoot, PSO-based technique developed for tuning the parameters of a fixed structure PSS whereas the design of controller was done off-line.

Compared to GA, PSO takes less time for each function evaluation as it does not use many of GA operators like mutation, crossover and selection operator. Due to the simple concept, easy implementation and quick convergence, nowadays PSO has gained much attention and wide applications in different fields [33]. However in the large scale and complex problems, the heuristic algorithm sometimes is easy to be trapped in a local optimum, while the convergence rate decreases considerably in the later period of evolution. In this situation, the algorithm stops optimizing when reaching a solution near the optimum, and thus the accuracy of algorithm is limited [34]. For this reason, the aim of this paper is to present an Improved Particle Swarm Optimization (IPSO) algorithm with fast convergence speed and high accuracy, which is applied to PSS. In this paper, first the IPSO algorithm is introduced by making some improvements in Conventional PSO (CPSO) algorithm. It is utilized to regulate the suggested PSS parameters. Then SMIB system is considered in the presence of different PSSs. Simulation results demonstrate the feasibility of proposed IPSO algorithm.

II. PSO ALGORITHM

A. CPSO Algorithm

The CPSO algorithm is a relatively new generation of combinatorial metaheuristic algorithms, which is fitted for optimizing complex numerical. The fundamental principles of CPSO are adaptability, diverse response, quality, and stability. It has roots in two major component methodologies: (1) evolutionary computation and (2) artificial life such as bird flocking. It lies somewhere in between evolutionary programming and the genetic algorithms. As in evolutionary computation examples, the concept of fitness is employed and candidate solutions to the problem are termed particles or sometimes individuals, each of which adjusts its flying based on the flying experiences of both itself and its companion. The major advantages of PSO are as follows:

(a) The objective function’s gradient is not required.

(b) PSO is more flexible and robust in comparison with traditional optimization methods.

(c) PSO ensures the convergence to the optimal solution.

(d) Compared to GA, PSO takes less time for each function evaluation as it does not use many of GA operators like mutation, crossover and selection operator.

CPSO starts with the random initialization of a swarm of particles in the search space and works on the social behavior of the particles in the swarm. As a result, it finds the global best solution by simply adjusting the trajectory of each particle towards its own best location and towards the best particle of the swarm at each time step (generation). Though, the trajectory of every particle in the search space is adapted by dynamically altering the position and velocity of every particle, according to its own flying experience and the flying experience of the other particles in the search space. The position and velocity of each particle are updated in each iteration according to the following equations:

$$v_i'^d = w v_i^d + c_1 r_1 (x_{pbest_i}^d - x_i^d) + c_2 r_2 (x_{gbest_i}^d - x_i^d) \quad (1)$$

$$x_i'^d = x_i^d + V_i'^d \quad (2)$$

where $x_i'^d$ and x_i^d represent the current and previous positions in the dth iteration of particle i, respectively; $v_i'^d$ and v_i^d are the current and previous velocities of particle i, respectively; x_{pbest_i} and x_{gbest} are the best position found by particle i, so far and the best position found by the whole swarm so far, respectively; $\omega \in (0, 1)$ is an inertia weight, which determines how much the previous velocity is preserved; c_1 and c_2 are positive constant parameters called acceleration coefficients; and r_1 and r_2 are two independent random numbers uniformly distributed in the range of [0, 1].

In CPSO, Equation (4) is utilized to update the new velocity according to its previous velocity and the distance of its current position from both its own personal best position and the global best position. The value of every velocity can be usually bounded to the range $[v_{\min}, v_{\max}]$ to control excessive roaming of the particles outside the search space $[x_{\min}, x_{\max}]$. Then the particle flies toward a new position according to Equation (2). The procedure is repeated until a stopping criterion is reached.

Based on defining the neighborhood for every particle, there are two major models of CPSO algorithm called the global best and local best. In the local best model, the neighborhood of a particle is defined by several fixed particles while in the global best model; the neighborhood of a particle consists of the particles in the whole swarm. Although, these models give different performances on different problems, but global best model has a faster convergence speed and a higher probability of getting stuck in local optima [Poli et al. [35]). The procedure of CPSO is summarized as follows:

Step 1: Initialize a swarm of particles with random positions and velocities.

Step 2: Evaluate the fitness values of all particles, set *pbest* of every particle and its fitness value equal to its current position and fitness value, and set *gbest* and its fitness value equal to the position and fitness value of the best initial particle.

Step 3: Update the velocity and position of every particle according to Equations (1) and (2), respectively.

Step 4: Evaluate the fitness values of all particles.

Step 5: For every particle, compare its current fitness value with the fitness value of its *pbest*. If current value is better, then update *pbest* and its fitness value with the current position and fitness value.

Step 6: Determine the best particle of current whole swarm with the best fitness value. If the fitness value is better than the fitness value of *gbest*, then update *gbest* and its fitness value with the position and fitness value of the current best particle.

Step 7: If a stopping criterion is met, then output *gbest* and its fitness value; otherwise go to Step 3.

B. The Proposed PSO Algorithm

Although CPSO has shown some important advances by providing high speed of convergence in specific problems; however it does exhibits some shortages. It sometimes is easy to be trapped in local optimum, and the convergence rate decreased considerably in the later period of evolution; when reaching a near optimal solution, the algorithm stops optimizing, and thus the achieved accuracy of algorithm is limited [16]. Several modifications have been proposed in literature to improve the performance of CPSO. Most of them are from one of the four categories: swarm topology [36-41], diversity maintenance [42-46], combination with auxiliary operations [47-49], and adaptive PSO [50-54].

