Design of PID Controller for Power System Stabilization Using Hybrid Particle Swarm-Bacteria Foraging Optimization

H. I. ABDUL-GHAFFAR
Post graduate student
Minia University
Faculty of Engineering, Minia
EGYPT
abdelghafar_ibrahim@yahoo.com

E. A. EBRAHIM
Power Electronics Department
Electronics Research Institute
El-Tahrir St., Dokki, Cairo
EGYPT
essamudin@yahoo.com

M. AZZAM
Electrical Engineering Department
Minia University
Faculty of Engineering, Minia
EGYPT
Azzam126@gmail.com

*Abstract***:-** This paper considers the stabilization of a synchronous machine connected to an infinite bus via a PID. The PID parameters are tuned using hybrid Particle Swarm-Bacteria Foraging Optimization (PSO-BFA). Simulation results are introduced with and without the proposed controller. Also, a comparison study is introduced when using classical PID, only bacteria foraging optimization and when using hybrid particle swarm- bacteria foraging optimization. The results show that using of hybrid control is capable of guaranteeing the stability and performance of the power system better than the PID-PSS based classical PID and BFA only.

Key words:- Bacteria Foraging Algorithm, Particle Swarm Optimization , PID Controller, PSS, Chemotaxis, Mutation, Objective function.

1 Introduction

The electrical energy is a primary prerequisite for economic growth. The demand for electrical energy has greatly increased due to large-scale industrialization. Modern power system operates under much stressed conditions because of growth in demand and deregulation of electric power system. This leads to many problems associated with operation and control of power systems. The economics of power generation has a major concern for the power utilities. Therefore, the power utilities always need new technology to solve its problems [1].

The complexity of power systems is continuously growing due to the increasing number of generation plants and load demand. Power systems are becoming heavily stressed due to the increased loading of the transmission lines and due to the difficulty of constructing new transmission systems as well as the difficulty of building new generating plants near the load centers. All of these problems lead to the voltage stability problem in the system [2].

An interconnected power system basically consists of several essential components. They are namely the generating units, the transmission lines and the loads. During the operation of the generators, there may be some disturbances such as sustained oscillations in the speed or periodic variations in the torque that is applied to the generator. These disturbances may result in voltage or frequency fluctuation that may affect the other parts of the interconnected power system. External factors, such as lightning, can also cause disturbances to the power system. All these disturbances are termed as faults. When a fault occurs, it causes the generators to lose synchronism. With these factors in mind, the basic condition for a power system with stability is synchronism. Besides this condition, there are other important conditions such as steady-state stability, transient stability, harmonics and disturbance, collapse of voltage and the loss of reactive power [3].

The stability of a system is defined as the tendency and ability of the power system to develop restoring forces equal to or greater than the disturbing forces to maintain the state of equilibrium [4]. There are many major blackouts caused by instability of a power system which illustrates the importance of this phenomenon [5]. The stability has been acknowledged as an important problem for secure system operation since the 1920’s [6].

Damping of power system oscillation between interconnected areas is very important for the system secure operation. Power system stabilizer (PSS) is the most widely used device for resolving oscillatory stability problems [7], and to enhance the power system damping. Traditionally, lead-lag structures have been used as power system stabilizers. Many researches had been published explaining the ways of tuning the parameters of the lead lag controller. The methods used for tuning range from pole placement, to the more recent one using the heuristic optimization techniques such as Genetic Algorithms (GAs) [8], Tabu Search Algorithm (TSA) [9], Simulated Annealing (SA) [10], Particle Swarm Optimization (PSO) [11], and Bacteria Foraging Algorithm (BFA) [12].

The PID controller is a well-established type of controller and has been in use for a long time. Tuning PID controllers are traditionally tuned using standard techniques such as the root locus , and classical PID controllers which tuned by “Ziegler-Nichols” methods [13].

This paper produces a design method for the stability enhancement of a single machine infinite bus power system using PID-PSS which its parameters are tuned by Hybrid Particle Swarm-Bacteria Forging Optimization method. The advantage of tuning the parameters of the optimum controller is that the possibility of including time-domain specifications such as rise time, maximum overshoot, damping ratio, and steady-state error.

