ADAPTIVE PSO USING PATTERN SEARCH METHOD FOR LOAD FREQUENCY CONTROL

A. PARASSURAM P. SOMASUNDARAM
Research Scholar Associate Professor
Department of Electrical and Electronics Engineering,
Anna University, Chennai, Tamilnadu, India
aparassu@yahoo.co.in

Abstract: *This paper proposes an optimal tuning of Proportional-Integral-Differential (PID) controller in Load Frequency Control (LFC) using Adaptive Particle Swarm Optimization (APSO). The proposed algorithm is based on tuning of Particle Swarm Optimization's (PSO) control parameters by Pattern Search (PS) method. The performance of the proposed method is evaluated using traditional cost function - the integral of time absolute error (ITAE).The simulations are carried out for two area power system with the proposed method. The results show that proposed PID controller improves the performance of the LFC than PSO.*

Keywords: *Load Frequency Control (LIC), Proportional-Integral-Differential (PID), Particle Swarm Optimization (PSO), Pattern Search (PS), Adaptive Particle Swarm Optimization (APSO) and the integral of time absolute error (ITAE).*

1. Introduction

Large-scale power systems with industrial and commercial loads may experience deviations in constant frequency due to random load disturbances. Therefore, there is a need for maintaining frequency control and power control within operating limits. Load Frequency Control (LFC) is important for regulating output power of a generator, reducing the change in frequency in power system and to distribute generation between interconnected areas at economic value [1].

Several literatures addressed about LFC in past decades. A detailed survey on various control methods of LFC for Distributed Generation (DG) system with FACTS devices, storage devices, etc. are done in [4]. Since LFC consist of nonlinear properties, conventional controllers lack to attain zero steady-state condition. Artificial Intelligence helps LFC to operate in nonlinear conditions with many benefits, unlike conventional controllers [4]. Some of the well-known algorithms used in power system are Genetic Algorithm (GA), Particle Swarm Optimization (PSO), Bacterial foraging Algorithm (BFA), Tabu search algorithm (TSA), the Fuzzy logic controller (FLC), Artificial neural network (ANN), etc., [4]. The lozi map based chaotic optimization algorithm was proposed to find optimal gain parameters for PID controller to solve LFC problem [5]. In [6], Fuzzy Gain scheduled Proportional and Integral (FGPI) Controller was proposed for regulating LFC in a multi-area interconnected power system.

Among several optimization algorithms, PSO is one of the few derivative-free algorithms. Kennedy and Eberhart developed PSO algorithm in 1995. Some of the advantages of PSO are that it neither requires any gradient information of the problem nor require an initial point to start. Also, it is easy to form a hybrid with other optimization problem. The application of PSO in various power system problems is addressed in [7]. PSO helps in to obtain optimal solutions even in nonlinear condition [3]. PSO algorithm was implemented in a new technique which is based on the implicit integration trapezoidal rule and the iterative Newton-Raphson method to solve LFC for two area interconnected power system in [8].

Despite many advantages, PSO algorithm has the drawback of escaping local optimal due to premature convergence. The performance of PSO depends on its control parameters such as ω (inertia weight), C_1 and C_2 (Acceleration Coefficients). So proper tuning of control parameters is necessary to avoid the drawbacks of PSO. In 1998, Shi and Eberhart proposed a new method to control PSO parameters known as Adaptive PSO (APSO). Several modified APSO is proposed in the past. In [9], the inertia weight of the PSO is reduced linearly in a dynamic environment, whereas acceleration coefficients are tuned self-adaptively. An adaptive PSO (APSO) is developed in [10] by an evolutionary state estimation (ESE) technique is used to automatically tune control parameters with the help of an elitist learning strategy (ELS). In [11], Adaptive Weighted PSO (AWPSO) algorithm is employed for tuning of PI/PID controller on LFC problem by changing inertia weight (W) randomly and acceleration factor (A) linearly.

This paper proposes an optimal tuning of PID controller in LFC using a new Adaptive Particle Swarm Optimization (APSO). The proposed adaptive PSO technique works by controlling PSO parameters using the Pattern Search (PS) method. Similar to PSO, PS also a derivative-free optimization technique which does not require any information regarding the objective function [12]. In PS algorithm, at each poll, the set of vectors called mesh is multiplied by an integer value called as mesh size. If the poll is successful, this mesh size is increased which is termed as expansion factor. If the poll is unsuccessful, this mesh size is decreased which is termed as contraction factor. Similarly, in APSO, inertia weight and acceleration coefficients are considered as a mesh size whereas velocity, cognitive and social component are considered as a mesh. The mesh size is increased or decreased by comparing the distance between the particle's current position, local best position, and global best position.