Adaptation is the most promising category in PSO. Many approaches are attempted to improve the performance of CPSO by adaption of inertia weight. Empirical studies of PSO with inertia weight have shown that a relatively large inertia weight have more global search ability while a relatively small inertia weight results in a faster convergence. Consequently, the inertia weight decreases as a linear or nonlinear function of iterative generation [55-58]. In addition to efficiently control the local search and convergence to the global optimum solution, time-varying acceleration coefficients were proposed in addition to the time-varying inertia weight factor [46]. Since the search process of PSO is nonlinear and highly complicated, linearly and nonlinearly decreasing inertia weight and acceleration coefficients with no feedback taken from the global optimum fitness cannot truly reflect the actual search process. In fact, if the global fitness is large, the particles are far away from the optimum point. Hence, a big velocity is needed to globally search the solution space and so the inertia weight and acceleration coefficients must be larger values.

Motivated by the aforementioned, in this paper, the inertia weight and acceleration coefficients are set as a function of global optimum fitness during search process of PSO algorithm. Based on this, two modifications are incorporated into the CPSO algorithm that prevents local convergence and provides excellent quality of final result. In this case, these parameters dynamically change according to the rate of global fitness improvement as follows:

$$c_i = 1 + 1/[1 + \exp(-\beta \times F(G_j))^{\alpha}], \quad i = 1, 2 \quad (3)$$

$$\omega = 1/[1 + \exp(-\beta \times F(G_t))^{\alpha}] \quad (4)$$

Initialize the swarm in an *M*-dimensional space // M is the number of system parameters
DO
// fitness evaluation and updating global memories
Evaluate fitness of particles, then:
FOR *i* = 1 to number of particles
IF $f(X_i) < f(P_i)$ **THEN** $P_i = X_i$, $f(P_i) = f(X_i)$
IF $f(X_i) < f(G)$ **THEN** $G = X_i$, $f(G) = f(X_i)$
END FOR
// inertia weight and acceleration coefficients calculation
Calculate ω **using Equation (3)**
Calculate c_1 and c_2 **using Equation (4)**
// updating velocity and positions of particles
Calculate new velocity of the particles using Equation (1)
Calculate new position of the particles using Equation (2)
UNTIL stop criteria is satisfied.

Figure 1. The pseudo-code of proposed PSO

where $F(G_t)$ is the fitness of global optimum in t-th iteration. The parameters α and β need to be predefined. The value of β can be set to the inverse of the value of global optimum fitness in the first iteration, i.e. $\beta = 1/F(G_1)$). Through the study of the nonlinear modulation parameter α and β reasonable set of choice for this parameter is derived within the range (1, 2). Moreover, under the assumption and definition above, it can be concluded that $0.5 \le \omega < 1$, $1.5 \le c_1 < 2$ and $1.5 \le c_2 < 2$. Considering Equations (3) and (4), it is obvious that the bigger global fitness requires the bigger inertia weight and the bigger accelerate coefficients, and vice versa. Therefore, until the fitness of global optimum does not improve significantly, the inertia weight and accelerate coefficients are big since it still needs globally explore the search space to give the algorithm a better ability to rapidly search and move out of the local optima. Conversely, these parameters decrease fast to facilitate finer local explorations since global optimum solution reaches a near optimum. The most important advantages of the proposed algorithm are to achieve faster convergence speed and better solution accuracy with minimum incremental computational burden [59]. Figure 1 illustrates the pseudo-code of proposed PSO.

III. PROBLEM FORMULATION

The stability maintenance in a power system is considered as one of the most significant and essential aspect of power systems quality. In this section, the design procedure is described. Figure 2 shows the system under study, which represents a Single Machine Infinite Bus system (SMIB). The nonlinear equations of the system are given as Equation (5).

$$\begin{aligned}
&\dot{\delta} = \omega_0\, \omega \\
&\dot{\omega} = (T_m - T_e)/M \\
&\dot{E}'_q = \frac{1}{T'_{do}}\left(E_{fd} - \frac{x_d + x_e}{x'_d + x_e} E'_q + \frac{x_d + x'_d}{x'_d + x_e} V \cos\delta \right) \\
&\dot{E}_{fd} = \frac{1}{T_A}\left(K_A\, E_{ref} - K_A\, V_t - E_{fd} \right)
\end{aligned} \tag{5}$$

The above equations can be linearized for small oscillation around an operating point [1, 2, 5, 6] and can be illustrated in the block diagram as shown in Figure 2 as well.

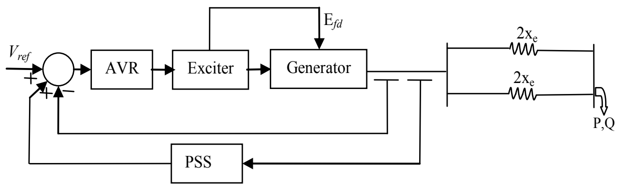

Figure 2. SIMB system

The state variables are defined as follows:

$X = [\Delta\omega \;\; \Delta\delta \;\; \Delta E'_q \;\; \Delta E_{fd}]^T$

Then a SMIB system can be represented in the following state-space form:

$$\begin{aligned}
&\dot{X} = A X + B u \\
&y = C X
\end{aligned} \tag{6}$$

where

$X = [\Delta\omega \;\; \Delta\delta \;\; \Delta E'_q \;\; \Delta E_{fd}]^T$

$$A = \begin{bmatrix} -\frac{D}{M} & -\frac{K_1}{M} & -\frac{K_2}{M} & 0 \\ \omega_0 & 0 & 0 & 0 \\ 0 & -\frac{K_4}{T'_{do}} & -\frac{1}{T'_{do} K_3} & \frac{1}{T'_{do}} \\ 0 & -\frac{K_A K_5}{T_A} & -\frac{K_A K_6}{T_A} & -\frac{1}{T_A} \end{bmatrix}, \tag{7}$$

$$B = \begin{bmatrix} \frac{1}{M} & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & \frac{K_A}{T_A} \end{bmatrix}, C = [0 \;\; 1 \;\; 0 \;\; 0], u = \left[\Delta T_m \;\; \Delta V_{ref} \right]^T$$

The parameters constants K_1 to K_6 represent the system parameters at a certain operation condition [5, 6]. Equation (8) describes the state equations of the system in the presence of PSS.