This paper will be organized as follows: synchronous machine model is presented in section 2. Section 3, discusses the bacteria foraging algorithm (BFA) while the particle swarm optimization (PSO) is introduced in section 4. In section 5, hybrid particle swarm–bacteria foraging optimization(PS-BFA) is introduced. PS-BFA based PID tuning for PSS in section 6 is obtained. Section 7, highlights the objective function. The implementation of the PSO-BFA based PID PSS is discussed in section 8. Some concluding remarks are highlighted in section 9. About twenty two research publications are reviewed, discussed, classified, and appended for a quick reference.

2 Synchronous machine model

The system under study in this work considered as a single machine connected to an infinite bus system through a transmission line as shown in Fig. 1.

Fig. 2 shows the block diagram model of the system. This model is known as Heffron-Phillips model [14]. In this model, The synchronous machine is described by a 4th –order model. The relations in the block diagram apply to a two-axis

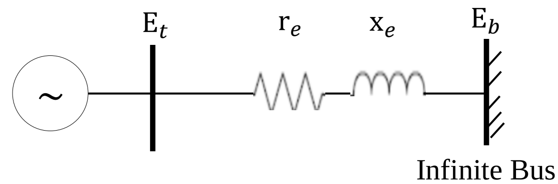


Fig. 1 A single machine infinite bus power system

machine representation with a field circuit in the direct axis but without damper windings. The interaction between the speed and voltage control equations of the machine is expressed in terms of six constants K1- K6 which depend on the real and reactive loading of the machine except for K3. The linearized equations describing the system of Fig.1 are given below:

$$\Delta\dot{\delta} = \omega_0 \Delta\omega$$

$$\Delta\dot{\omega} = \frac{1}{M}\left(-K_1\Delta\delta - D\Delta\omega - K_2\Delta\grave{E}_q\right)$$

$$\Delta\dot{\grave{E}}_q = \frac{1}{\grave{T}_{d0}}\left(-K_4\Delta\delta - \frac{\Delta\grave{E}_q}{K_3} + \Delta E_{fd}\right)$$

$$\Delta\dot{E}_{fd} = \frac{1}{T_e}\left(-k_e k_5\Delta\delta - k_e k_6\Delta\grave{E}_q - \Delta E_{fd} + k_e u\right) \quad (1)$$

Equation (1) can be rewritten in the state space form as given below:

$$\dot{x} = Ax + Bu \quad (2)$$

Where A is the system matrix and B is the input matrix. The model of the system in the state space form without any controllers is obtained in equation (3) [14].

$$\begin{bmatrix} \Delta\dot{\delta} \\ \Delta\dot{\omega} \\ \Delta\dot{E_q} \\ \Delta\dot{E_{fd}} \end{bmatrix} = \begin{bmatrix} 0 & \omega_0 & 0 & 0 \\ \frac{-K_1}{M} & 0 & \frac{-K_2}{M} & 0 \\ \frac{-K_4}{T_{d\grave{0}}} & 0 & \frac{-1}{K_3 T_{d\grave{0}}} & \frac{1}{T_{d\grave{0}}} \\ \frac{K_e K_5}{T_e} & 0 & \frac{-K_e K_6}{T_e} & \frac{-1}{T_e} \end{bmatrix} \times \begin{bmatrix} \Delta\delta \\ \Delta\omega \\ \Delta E_q \\ \Delta E_{fd} \end{bmatrix} +$$

$$\begin{bmatrix} 0 & 0 \\ \frac{1}{M} & 0 \\ 0 & 0 \\ 0 & \frac{K_e}{T_e} \end{bmatrix} \times \begin{bmatrix} \Delta P_m \\ \Delta V_{ref} \end{bmatrix} \quad (3)$$