The proposed controller is simulated on an interconnected two area power system. The simulation results of the APSO tuned PID controller is compared with PSO to prove the effectiveness of proposed method over traditional method in an LFC problem.

2. Modeling of Power system

Power system consists of several areas interconnected by tie lines. When there is a change in load, it leads to frequency deviation and tie line power deviation. LFC plays an important role in maintaining frequency and tie line power within desired limits in each area. A mathematical model should be designed for efficient control of LFC. Mostly, a thermal power plant is considered for study purpose. The power system in general consists of a governor, a turbine, a generator, a load and a speed governing system. The transfer function of each block of power plant model is as follows [13].

The transfer-function of Turbine:

$$\frac{\Delta P_{mi}}{\Delta P_{Vi}} = \frac{1}{1+sT_{Ti}} \qquad (1)$$

The transfer-function of a governor:

$$\frac{\Delta P_{Vi}}{\Delta P_{Gi}} = \frac{1}{1+sT_{Gi}} \qquad (2)$$

The transfer-function of the generator:

$$\frac{\Delta f_i}{(\Delta P_{Ti} - \Delta P_{Li} - \Delta P_{tie,i})} = \frac{1}{D_i + 2H_i s} \qquad (3)$$

The transfer-function of a speed governing system:

$$\Delta P_{Gi} = \Delta P_{ref,i} - \frac{1}{R_i}\Delta f_i \qquad (4)$$

Where

- T_{Ti} - Turbine time constant for area i
- T_{Gi} - Governor time constant for area i
- D_i - Frequency dependency on the load for area i
- H_i - Per unit inertia constant for area i
- R_i - Governor speed regulation for area i
- ΔP_{Gi} - Incremental change in speed governor output for area i
- ΔP_{mi} - Incremental change in turbine power output for area i
- ΔP_{Vi} - Incremental change in hydraulic actuator output for area i
- $\Delta P_{ref,i}$ - Incremental change in reference set power for area i
- ΔP_{Li} - Incremental change in load for area i
- Δf_i - Incremental change in frequency for area i
- $\Delta P_{tie,i}$ - Incremental change in tie line power for area i

The block diagram of two area power system considered for the study is shown in Figure 1.

3. Load frequency control

The purpose of the LFC is to regulate generator output and to maintain frequency within prescribed limits irrespective of frequency fluctuations. The operation of LFC consists of primary loop control and secondary loop control. In a power system, change in frequency occurs due to change in real power. Primary loop control maintains system frequency to steady state condition by measuring the change in tie line real power. Even though primary loop maintains frequency in steady state condition, the system frequency is not yet attained its nominal set point. Secondary loop reduces frequency by controlling Area Control Error (ACE) [2].

The area control error (ACE) is given by

$$ACE = B\,\Delta f_i + \Delta P_{tie,i} \qquad (5)$$

Where B is the frequency bias parameter.