$$\begin{bmatrix} \Delta\dot{\omega} & \Delta\dot{\delta} & \Delta\dot{E}'_q & \Delta\dot{E}_{fd} & \Delta\dot{N}_1 & \Delta\dot{N}_2 & \Delta\dot{U}_{PSS} \end{bmatrix}^T = \hat{A}.\begin{bmatrix} \Delta\omega & \Delta\delta & \Delta E'_q & \Delta E_{fd} & \Delta N_1 & \Delta N_2 & \Delta U_{PSS} \end{bmatrix}^T$$

$$\hat{A} = \begin{bmatrix} -\frac{D}{M} & -\frac{K_1}{M} & -\frac{K_2}{M} & 0 & 0 & 0 & 0 \\ \omega_0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -\frac{K_4}{T'_{do}} & -\frac{1}{T'_{do}K_3} & \frac{1}{T'_{do}} & 0 & 0 & 0 \\ 0 & -\frac{K_A K_5}{T_A} & -\frac{K_A K_6}{T_A} & -\frac{1}{T_A} & 0 & 0 & \frac{K_A}{T_A} \\ -\frac{D}{M}\frac{T_1}{T_2} & -\frac{K_1}{M}\frac{T_1}{T_2} & -\frac{K_2}{M}\frac{T_1}{T_2} & 0 & -\frac{1}{T_2} & \frac{1}{T_2}-\frac{T_1}{T_2.T_W} & 0 \\ -\frac{D}{M} & -\frac{K_1}{M} & -\frac{K_2}{M} & 0 & 0 & -\frac{1}{T_W} & 0 \\ -\frac{D T_1 K_{PSS} T_3}{M T_2 T_4} & -\frac{K_1 T_1 K_{PSS} T_3}{M T_2 T_4} & -\frac{K_2 T_1 K_{PSS} T_3}{M T_2 T_4} & 0 & \frac{K_{PSS}}{T_4}-\frac{K_{PSS} T_3}{T_2 T_4} & \left(\frac{1}{T_2}-\frac{T_2}{T_2.T_W}\right)\frac{K_{PSS} T_3}{T_4} & -\frac{1}{T_4} \end{bmatrix} \quad (8)$$

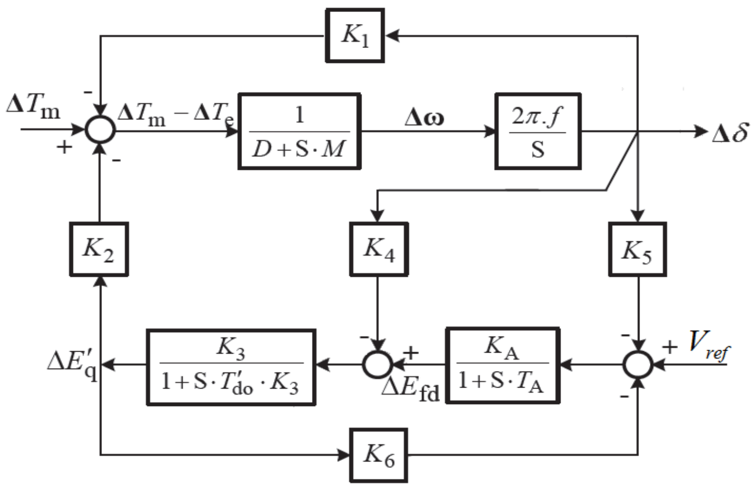

Figure 3. Linearized model of SMIB system

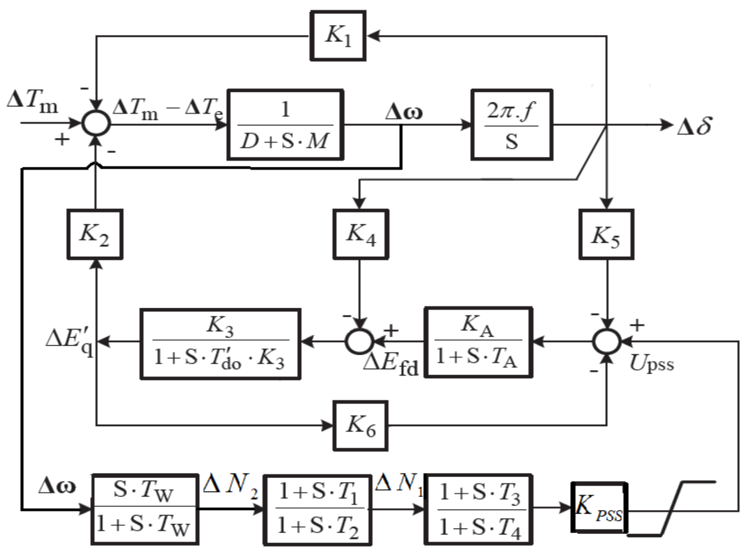

Figure 4. Linearized model of SMIB system with PSS attendance

By considering PSS, Figure 2 can be represented as Figure 3. Recall that a necessary and sufficient condition for the system to be stable is that the eigenvalues of the closed-loop system must be lie in the left hand side of complex s-plane. First of all, the following eigenvalues have been proposed to achieve the least damping of LFO based on LQR method by considering PSS [6]. In this paper, to achieve the desired performance, we also use these eigenvalues values.

$$\mathrm{eig}_{\mathrm{index}}(\hat{A}) = \{-18.62 \quad -11.6 \quad -2.155 \quad -0.987 \quad -0.3124 \pm j6.96 \quad -0.102\} \tag{9}$$

Before proceeding with the optimization operations, a performance criterion or an objective function should be first defined. In general, the heuristic algorithm such as PSO only needs to evaluate the objective function to guide its search and no requirement for derivatives about the system. In this study, the sum of ratio between desired eigenvalues and real eigenvalues is considered as fitness. So, the following fitness function is defined.

$$F = \sum_{R=1}^{3} \sum_{i=1}^{7} \frac{\sigma_{desired}(i)}{\sigma(i)} \tag{10}$$

where $\sigma_{desired}$ and σ is the real part of desired eigenvalues $\mathrm{eig}_{\mathrm{desired}}$ and $\mathrm{eig}(\hat{A})$, respectively. It is noticeable that in order to suitable compare of the corresponding poles, the real part of desired and actual eigenvalues are sorted. In other words, the farthest actual eigenvalue is compared to the farthest desired eigenvalue whereas the nearest actual eigenvalue is compared to the nearest desired eigenvalue. In fact, it confirms that we do not have any unstable eigenvalues.