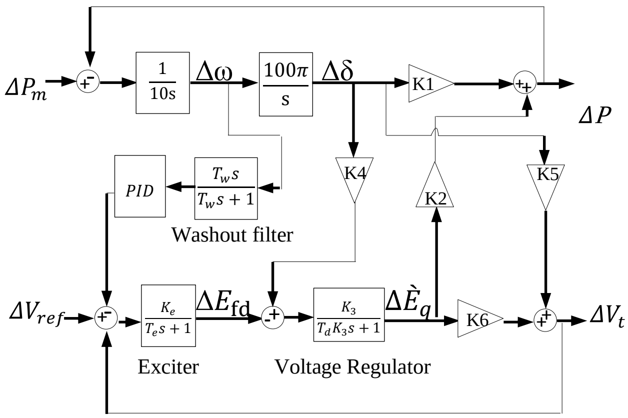


Fig. 2 Heffron-Phillips block diagram

3 Bacterial Foraging Optimization

3.1 A brief overview

The survival of species in any natural evolutionary process depend upon their fitness criteria, which relies upon their food searching and motile behavior. The law of evolution supports those species who have better food searching ability and either eliminates or reshapes those with poor search ability. The genes of those species who are stronger gets propagated in the evolution chain since they possess ability to reproduce even better species in future generations. So, a clear understanding and modeling of foraging behavior in any of the evolutionary species, leads to its application in any non-linear system optimization algorithm. The foraging strategy of Escherichia Coli bacteria present in human intestine can be explained by four processes, namely chemotaxis, swarming, reproduction, elimination–dispersal [15].

3.2 Chemotaxis

The characteristics of movement of bacteria in search of food can be defined in two ways, i.e. swimming and tumbling together known as chemotaxis. A bacterium is said to be 'swimming' if it moves in a predefined direction, and 'tumbling' if moving in an altogether different direction. Mathematically, tumble of any bacterium can be represented by a unit length of random direction $\varphi(j)$ multiplied by step length of that bacterium $C(i)$. In case of swimming, this random length is predefined.

3.3 Swarming

For the bacteria to reach at the richest food location (i.e. for the algorithm to converge at the solution point), it is desired that the optimum bacterium till a point of time in the search period should try to attract other bacteria so that together they converge at the desired location (solution point) more rapidly. To achieve this, a penalty function based upon the relative distances of each bacterium from the fittest bacterium till that search duration, is added to the original cost function. Finally, when all the bacteria have merged into the solution point, this penalty function becomes zero. The effect of swarming is to make the bacteria congregate into groups and move as concentric patterns with high bacterial density.

3.4 Reproduction

The original set of bacteria, after getting evolved through several chemotactic stages reach the reproduction stage. Here, best set of bacteria (chosen out of all the chemotactic stages) get divided into two groups. The healthier half replaces with the other half of bacteria, which gets eliminated, owing to their poorer foraging abilities. This makes the population of bacteria constant in the evolution process.

3.5 Elimination and dispersal

In the evolution process, a sudden unforeseen event can occur, which may drastically alter the smooth process of evolution and cause the elimination of the set of bacteria and/or disperse them to a new environment. Most ironically, instead of disturbing the usual chemotactic growth of the set of bacteria, this unknown event may place a newer set of bacteria nearer to the food location. From a broad perspective, elimination, and dispersal are parts of the population-level long-distance motile behavior. In its application to optimization, it helps in reducing the behavior of *stagnation* (i.e. being trapped in a premature solution point or local optima) often seen in such parallel search algorithms. This section is based on the work in [16]. The detailed mathematical derivations as well as theoretical aspect of this new concept are presented in [15,16].

3.6 Rule based bacteria foraging

The basic bacteria foraging strategy lacks in adaptation, as there is a constant run length unit (C) used in the chemotaxis step. Keeping the value of (C) fixed makes the convergence of the algorithm slow, particularly towards the end of the convergence process. To eradicate this bottleneck, the value of (C) is made to vary as per a simple heuristic rule. This speeds up the convergence of the algorithm both in the cases where swarming effect is included or excluded. The philosophy of the rule is to increase (C) when the bacteria is moving in better direction by a very small fixed step, and to decrease its value by the same amount otherwise. Details are discussed in the algorithm presented below.