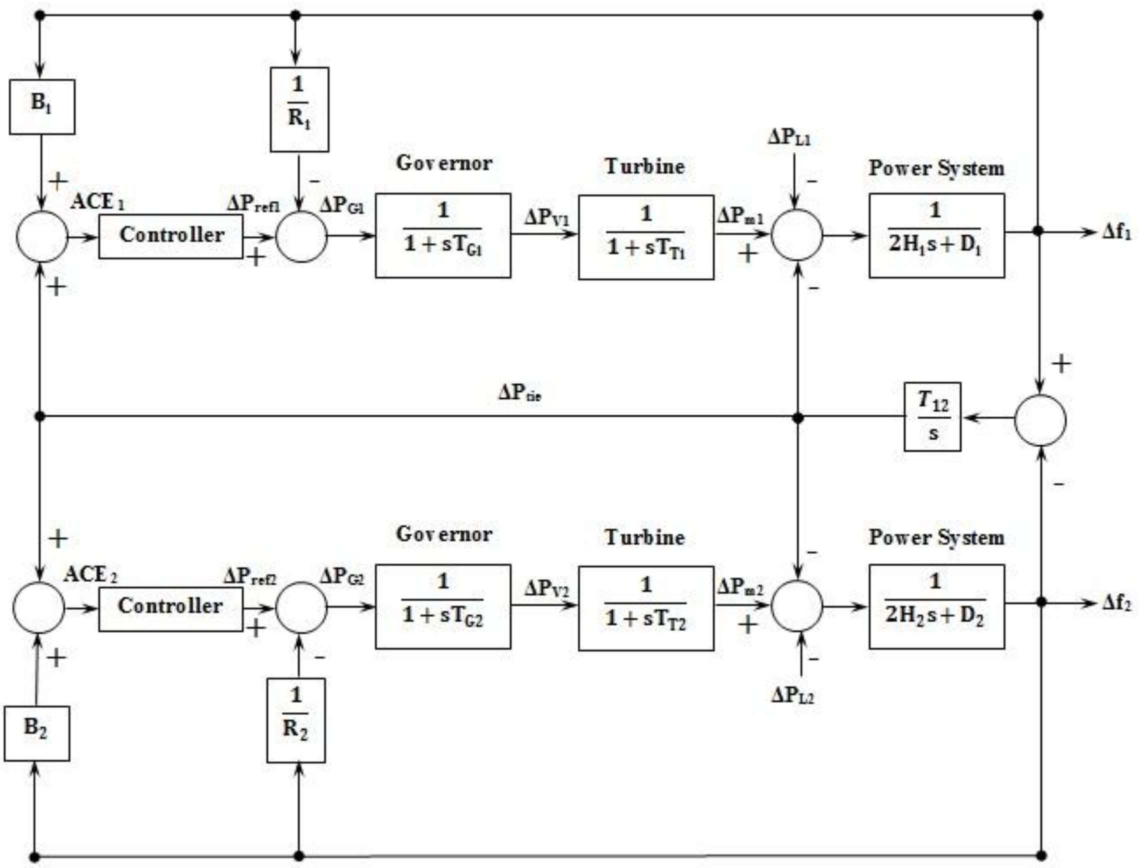

Fig.1. Two area power system

4. PID controller

In this paper, PID controller is used to minimizing ACE. By applying the optimized gain parameters in the PID controller, improved performance with zero steady-state error is achieved.

The function of PID controller is to minimize the error signal which is generated by comparing an output signal of the controlled system and a reference signal. Conventional PID controller consists of three components such as the proportional part, an integral part and the derivative part [14]. The transfer function of the basic PID controller,

$$U_{PID}(s) = K_P + \frac{K_I}{s} + K_D S \qquad (6)$$

Where

U_{PID} : The control signal

K_P, K_I, K_D : The gain parameters

5.Adaptive PSO (APSO) algorithm

5.1 Formulation of Numerical optimization problem

Let the objective function to be minimized is represented as

$$\text{minimize } f(x), \quad x\epsilon\Psi \qquad (7)$$

Where the objective function $f: \mathbb{R}^n \rightarrow \mathbb{R}$, and

$$\Psi \triangleq \{x\epsilon\mathbb{R}^n | l_i \leq x_i \leq u_i, i\epsilon\{1, \cdots n\}\} \qquad (8)$$

$x\epsilon\mathbb{R}^n$ is a vector of decision parameter and it is also known as search space. n is the dimension of search space. The boundary conditions of the constraints are $l\epsilon(-\infty, \mathbb{R})^n$, $u\epsilon(\mathbb{R}, +\infty)^n$ and $l < u$.

5.2 Particle swarm optimization

In 1995, Kennedy and Eberhart introduced particle swarm optimization (PSO) based on the social behavior of bird flocking and fish schooling [15]. In the basic PSO algorithm, the swarm is represented by X_i and each member of the swarm is called particle 'i'. Each particle flies around in the search space with a sufficient velocity which is represented by V_i. It is dynamically updated by the particle's personal information and the information of the particle's neighbors or the information of the entire swarm. Every individual particle has to maintain its knowledge of its movements in the search space, which are related to the optimal position which is called $pbest_i$ and the best particle in the entire swarm is called $gbest_i$. The updated velocity and position of each particle can be calculated using the following formulas:

$$V_i^{k+1} = wV_i^k + c_1 r_1 (pbest_i^k - X_i^k) + c_2 r_2 (gbest_i^k - X_i^k) \quad (9)$$

$$X_i^{k+1} = X_i^k + V_i^{k+1} \quad (10)$$

where

- V_i^k : velocity of particle i at iteration k
- ω : inertia weight parameter
- c_1, c_2 : acceleration coefficients
- r_1, r_2 : random numbers between 0 and 1
- X_i^k : position of particle i at iteration k
- $pbest_i^k$: the best position of particle i until iteration k
- $gbest_i^k$: the best position of the group until iteration k