Now the constraint optimization problem is to find the optimal parameters of PSS (i.e. T_1, T_2, T_3, T_4 and K_{PSS}) whereas the problem constraints are the optimized parameter bounds. Therefore, the design problem can be formulated as the following optimization problem.
minimization F

subject to $T_i^{min} \le T_I \le T_i^{max}$, $K_{pss}^{min} \le K_{pss} \le K_{pss}^{max}$

The proposed approach employs IPSO algorithm to solve this optimization problem and search for the optimal set of PSS parameters. The typical ranges of these parameters are:

$$0.01 \le T_1 \le 1.5,\ 0.01 \le T_3 \le 1.5,\ 0.001 \le T_2 \le 2,\ 0.001 \le T_4 \le 2,\ 10 \le K_{PSS} \le 50 \tag{11}$$

IV. SIMULATION RESULTS

This section is devoted to the assessment of proposed method. The power system stabilization using the proposed IPSO algorithm is evaluated by comparing with several conventional methods in different loading regimes. In order to this, simulation results are carried out in five general cases:
Case 1: SMIB without PSS.
Case 2: SMIB with designed PSS based using LQR method [60].
Case 3: SMIB with designed PSS using lead controller $(G_c = 1.2\frac{1+1.27s}{1+0.092s})$ [4].
Case 4: SMIB with designed PSS using CPSO algorithm.
Case 5: SMIB with designed PSS using the proposed IPSO algorithm

The typical ranges of PSS parameters values are summarized in the appendix. Moreover, to cover the wide operating conditions of machine under study, the three different loading regimes are opted as
(1) Heavy loading regime $(P = 1.2_{p.u.}, Q = 0.2_{p.u.})$
(2) Normal loading regime $(P = 1_{p.u.}, Q = 0_{p.u.})$
(3) Light Loading Regime $(P = 0.7_{p.u.}, Q = 0.3_{p.u.})$

Hence, the proposed controller is designed based on the regimes. Testing the proposed designed controller is also checked on the different operating conditions. The parameters of controllers are tuned using the PSO algorithms by minimizing the fitness function given in Equation (9). To achieve this, a proper choice of the PSO parameters is required. To perform fair comparison, the same computational effort is used in both of the PSO algorithms. Thereby, the population size and maximum generation are considered as 20 and 100, respectively. Moreover, in both CPSO and IPSO algorithms, we set $c_1 = c_2 = 2$ and $V_{\max}$ and $V_{\min}$ are equal to the length of the search space [34]. Furthermore, the inertia weight in CPSO is set to 0.4 [26]. After 100 iterations, the optimized PSS parameters values using IPSO algorithm are determined as follows:
$T_1 = 0.05,\ T_2 = 0.001,\ T_3 = 1.39,\ T_4 = 0.001,\ K_{PSS} = 49$

Simulation results are shown in Figures 5-12. Figure 5 depicts the rotor speed variations $\Delta\omega$ whereas Figure 6 represents the rotor angle variations $\Delta\delta$ after outage the transformation line for the case 1. In addition, Figures 7-12 depict the responses of $\Delta\omega$ and $\Delta\delta$ in different loading regimes for cases 2, 3, 4 and 5 under the fault occurred at $t = 1\sec$, respectively. Referring to Figures 5-12, it can be concluded the effectiveness of the proposed approach to damp out the electromechanical oscillation and enhance the performance of system in the different loading regimes. Although the results of proposed algorithm is better than CPSO algorithm, but significant advantage of proposed PSO is in terms of convergence speed. To confirm this, for instance, Figure 13 depicts the convergence speed for the optimal K_{PSS}.