3.7 Bacteria foraging algorithm

The algorithm of the proposed scheme is as follows:
Step1-initialization

i. Number of parameters (p)to be optimized.
ii. Number of bacteria (S) in the population.
iii. Swimming length N_s after which tumbling of bacteria will be undertaken in a chemotactic loop.
iv. N_c the number of chemotactic steps. ($N_c > N_s$).
v. N_{re} the number of reproduction steps.
vi. N_{ed} the number of elimination and dispersal events.
vii. P_{ed} the probability of the elimination and dispersal bacteria.
viii. The location of each bacterium P (i, j ,k) which is specified $P(i,j,k)=\{\theta^i(j,k,l)$ for i= 1,2,..., S }
ix. The value of C(i) which is assumed to be constant in our case for all the bacteria to simplify the design strategy.

In this study, the following initial values are selected randomly: p=2,S=6, $N_s = 4$, $N_c = 4$, $N_{re} = 100$, N_{ed} =2, P_{ed} =0.25.

Step-2 Iterative algorithm for optimization

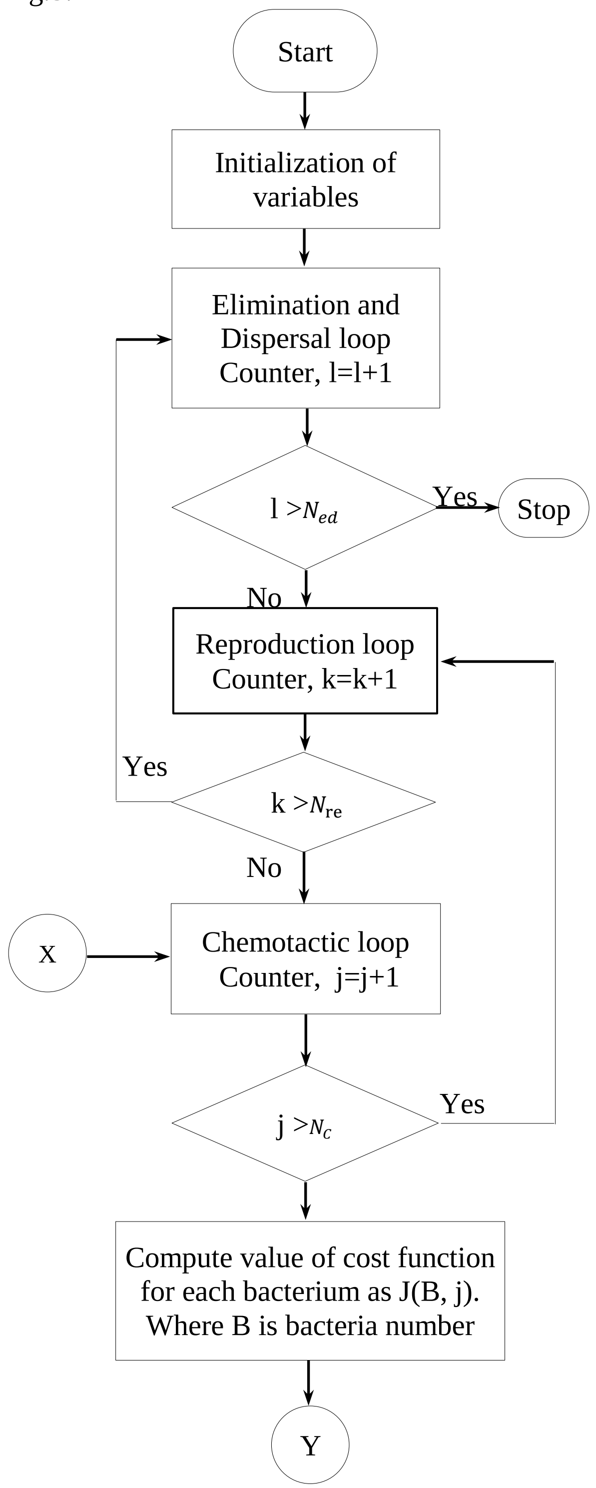
This section represents the bacterial population chemotaxis, swarming, reproduction, elimination and dispersal (initially, $j=k=l=0$). For the algorithm updating θ^i automatically results in updating of 'P'.