The velocity update has limits which are given by:

$$V_i^{min} \le V_i^k \le V_i^{max} \quad (11)$$

where

- V_i^{min} : Minimum limit of velocity update
- V_i^k : Current velocity update
- V_i^{max} : Maximum limit of velocity update

The inertia weight ω is calculated as follows:

$$\omega = \omega_{max} - \frac{(\omega_{max} - \omega_{min})}{itermax} \times iter \quad (12)$$

where

- ω_{max} , ω_{min} : Inertia weight limits
- $itermax$: Maximum iteration
- $iter$: current iteration

Procedure of PSO algorithm

1. Initialize random particle positions, velocities, inertia weight limits, acceleration coefficients and random numbers.
2. Evaluate fitness $f(X_i^k)$ at current position X_i^k.
3. If $f(X_i^k)$ is better than $f(pbest_i^{k-1})$ then update $pbest_i^k$.
4. If $min(f(pbest^k))$ is better than $f(gbest^{k-1})$ then update $gbest^k$.
5. Calculate ω using (12).
6. Update velocity V_i^k using equation (9) and check for its limits using (11).
7. Update position X_i^k using equation (10).
8. Stop if maximum iteration is reached or increment the iteration and repeat from step 2 again.
9. The $gbest^k$ at maximum iteration is the global optimal position.

5.3 Pattern search method

In 1961, Hooke and Jeeves developed Pattern search (PS) for solving complex and discontinuous problems. Unlike other optimization algorithms, PS is a derivative-free search method which doesn't require any higher gradient information of the problem. In the basic PS algorithm, a starting point (X0) is initialized at first iteration. This starting point may be random, or user-defined input. Then a set of vector points called mesh is formed which is multiplied by a scalar value (= *1*) which is also known as mesh size [16]. The vector points are defined as [0 1], [1 0], [-1 0] and [0 -1]. The starting point is now added to the product of mesh and scalar value to form the mesh points as X0+ [1 0], X0+ [0 1], X0+ [-1 0], X0+ [0 -1]. The mesh points are updated for next iteration according to following two cases.

Case 1:

If any of current mesh's fitness value is lesser than the starting point's fitness value, then the poll is said to be successful. For next iteration (i.e., iteration = *2*), the mesh point which is better than previous starting point is set as new starting point X1. Also, current mesh size is multiplied by an expansion factor (by default it is *2*). The mesh points for next iteration will look like 2*[1 0] + X1, 2*[0 1] + X1, 2*[-1 0] + X1and 2*[0 -1] + X1.

Case 2:

If none of current mesh's fitness value is lesser than the starting point's fitness value, then the poll is said to be unsuccessful. For next iteration (i.e., iteration = *2*), the starting point doesn't change i.e., X1=X0. Also current mesh size is multiplied by a contraction factor (by default it is *0.5*). The mesh points for next iteration will look like 0.5*[1 0] + X1, 0.5*[0 1] + X1, 0.5*[-1 0] + X1and 0.5*[0 -1] + X1.

The algorithm then repeats the previous steps until it attains stopping condition such as a maximum number of iterations or a set tolerance. The current point when the algorithm reaches stopping condition is the optimal solution.

5.4 Proposed adaptive PSO algorithm

Particle swarm optimization (PSO) is a well-known optimization algorithm for its capability to solve complex nonlinear problems and good at global optimal search. But PSO has the disadvantage of premature convergence which leads to poorer local search [17]. On the other hand, Pattern Search (PS) method has strong local search ability but needs an arbitrary starting point to start the algorithm [18]. The goal of the proposed algorithm is to improve searchability of PSO by adaptively tuning the inertia weight (w) and acceleration coefficients (C_1, C_2) using Pattern search methodology. As both PSO and PS are simple and derivative-free optimization technique, it will be easy to incorporate one's advantage with other.

In proposed APSO, inertia weight and Acceleration Coefficients of PSO are considered as mesh size as in PS whereas velocity, cognitive and social component are considered as mesh points. Here, only the mesh sizes are updated at each poll, unlike in PS both mesh size and mesh points are updated depending on each poll successful and unsuccessful condition.