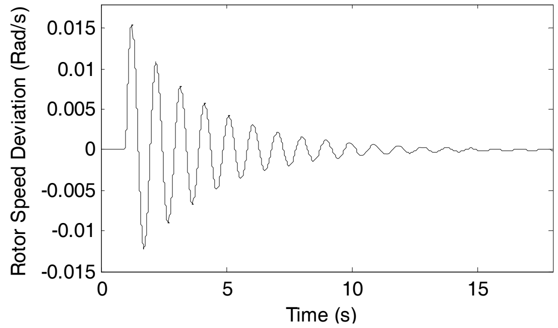

Figure 5. Rotor speed profile without PSS

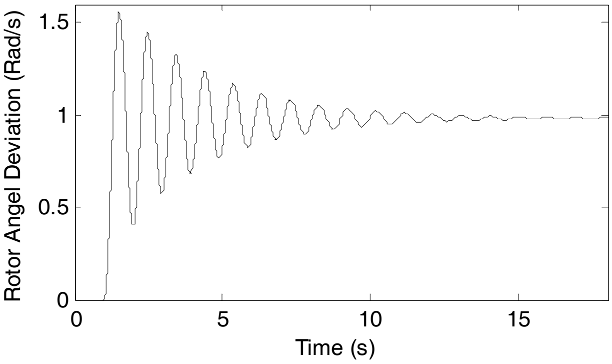

Figure 6. Rotor angel profile without PSS

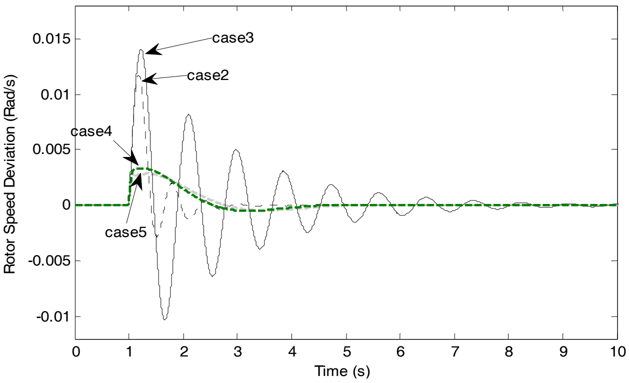

Figure 7. Rotor speed profile in heavy loading regime

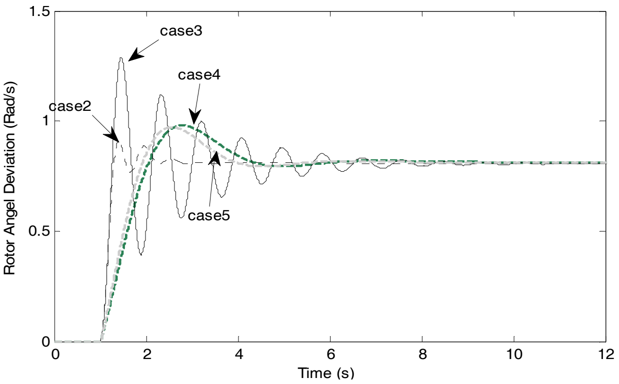

Figure 8. Rotor angel profile in heavy loading regime

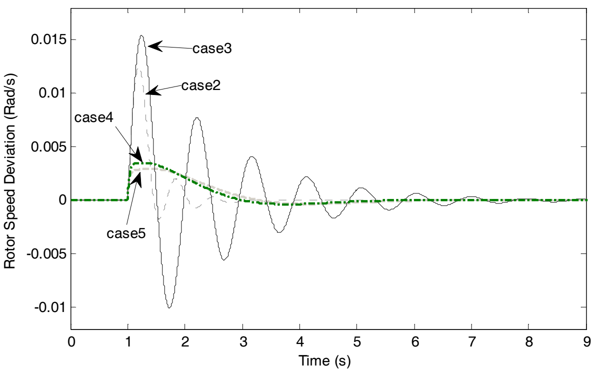

Figure 9. Rotor speed profile in normal loading regime

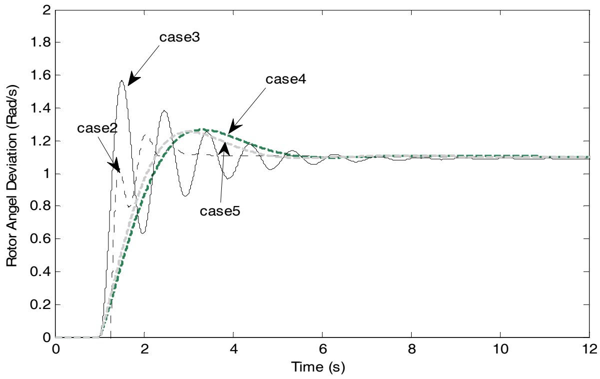

Figure 10. Rotor angel profile in normal loading regime

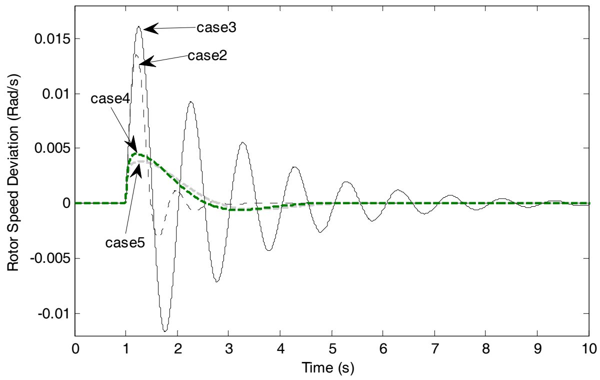

Figure 11. Rotor speed profile in light loading regime

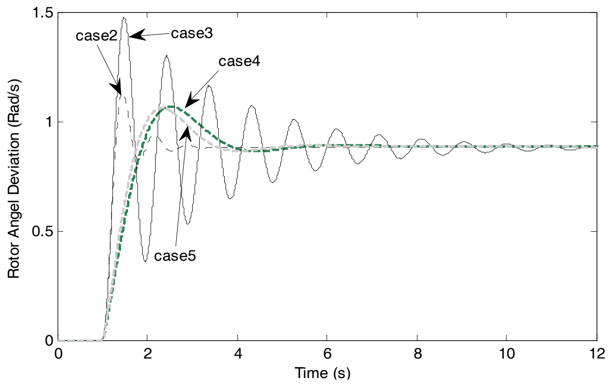

Figure 12. Rotor angel profile in light loading regime

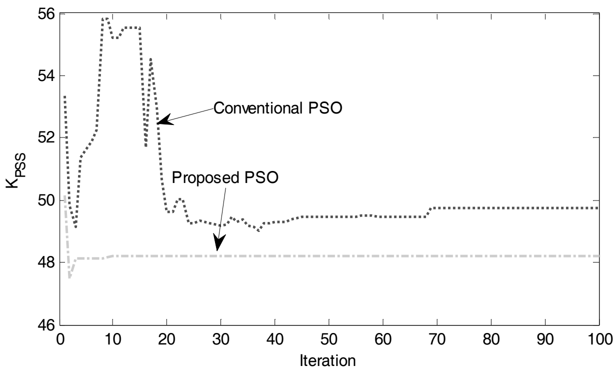

Figure 13. Comparison of convergence speed using CPSO and the proposed IPSO algorithms

V. CONCLUSIONS

The AVR function to regulate voltage can reduce damping torque, the generator's stability limitation and power network. In addition, to eliminating the negative effect of AVR, one can guarantee the network stability by using a feedback from a signal of rotor speed deviations and engaging it in the controlling excitation voltage. This feedback is so-called Power System Stabilizer (PSS) that can improve the stability of network by its proper design and damp the LFO. In this paper, the IPSO algorithm was introduced. This proposed IPSO was utilized to find the optimize parameters of PSS for SMIB system by minimizing the fitness function. Using the proposed algorithm, the LFO can be reduced appropriately. The main advantage of proposed algorithm is to achieve faster convergence speed whereas the appropriate performance of system at different loading conditions was guaranteed. Simulation results demonstrated the effectiveness of developed technique.