1) Elimination-dispersal loop: $l=l+1$
2) Reproduction loop: $k=k+1$
3) Chemotaxis loop: $j=j+1$
   a) For $i=1, 2,\dots, S$, calculate cost function value for each bacterium i as follows.
      - Compute value of cost function $J(i, j, k, l)$. Let
        $$J_{sw}(i,j,k,l) = J(i,j,k,l) + J_{cc}(\theta^i(j,k,l),P(j,k,l))$$
        (J_{cc} (θ) is used to model the cell-to-cell signaling).
      - Let $J_{last} = J_{sw}(i,j,k,l)$ to save this value since we may find a better cost via a run.
      - End of For loop.
   b) For $i=1,2,\dots.S$ take the tumbling /swimming decision
      - Tumble: Generate a random vector $\Delta(i) \in \mathcal{R}^p$ with each element $\Delta_m(i)$ $m=1,2,..p$.
      - Move: let
        $\theta^i(j+1,k,l)=\theta^i(j,k,l)+C(i)\frac{\Delta(i)}{\sqrt{\Delta^T(i)\Delta(i)}}$
      - Compute $J(i, j+1, k, l)$
      - Swim:
        i) Let m=0; (counter for swim length).
        ii) While m<N_s (have not climbed down too long)
           - Let m=m+1
           - Compute $J_{sw}(i,j+1,k,l)$
           - If $J_{sw}(i,j+1,k,l) < J_{last}$, let $J_{last} = J_{sw}(i,j+1,k,l)$ and then $\theta^i(j+2,k,l)=\theta^i(j+1,k,l)+C(i)\frac{\Delta(i)}{\sqrt{\Delta^T(i)\Delta(i)}}$ and use the equation $(\theta^i(j+1,k,l))$ to compute the new $J(i,j+1,k,l)$
           - Else, let m=N_s. this the end of the while statement.
   c) Go to the next bacterium (i+1) if $i \neq$S (i.e. go to b) to process the next bacterium .
4) If j<N_c, go to (3). In this case, continue chemotaxis since the live of the bacteria is not over.
5) Reproduction
   a) For the given k and l, and for $i=1,2,..S$, let J^i_{health}=min $\{J_{sw}(i,j,k,l)\}$ be the health of the bacterium i (a measure of how many nutrients it got over its life time and how successful it was at avoiding noxious substance). Sort bacteria in order of ascending cost J_{health} (higher cost means lower health).
   b) the S_r=S/2 bacteria with highest J_{health}value die and other $S_r$bacteria with the best value split (and the copies that are made are placed at the same location as their parent).
6) If k < N_{re} go to 2, in this case we have not reached the number of specified reproduction

steps, so we start the next generation in the chemotactic loop.

7) Elimination-dispersal: For *i=1,2,..S*, with probability $P_{\rm ed}$, eliminate and disperse each bacteria (this keeps the number of bacteria in the population constant) to a random location on the optimization domain.

The flow chart of the above algorithm is shown in Fig.3.