The inertia weight, acceleration coefficients and updated velocity of each particle for proposed APSO can be calculated using the following formula:

$$\omega_p = \begin{cases} \omega_p \times EP, & if\ f\ (X^k) < f(gbest^k) \\ \omega_p \times CP, & otherwise \end{cases} \tag{13}$$

$$c_{1p} = \begin{cases} c_{1p} \times EP, & if\ f\ (X^k) < f(pbest^k) \\ c_{1p} \times CP, & otherwise \end{cases} \tag{14}$$

$$c_{2p} = \begin{cases} c_{2p} \times EP, & if\ f\ (pbest^k) < f(gbest^k) \\ c_{2p} \times CP, & otherwise \end{cases} \tag{15}$$

$$V_i^{k+1} = \omega_p V_i^k + c_{1p}\left(pbest_i^k - X_i^k\right) + c_{2p}\left(gbest_i^k - X_i^k\right) \tag{16}$$

where

ω_p : Inertia weight parameter of proposed algorithm

c_{1p},c_{2p}: Acceleration coefficients of proposed algorithm

EP : Expansion factor

CP : Contraction factor

Procedure of APSO algorithm

1. Initialize random particle positions, velocities, expansion factor, contraction factor, initial acceleration coefficients and inertia weight.
2. Evaluate fitness $f\ (X_i^k)$ at current position X_i^k.
3. If $f\ (X_i^k)$ is better than $f\ (pbest_i^{k-1})$ then update $pbest_i^k$.
4. If $min\ (f\ (pbest^k))$ is better than $f\ (gbest^{k-1})$ then update $gbest^k$.
5. Update c_{1p}, c_{2p} and ω_p using equations (13), (14) and (15).
6. Update velocity V_i^k using equation (16) and check for its limits using equation (11).
7. Update position X_i^k using equation (10).
8. Stop if maximum iteration is reached or repeat from step 2 again.
9. The $gbest_i^k$ at maximum iteration is the global optimal position.

6 Objective function

To analyze or to evaluate the performance of the PID controller, error criterion are used as the objective function.

Integral of Time multiplied by Absolute Errors (ITAE) is used as error criterion in this paper since ITAE is highly sensitive to error. The sum of absolute of incremental change in frequency of each area and tie line power is considered as an error. Therefore, the objective function is defined as follows.

$$F = \int_0^t t.\left(|\Delta f_1(t)| + |\Delta f_2(t)| + |\Delta P_{tie}|\right)dt \tag{17}$$

7 Results and Comparison

A two area power system with frequency=60 Hz and MVA base=1000 were considered for the study. The system parameters are provided in Table 1 [5].

Table 1

Parameters	Area 1	Area 2
H (sec)	5	4
D (pu MW/Hz)	0.6	0.9
T_G (sec)	0.2	0.3
T_T (sec)	0.5	0.6
R (Hz/ pu MW)	0.05	0.0625
B (pu MW/Hz)	20.6	16.9

7.1 Parameter setting for algorithm

The common settings for both algorithms are

- Population size: 100
- Maximum iteration: 50
- Particle length:

  X_{min}= 0 and X_{max} = 1

7.1.1 PSO parameters

- Inertia weight limits:

  ω_{max}= 0.5 and ω_{min}= 0.1
- Acceleration coefficients:

  c_1 = 2 and c_2 = 1

7.1.2 APSO Parameters

- Expansion factor (EP): 2
- Contraction factor (CP): 0.5

7.2 Analysis of system performance

Two interconnected thermal power plant is taken as a test system. The test system is simulated by applying a 2% step load change in area1. The simulation model and program of the system under study has been developed in MATLAB software. The optimized parameters for PID controller using PSO and proposed APSO with objective function ITAE are shown in Table 2.

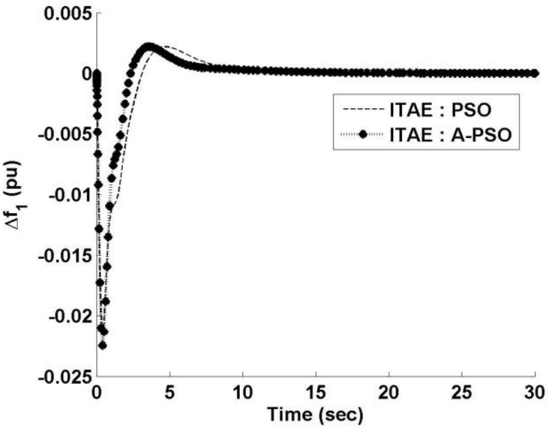

Fig.2. Comparison of Frequency deviation in area1 (for 2% step load change in area 1)