NOMENCLATURES

$$K_1 = C_3 \frac{P^2}{P^2 + (Q + C_1)^2} + Q + C_1$$

$$K_2 = C_4 \frac{P}{\sqrt{P^2 + (Q + C_1)^2}}$$

$$K_3 = \frac{x'_d + x_e}{x_d + x_e}$$

$$K_4 = C_5 \frac{P}{\sqrt{P^2 + (Q + C_1)^2}}$$

$$K_5 = C_4 \, x_e \frac{P}{V^2 + Q x_e} \left(C_6 \frac{C_1 + Q}{P^2 + (C_1 + Q)^2} \right)$$

$$K_6 = C_7 \frac{\sqrt{P^2 + (C_1 + Q)^2}}{V^2 + Q x_e} \left(x_e + \frac{C_1 x_q (C_1 + Q)}{P^2 + (C_1 + Q)^2} \right)$$

$$C_1 = \frac{V^2}{x_e + x_q}, \; C_2 = k_3$$

$$C_3 = C_1 \frac{x_q - x'_d}{x_e + x'_d}, \; C_4 = \frac{V}{x_e + x'_d}$$

$$C_5 = \frac{x_d - x'_d}{x_e + x'_d}, \; C_6 = C_1 \frac{x_q (x_q - x'_d)}{x_e + x_q}$$

$$C_7 = \frac{x_e}{x_e + x'_d}$$

T_m : Mechanical torque

T_e : Electrical torque

V_t : Terminal voltage

E_q : Inducted *emf* proportional to field current

E_{fd} : Generator field voltage

V_{ref} : Reference value of generator field voltage

x'_d, x_d, x_q : Generator, d-axis and q-axis synchronous reactances, respectively

x_e : Line reactance

V : Infinite busbar voltage

T'_{do} : Open circuit direct-axis transient time constant

M : Inertia coefficient

D : Damping factor

K_A, T_A : AVR and exciter gain and time constant, respectively

$x_e = 0.4,\; x_q = 1.55,\; x_d = 1.6,\; x'_d = 0.32,\; V = 1,\; f = 50$Hz, $T'_{do} = 6$sec, $M = 10,\; T_A = 0.05$sec, $K_A = 25,\; D = 0$

REFERENCES

[1] P. Kundur, "Power System Stability and Control", McGraw-Hill, New York, 1994.

[2] J. Machowski, J.W. Bialek, J.R. Bumby, "Power System Dynamics and Stability", John Wiley & Sons, 1997.

[3] E.V. Larsen, D.A. Swann, "Applying Power System Stabilizers, P-III", IEEE Trans. on Power Apparatus and Systems, Vol. PAS-100, No. 6, pp. 3017-3046, 1981.

[4] O.P. Malik, G.S. Hope, Y.M. Gorski, V.A. Ushakov, A.L. Rackevich, "Experimental Studies on Adaptive Microprocessor Stabilizers for Synchronous Generators", Proceedings of IFAC Power System and Power Plant Control, pp. 125-130, 1986.

[5] M. Khosravi, M. Banejad, "Design of SVC Controller Based on Fuzzy Logic for Damping of Low Frequency Oscillation in Power System", Proc. of the Int. Conf. on Electrical Engineering, Japan, 2008.

[6] H.M. Soliman, E. Bayoumi, M.F. Hassan, "PSO-Based Power System Stabilizer for Minimal Overshoot and Control Constraints", Journal of Electrical Engineering, Vol. 59, No. 3, pp. 153-159, 2008.

[7] F.P. Demello, C. Concordia, "Concepts of Synchronous Machine Stability as Affected by Excitation Control", IEEE Trans. on Apparatus and System, Vol. 88, No. 4, pp. 316-329, 1969.

[8] P. Kundur, M. Klein, G.J. Rogers, M.S. Zywno, "Application of Power System Stabilizers for Enhancement of Overall System Stability", IEEE Trans. on Power System, Vol. 4, pp. 614-629, 1989.

[9] Y.N. Yu, C. Siggers, "Stabilization and Optimal Control Signals for a Power System", IEEE Trans. on Power Apparatus and System, Vol. PAS-90, pp. 1469-1481, 1971.

[10] H.A. Moussa, Y.N. Yu, "Optimal Power System Stabilization through Excitation and/or Governor Control", IEEE Trans. on Power Apparatus and System, Vol. PAS-91, pp. 1166-1174, 1972.

[11] R.J. Fleming, M.A. Mohan, K. Parvatisam, "Selection of Parameters of Stabilizers in Multi-Machine Power System", IEEE Trans. on Power Apparatus and System, Vol. PAS-100, pp. 2329-2333, 2007.

[12] A. Doi, S. Abe, "Coordinated Synthesis of Power System Stabilizers in Multimachine Power System", IEEE Trans. on Power Apparatus and System, Vol. PAS-103, pp. 1473-1479, 1984.