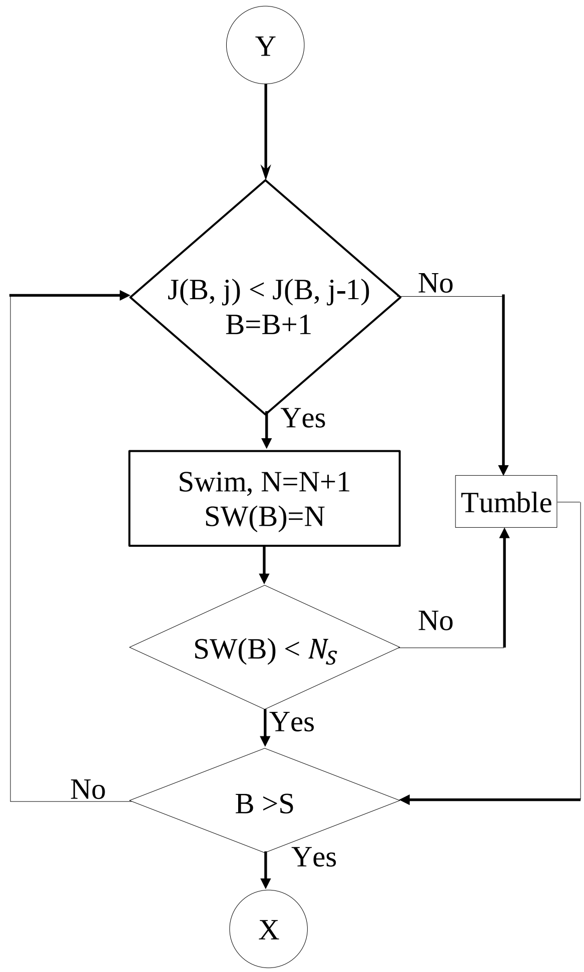


Fig. 3 Flow chart of Bacterial Foraging algorithm

4 Particle Swarm Optimization

4.1 Overview

Particle Swarm Optimization (PSO) algorithm was originally introduced by Kennedy and Eberhart in 1995 [17]. Many researches have been introduced to improve the performance of the PSO with respect to the speed of convergence and to make sure that the PSO will not get stuck in a local minima [18-20]. The PSO has been developed through a simulation of simplified social models. The feature of the method are as follows:

1) The method is based on researches on the swarms such as fish schooling and flocking.
2) It is based on a simple concept. Therefore, the computation time is short and it requires few memories.

According to the research results for bird flocking, birds are finding food by flocking (not by each individual). It leaded the assumption that information is owned jointly in flocking. According to observation of behavior of human groups, behavior pattern on each individual is based on several behavior patterns authorized by the groups such as customs and the experiences by each individual (agent).

PSO is basically developed through simulation of bird flocking in two-dimension space. The position of each individual (agent) is represented by XY axis position and the velocity is expressed by v_x (the velocity of X axis) and v_y (the velocity of Y axis). Modification of the agent position is realized by the position and velocity information.

An optimization technique depends on the above concept. Bird flocking optimizes a certain objective function. Each agent knows its best value so far (pbest) and its XY position. Moreover, each agent knows the best value so far in the group (gbest) among pbests. Each agent tries to modify its position using the following information:

1 the current position (x,y),
2 the current velocities (v_x, v_y),
3 the distance between the current position, and pbest and gbest.

This modification can be represented by the concept of velocity. Velocity of each agent can be modified by the following equation:

$$v_i^{k+1} = wv_i^k + c_1\text{rand}_1\left(\text{pbest}_i - s_i^k\right) + c_2\text{rand}_2(\text{gbest-}s_i^k) \quad (4)$$

Using the above equation, a certain velocity, which gradually gets close to pbest and gbest can be calculated. The current position (searching point in the solution space) can be modified by the following equation:

$$s_i^{k+1} = s_i^k + v_i^{k+1} \quad (5)$$

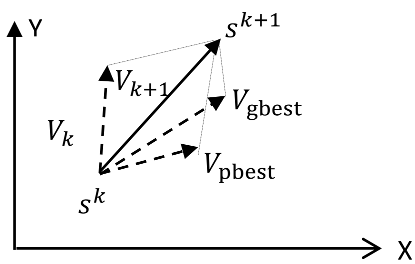


Fig. 4 Concept of modification of a searching point

4.2 PSO algorithm

Step. 1: Initialize an array of particles with random positions and their associated velocities to satisfy the inequality constraints.

Step. 2: Check for the satisfaction of the quality constraints and modify the solution if required.

Step. 3 Evaluate the fitness function of each particle.