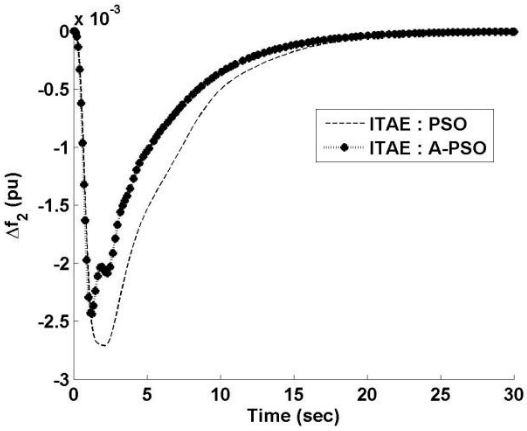

Fig.3. Comparison of Frequency deviation in area 2 (for 2% step load change in area 1)

The Figures 2, 3 and 4 shows the comparison of PSO and proposed APSO algorithm for frequency deviation in area 1, area 2 and tie line power deviation. The comparison of system performance using rise time, settling time and overshoot of PSO and proposed APSO algorithm are shown in Table 3.

The proposed PID controller improves the system performance and overall efficiency compared to the conventional PID controller. The settling time and overshoot are decreased to a great extent.

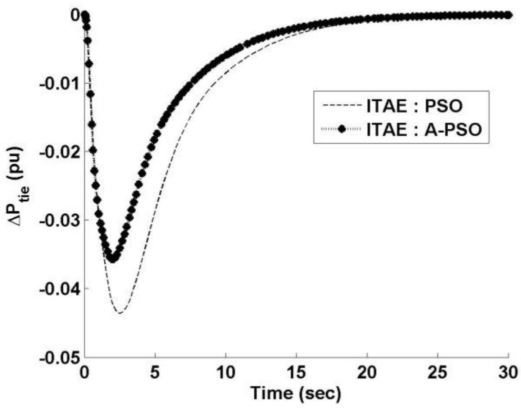

Fig.4. Comparison of Tie line power deviation (for 2% step load change in area 1)

7.3 Analysis of algorithm parameters

According to researchers, the inertia weight ω should be linearly decreasing such that ω is large for global search and small for local search [10]. In this paper, ω is decreased from 0.5 to 0.1 over time for PSO as shown in Figure 5 (as per equation (12)).

But it is also proved that ω need not be always decreased as pure linear for a better result. The proposed method in this paper adaptively tunes ω as shown in Figure 6. The value of ω is large at initial for global search i.e., exploration state (up to 5 iterations), and then it decreases rapidly as

Table 2

Algorithm	PSO			APSO		
Controller gain parameters	K_p	K_i	K_d	K_p	K_i	K_d
Area 1	1.4548	1.5669	1.5536	1.6808	2.1363	1.4140
Area 2	0.8186	1.9399	1.4327	2.2635	2.5273	1.8242

iteration is increased for local search i.e., exploitation state.