[13] N.S.D. Arrifano, V.A. Oliveira, R.A. Ramos, "Design and Application of Fuzzy PSS for Power

Systems Subject to Random Abrupt Variations of the Load”, Proc. of American Control Conference, Boston, Massachusetts, pp. 1085-1090, 2004.
[14] N.M. Tabatabaei, M. Shokouhian Rad, “Designing Power System Stabilizer with PID Controller”, International Journal on Technical and Physical Problems of Engineering (IJTPE), Issue 3, Vol. 2, No. 2, pp. 1-7, June 2010.
[15] G. Chen, O. Malik., “Tracking Constrained Adaptive Power System Stabilizer”, Proceedings of IEEE on Generation, Transmission and Distribution, Vol. 142, No. 2, pp. 149-156, 1995.
[16] M. Kothari, L. Bhattacharya, J. Nanda, “Adaptive Power System Stabilizer Based on Pole Shifting Technique”, Proceedings of Generation, Transmission and Distribution, Vol. 143, No. 1, pp. 96-98, 2002.
[17] H. Werner, P. Korba, T.C. Yang, “Robust Tuning of Power System Stabilizers Using LMI Techniques”, IEEE Trans. on Control System Technology, Vol. 11, No. 1, pp. 147-152, 2003.
[18] M. Yue, R.A. Schlueter, “μ-Synthesis Power System Stabilizer Design Using a Bifurcation Subsystem Based Methodology”, IEEE Trans. on Power Systems, Vol. 18, No. 4, pp. 1497-1506, 2003.
[19] R.A. Ramos, L.F.C. Alberto, N.G. Bretas, “A New Methodology for the Coordinated Design of Robust Decentralized Power System Damping Controllers”, IEEE Trans. on Power System, Vol. 19, No. 1, pp. 444-454, 2004.
[20] L.H. Hassan, M. Moghavvemi, H.A.F. Mohamed, “Power System Stabilization Based on Artificial Intelligent Techniques; A Review”, Proc. of the Int. Conf. for Technical Postgraduates, pp. 1-6, 2009.
[21] H.M. Soliman, A. Elshafei, A. Shaltout, M. Morsi, “Robust Power System Stabilizer”, IEE Proceeding Generation, Transmission, and Distribution, Vol. 147, pp. 285-291, 2002.
[22] K.A. El-Metwally, A.L. Elshafei, H.M. Soliman, “A Robust Power System Stabilizer Design Using Swarm Optimization”, International Journal of Modelling, Identification and Control, Vol. 1, No. 4, pp. 263-271, 2006.
[23] S.K. Yee, J.V. Milanovic, “Fuzzy Logic Controller for Decentralized Stabilization of Multi-machine Power Systems”, IEEE Trans. on Fuzzy Systems, Vol. 16, No. 4, pp. 971-981, 2008.
[24] Y.L. Abdel-Magid, M.A. Abido, “Optimal Multiobjective Design of Robust Power System Stabilizers using Genetic Algorithms”, IEEE Trans. on Power Systems, Vol. 18, pp. 1125-1132, 2003.
[25] K. Sebaa, M. Boudour, “Optimal Locations and Tuning of Robust Power System Stabilizer Using Genetic Algorithms”, Electric Power Systems Research, Vol. 79, pp. 406-416, 2006.
[26] S. Mishra, M. Tripathy, J. Nanda, “Multi-Machine Power System Stabilizer by Rule Based Bacteria Foraging”, Electric Power Systems Research, Vol. 77, pp. 1595-1607, 2007.
[27] H. Shayeghi, H.A. Shayanfar and A. Ghasemi, “A Robust ABC Based PSS Design for a SMIB Power System”, International Journal on Technical and Physical Problems of Engineering (IJTPE), Issue 8, Vol. 3, No. 3, pp. 86-92, September 2011.
[28] M.A. Abido, “A Novel Approach to Conventional Power System Stabilizer Design Using Tabu Search”, Electrical Power and Energy Systems, Vol. 21, pp. 443-454, 1999.
[29] A.M. El-Zonkoly, A.A. Khalil, N.M. Ahmied, “Optimal Tuning of Lead-Lag and Fuzzy Logic Power System Stabilizers using Particle Swarm Optimization”, Expert Systems with Applications, Vol. 36, pp. 2097-2106, 2009.
[30] H.M. Soliman, E.H.E. Bayoumi, M.F. Hassan, “PSO-Based Power System Stabilizer for Minimal Overshoot and Control Constraints”, Journal of Electrical Engineering, Vol. 59, No. 3, pp. 153-159, 2008.
[31] A. Safari, H. Shayeghi, H.A. Shayanfar, “Optimization Based Control Coordination of STATCOM and PSS Output Feedback Damping Controller Using PSO Technique”, International Journal on Technical and Physical Problems of Engineering (IJTPE), Issue 5, Vol. 2, No. 4, pp. 6-12, December 2010.
[32] H. Shayeghi, A. Ghasemi, “Application of PSO-TV to Improve Low Frequency Oscillations”, International Journal on Technical and Physical Problems of Engineering (IJTPE), Issue 9, Vol. 3, No. 4, pp. 36-44, December 2011.
[33] M. Clerc, J. Kennedy, “The Particle Swarm-Explosion, Stability and Convergence in a Multidimensional Complex Space”, IEEE Trans. on Evolutionary Computation, Vol. 6, pp. 58-73, 2002.
[34] J. Kennedy, R.C. Eberhart, Y. Shi, “Swarm Intelligence”, Morgan Kaufman, San Francisco, 2001.
[35] R. Poli, J. Kennedy, T. Blackwell, “Particle Swarm Optimization: An Overview”, Swarm Intelligence, Vol. 1, No. 1, pp. 33-58, 2007.
[36] J. Kennedy, “Small Worlds and Mega-Minds: Effects of Neighborhood Topology on Particle Swarm performance”, Proceedings of Congress on Evolutionary Computation, pp. 1931-1938, 2002.
[37] J. Kennedy, R. Mendes, “Population Structure and Particle Swarm Performance”, Proceedings of Congress on Evolutionary Computation, pp. 1671-1676, 2002.
[38] X. Li, “Niching without Niching Parameters: Particle Swarm Optimization Using a Ring Topology”, IEEE Trans. on Evolutionary Computation, Vol. 14, No. 1, pp. 150-169, 2010.
[39] J.J. Liang, A.K. Qin, P.N. Suganthan, S. Baska, “Comprehensive Learning Particle Swarm Optimizer for Global Optimization of Multimodal Functions”, IEEE Trans. on Evolutionary Computation, Vol. 10, No. 3, pp. 281-295, 2006.
[40] R. Mendes, J. Kennedy, J. Neves, “The Fully Informed Particle Swarm: Simpler, Maybe Better”, IEEE Trans. on Evolutionary Computation, Vol. 8, No. 3, pp. 204-210, 2004.
[41] P.N. Suganthan, “Particle Swarm Optimizer with Neighborhood Operator”, Proceedings of Congress on Evolutionary Computation, pp. 1958-1962, 2002.