Step. 4 Compare the current value of the fitness function with the particles previous best value (*pbest*). If the current fitness value is less, then assign the current coordinates (positions) to *pbestx.*

Step. 5 Determine the current global minimum fitness value among the current positions.

Step. 6 Compare the current global minimum with the previous global minimum (gbest). If the current global minimum is better than gbest, then assign the current global minimum to *gbest* and assign the current coordinates (positions) to *gbestx.*

Step. 7 Change the velocities according to equation (4).

Step. 8 Move each particle to the new position according to equation (5) and return to step 2.

Step. 9 Repeat Step 2-8 until optimization is satisfied or the maximum number of iterations is reached [21].

5 Hybrid Particle Swarm-Bacteria Foraging Optimization (PS-BFO)

5.1 Overview

In BFA, chemotaxis provides a basis for local search, and the reproduction process speeds up the convergence. Elimination and dispersal help to avoid premature convergence and progresses search towards global optima by eliminating and dispersing the bacteria. But chemotaxis and reproduction are not enough for global optima searching as bacteria may get stuck around the initial positions or local optima because dispersion event happens after a certain number of reproduction processes. In BFA, it is possible either to change gradually or suddenly to eliminate the accidents of being trapped in the local optima by introducing mutation operator. The mutation operator brings about diversity in the population to avoid premature convergence getting trapped in some local optima.

In BFA, step size of each generation is the main determining factor for accuracy as well as convergence of global best optima. Bacterial Foraging Optimization with fixed step size suffers from two main problems.

1. If step size is very small then it requires many generations to reach optimum solution. It may not achieve global optima with less number of iterations.
2. If the step size is very high then the bacterium reach to optimum value quickly but accuracy of optimum value gets low.

In this paper, a new strategy of updating positions of bacteria is proposed to improve convergence, accuracy and precision in optimal solution. In the first step, all bacteria positions updated after all fitness evaluations in same generation. In second step, diversity of changing the bacteria positions gradually or suddenly and fine-tuning is achieved by mutation using the parameters of PSO. The PSO parameters do not require any additional parameter such as inertia weight and acceleration coefficients for fine-tuning to reach the global optima. The algorithm of the hybrid PSO-BFA is presented below. In proposed technique, a local search is accomplished by chemotaxis events and while global search in entire search space is attained by reproduction and mutation. The chemotaxis and reproduction are explained before in section 3.2 and 3.4 then, BFA will be explained below.

5.2 Mutation

After completing chemotaxis step, the bacteria positions are updated by mutation operator in order to move towards the global optimum. The mutation plays an important role in the fine tuning of hybrid PSO-BFA as well as achieving high precision in optima value. At the initial stage, ratio of global θ and $\theta(i, j, k)$ is very small resulting in large step size and during later stages, the step size is decreased because $\theta(i, j, k)$ becomes almost equal to global θ . As the number of generations is increased, bacteria will get attracted towards global optimum. Now $\theta(i, j + 1, k)$ can be updated as expressed below.

$$\theta(i, j+1, k)=\left(1-\frac{\theta_{\text{global}}}{\theta(i,j,k)}\right)*\text{r}_1*\theta_{\text{global}}+\left(\frac{\theta_{\text{global}}}{\theta(i,j,k)}\right)*\text{r}_2*\theta_{\text{pbest}}(\text{j, k})$$

Where, $\theta(i, j, k)$ = Position vector of i-th bacterium in j-th chemotaxis step and k-th reproduction steps.
θ_{pbest} (j, k) = Best position in j-th chemotaxis and k-th reproduction steps.
θ_{global} = Best position in the entire search space.
r_1 and r_2 = Random values.