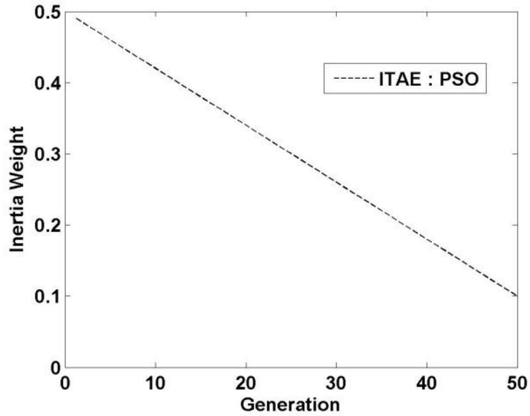

Fig.5. Inertia weight of PSO

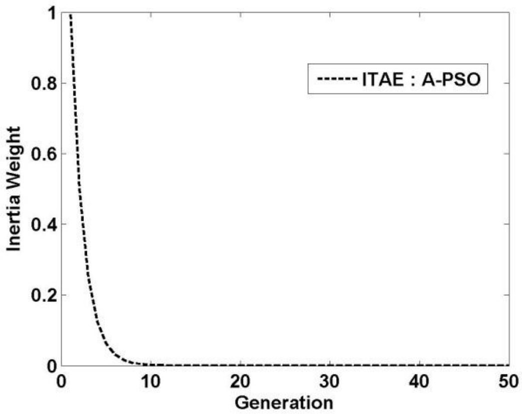

Fig.6. Inertia weight of APSO

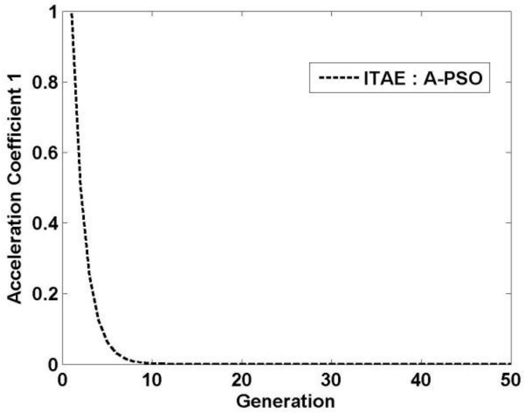

Fig.7. Acceleration Coefficient 1 of APSO

The acceleration coefficients c_1 and c_2 are responsible for particles attraction toward local best (pbest) and global best (gbest) positions. In most of the cases, the acceleration coefficients are set as 2.0 [19]. In this paper, Acceleration coefficients c_1 and c_2 are adaptively changed. The acceleration coefficients of the proposed APSO algorithm are shown in figures 7 and 8 respectively. c_1 is decreased rapidly and c_2 is increased slowly at initial iteration for good exploration (local search). c_2 is then increased linearly for good exploitation (global search). At the end, c_1 is smaller and c_2 is larger for good convergence.

From Figure 9, it is evident that the fitness function value converges smoothly to the optimum value without any hasty oscillations and further it is also proved that the APSO algorithm has much faster convergence than PSO algorithm.

Table 3

Algorithm	PSO			APSO		
Performance Parameters	Settling time (sec)	Overshoot (%)	Rise time (sec)	Settling time (sec)	Overshoot (%)	Rise time (sec)
Area 1	9.4488	10.0484	0.0012	*7.4363*	*9.8464*	*0.0008*
Area 2	19.8771	0.2439	0.0526	*18.7604*	*0.1905*	*0.0482*
Tie line power	20.0172	0.2533	0.0207	*19.3148*	*0.2177*	*0.0176*
Fitness Function Value	*1.5301*			*0.0689*		

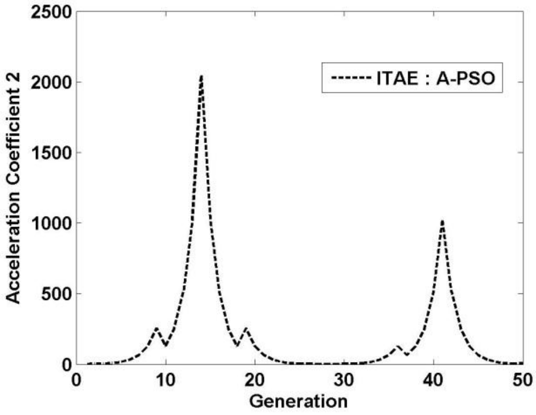

Fig.8. Acceleration Coefficient 2 of APSO

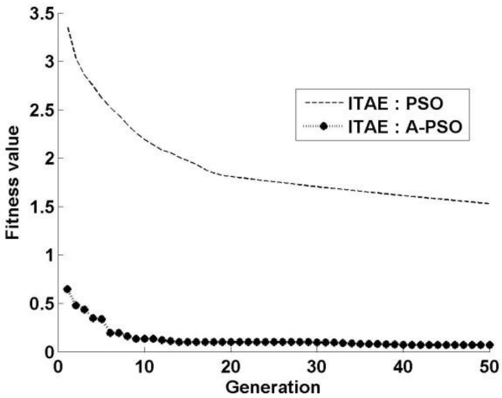

Fig.9. Comparison of Fitness function value

8 Conclusion

In this paper, an adaptive particle swarm optimization based LFC has been examined for a two area power system. The simulations are conducted using MATLAB Simulink. The proposed adaptive PSO is developed based on the pattern search algorithm. The comparison of results shows that PID controller using APSO had lesser rise time, settling time and lower overshoot. Also proposed APSO's faster convergence to optimal solution shows algorithms ability to avoid premature convergence than PSO. This proves APSO based PID controller is better for LFC than other conventional methods.

Acknowledgement

The author would like to thank faculty members of Anna University, Chennai for their support in this project work.