[42] T.M. Blackwell, P. Bentley, "Don't Push Me! Collision-Avoiding Swarms", Proceedings of Congress on Evolutionary Computation, Vol. 2, pp. 1691-1696, 2002.
[43] T.M. Blackwell, J. Branke, "Multiswarms, Exclusion, and Anticonvergence in Dynamic Environments", IEEE Trans. on Evolutionary Computation, Vol. 10, No. 4, pp. 459-472, 2006.
[44] R. Brits, A. Engelbrecht, F. Van Den Bergh, "A Niching Particle Swarm Optimizer", Proc. of Asia-Pacific Conf. on Simulated Evolution and Learning, Vol. 2, pp. 692-696, 2002.
[45] C. Li, S. Yang, "An Adaptive Learning Particle Swarm Optimizer for Function Optimization", Proceedings on Congress on Evolutionary Computation, pp. 381-388, 2009.
[46] A. Ratnaweera, S.K. Halgamuge, H.C. Watson, "Self-Organizing Hierarchical Particle Swarm Optimizer with Time-varying Acceleration Coefficients", IEEE Trans. on Evolutionary Computation, Vol. 8, No. 3, pp. 240-255, 2004.
[47] P. Angeline, "Evolutionary Optimization versus Particle Swarm Optimization: Philosophy and Performance Differences", Proc. of Conf. on Evolutionary Programming, pp. 601-610, 1998.
[48] R. Poli, C.D. Chio, W.B. Langdon, "Exploring Axtended Particle Swarms: A Genetic Programming Approach", Proc. of Conf. on Genetic Evolutionary and Computation, pp. 33-57, 2005.
[49] C. Wei, Z. He, Y. Zhang, W. Pei, "Swarm Directions Embedded in Fast Evolutionary Programming", Proceedings of Congress on Evolutionary Computation, Vol. 2, pp. 1278-1283, 2002.
[50] S. Hsieh, T. Sun, C. Liu, S. Tsai, "Efficient Population Utilization Strategy for Particle Swarm Optimizer", IEEE Trans. on System, Man and Cybernetics - Part B: Cybernetics, Vol. 39, No. 2, pp. 444-456, 2009.
[51] Y. Shi, R.C. Eberhart, "A Modified Particle Swarm Optimizer", Proc. of IEEE Int. Conf. on Evolutionary Computation, pp. 69-73, 1998.
[52] Y. Shi, R. Eberhart, "Fuzzy Adaptive Particle Swarm Optimization", Proceedings of Congress on Evolutionary Computation, Vol. 1, pp. 101-106, 2002.
[53] Z. Zhan, J. Zhang, Y. Li, H.S. Chung, "Adaptive Particle Swarm Optimization", IEEE Trans. on System, Man and Cybernetics - Part B: Cybernetics, Vol. 39, No. 6, pp. 1362-1381, 2009.
[54] C. Li, S. Yang, T.T. Nguyen, "A Self-Learning Particle Swarm Optimizer for Global Optimization Problems", IEEE Trans. on System, Man and Cybernetics - Part B: Cybernetics, pp. 1-20, 2011.
[55] A. Chatterjee, P. Siarry, "Nonlinear Inertia Weight Variation for Dynamic Adaptation in Particle Swarm Optimization", Computers and Operations Research, Vol. 33, No. 3, pp. 859-871, 2006.
[56] B. Jiao, Z. Lian, X. Gu, "A Dynamic Inertia Weight Particle Swarm Optimization Algorithm", Chaos Solutions and Fractals, pp. 698-705, 2008.
[57] X. Yang, J. Yuan, H. Mao, "A Modified Particle Swarm Optimizer with Dynamic Adaptation", Applied Mathematics and Computation, Vol. 189, pp. 1205-1213, 2007.
[58] Y.P. Chang, C.N. Ko, "A PSO Method with Nonlinear Time-Varying Evolution Based on Neural Network for Design of Optimal Harmonic Filters", Expert Systems with Applications, Vol. 36, pp. 6809-6816, 2009.
[59] H. Modares, A. Alfi, M.M. Fateh, "Parameter Identification of Chaotic Dynamic Systems through an Improved Particle Swarm Optimization", Expert System with Application, Vol. 37, pp. 3714-3720, 2010.
[60] W.L. Brogan, "Modern Control Theory", 3rd Edition, Englewood Cliffs, N.J., Prentice Hall, 1991.

BIOGRAPHIES

Alireza Alfi has received his B.Sc. degree from Ferdowsi University of Mashhad, Mashhad, Iran, in 2000, and his M.Sc. and Ph.D. degrees from Iran University of Science and Technology, Tehran, Iran, in 2002 and 2007, all in Electrical Engineering. He joined Shahrood University of Technology, Shahrood, Iran in 2008, where he is currently an Assistant Professor of Electrical Engineering. His research interests include time delay systems, Robust control, heuristic optimization, and chaotic systems.

Mohsen Khosravi was born in Shahrood, Iran in May 1983. He received the B.Sc. degree in Electrical Engineering in 2006 and the M.Sc. degree in the same field in 2009 from Shahrood University of Technology, Shahrood, Iran. He is now a Ph.D. student in Electrical Engineering of Shahrood University of Technology jointed with Amirkabir University of Technology, Tehran, Iran. His research interests include hysteresis modeling, control, design and modeling of electrical machines.