References

1. Chaturvedi D.K., Man Mohan, Rahul Umrao, Sanjeev Kumar.: *Load Frequency Control Methodologies for Power System*. In: Proceedings of the 2nd International Conference on Power, Control and Embedded Systems ICPCES, Dec 17 – 19, 2012, Allahabad, India, p.585–594.
2. Saadat, H: *Power system analysis*, In: Electrical and computer engineering, McGraw-Hill, New York, 1999.
3. Saumya Kr., Gautam, Nakul Goyal.: *Improved Particle Swarm Optimization Based Load Frequency Control In A Single Area Power System*. In: Proceedings of the Annual IEEE India Conference INDICON, Dec 17 – 19, 2010, Kolkata, India, p.508–511.
4. Nand Kishor, Shashi Kant Pandey, Soumya Mohantya, R.: *A literature survey on load–frequency control for conventional and distribution generation power systems*. In: Renewable and Sustainable Energy Reviews, 25 (2013), May 2013, p. 318–334.
5. Farahani, M., Ganjefar, S., Alizadeh, M.: *PID controller adjustment using chaotic optimization algorithm for multi-area load frequency control*. In: IET Control Theory and Applications, 6 (2012), No. 13, Sep 2012, p. 1984–1992.
6. Kocaarslan, I., Cam, E.: *Fuzzy logic controller in interconnected electrical power systems for load-frequency control*. In: Electrical Power and Energy Systems, 27 (2005), No. 8, Oct 2005, p. 542–549.
7. AlRashidi, M.R., El-Hawary, M.E.: *A Survey of Particle Swarm Optimization Applications in Electric Power Systems*. In: IEEE Transactions on Evolutionary Computation, 13 (2009), No. 4, Aug 2009, p. 913 – 918.
8. Bouziane Boussahoua, Mohamed Boudour, Mohamed Menaa, Mourad Hasni, Nour EL Yakine Kouba.: *Optimal Load Frequency Control in Interconnected Power System Using PID Controller Based on Particle Swarm Optimization.* In: Proceedings of the International Conference on Electrical Sciences and Technologies in Maghreb CISTEM, Nov 3 – 6, 2014, Tunis, Tunisia, p.638–645.
9. Niknam, T., Golestaneh, F.: *Enhanced adaptive particle swarm optimization algorithm for dynamic economic dispatch of units considering valve-point effects and ramp rates*. In:

IET Generation, Transmission & Distribution, 6 (2012), No. 5, May 2012, p. 424–435.
10. Zhi-Hui Zhan, Jun Zhang, Yun Li, Henry Shu-Hung Chung.: *Adaptive Particle Swarm Optimization*. In: IEEE Transactions on Systems, Man, and Cybernetics—Part B: Cybernetics, 39 (2009), No. 6, Dec 2009, p. 1362-1381.
11. Rania Helmy Mansour Hennas, Magdy A.S. Aboelela, Hassen T. Dorrah.: *Development of Distributed LFC Controllers Using Adaptive Wieghted PSO Algorithm.* In: Journal of Electrical Engineering, 16 (2016), No. 1, p. 55 - 60.
12. Al-Sumait, J.S., AL-Othman, A.K., Sykulski, J.K.: *Application of pattern search method to power system valve-point economic load dispatch*. In: Electrical Power and Energy Systems, 29 (2007), No. 10, Dec 2007, p. 720–730.
13. Elgerd, O.I: *Electric energy systems theory, in: An introduction.* In: Electric power systems, McGraw-Hill, New Delhi, 1983.
14. dos Santos Coelho, L.: *Tuning of PID controller for an automatic regulator voltage system using chaotic optimization approach*. In: Chaos, Solitons and Fractals, 39 (2009), No. 4, Feb 2009, p. 1504–1514.
15. Zwe-Lee Gaing, *A Particle Swarm Optimization Approach for Optimum Design of PID Controller in AVR System*. In: IEEE Transactions on Energy Conversion, 19 (2004), No. 2, June 2004, p.384–391.
16. Jianhua Yuan, Junling Hao, Wenqiao Lin, Xinchao Zhao, Xingquan Zuo.: *Clustering and pattern search for enhancing particle swarm optimization with Euclidean spatial neighborhood search*. In: Neurocomputing, 171 (2016), Jan 2016, p. 966–981.
17. Xue-mei Wang, Yi-zhuo Guo, Gui-jun Liu.: *Self-adaptive Particle Swarm Optimization Algorithm with Mutation Operation based on K-means.* In: Proceedings of the 2nd International Conference on Computer Science and Electronics Engineering ICCSEE, 2013, Hangzhou, China, Vol. 4, p.2996–2999.
18. Vaz, A.I.F., Vicente, L.N.: *A particle swarm pattern search method for bound constrained global optimization*. In: Journal of Global Optimization, 39 (2007), No. 2, Oct 2007, p. 197–219.
19. Banaja Mohanty, Hota, P.K., Sidhartha Panda.: *Hybrid BFOA–PSO algorithm for automatic generation control of linear and nonlinear interconnected power systems*. In: Applied Soft Computing, 13 (2013), No. 12, Dec 2013, p.4718–4730.