5.3 Hybrid PSO-BFA algorithm

1. Initialize Parameters S, N_C, N_S, N_{re}.
   Where, $J(i, j, k)$ = Fitness value or cost of i-th bacteria in the j-th chemotaxis and k-th reproduction steps.
   J_{best} (j, k) = Fitness value or cost of best position in the j- th chemotaxis and k-th reproduction steps J_{global} = Fitness value or cost of the global best position in the entire search space.
2. Update the following parameters
   J_{best} (j, k)
   J_{global}= J_{best} (j, k)
3. Reproduction loop: $k = k+1$
4. Chemotaxis loop: $j = j+1$
   a) compute fitness function $J(i, j, k)$ for i =1, 2, 3…S Update J_{best} (j, k) and θ_{pbest} (j, k)
   b) Tumble: Generate a random vector $\Delta(\text{i})\in R^p$ with each element Δ_m(i),m=1,2,…p, a random number
   c) Compute θ for i = 1, 2…S
   $$\theta(i, j+1, k)= \theta(i, j, k)+C(i)\frac{\Delta(i)}{\sqrt{\Delta^T(i)\Delta(i)}}$$
   d) Swim
      i) Let m =0 (counter for swim length)
      ii) While m < N_S
         Let m = m+1
         Compute fitness function $J(i, j+1, k)$ for $i = 1, 2, 3...S$
         Update J_{best} $(j+1, k)$
         - If J_{best} (j+1,k) < J_{best} (j, k) (if doing better) then J_{best} (j, k) = J_{best} (j+1, k) *and* θ_{pbest} (j, k) =θ_{pbest} (j+1, k)
           Use the equation of θ (i, j +1, k) to compute the new J(i, j+1, k).
         - Else, let m = Ns. This is the end of the while statement.
   e) Mutation: Change the position of bacteria by mutation. Compute θ for i = 1, 2…S

$$\theta(i, j+1, k)=\left(1-\frac{\theta_{\text{global}}}{\theta(i,j,k)}\right)*\text{r}_1*\theta_{\text{global}}+\left(\frac{\theta_{\text{global}}}{\theta(i,j,k)}\right)*\text{r}_2*\theta_{\text{pbest}}(\text{j, k})$$

5. If j < N_C, go to step 4. In this case, continue chemotaxis, since the life of bacteria is not over.
6. The S_r=S/2 bacteria with the highest cost function (J) values die and other $S_r$bacteria with the best values split. Update J_{global} and θ_{global} .
7. If k < N_{re}, go to step 3. Otherwise end.

6 PSO-BFA Based PID Tuning for PSS

In the proposed design approach, the PID control structure shown in Fig. 5 is used as the power system stabilizer as opposed to the traditional lead-lag controller. The PID parameters K_P, K_i and K_d are tuned using the PSO-BFA technique discussed in sections 3,4.

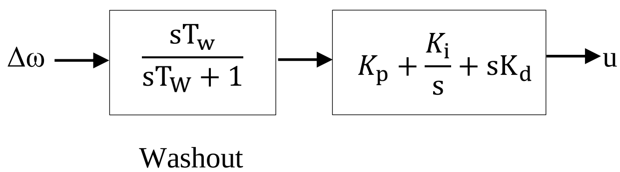


Fig. 5 The PID power system stabilizer

Where the speed deviation is the input to the controller, u is the supplementary stabilizing signal, and the washout filter is used to remove the controller effect at steady state conditions.

The design starts with incorporating the PID structure with the washout in the system matrix discussed in equation (2) to form the augmented *A* matrix given below:

$$A=\begin{bmatrix} 0 & \omega_0 & 0 & 0 & 0 \\ \frac{-K_1}{M} & 0 & \frac{-K_2}{M} & 0 & 0 \\ \frac{-K_4}{T_{d\grave{o}}} & 0 & \frac{-1}{K_3 T_{d\grave{o}}} & \frac{1}{T_{d\grave{o}}} & 0 \\ \frac{MK_e(-K_5+M^{K_i}/\omega_0-K_dK_1)}{MT_e} & \frac{K_eK_p}{T_e} & \frac{-MK_6K_e-K_2K_eK_d}{MT_e} & \frac{-1}{T_e} & \frac{K_e}{T_e} \\ \frac{M^{K_i}/\omega_0-K_dK_1}{MT_w} & \frac{K_p}{T_w} & \frac{-K_2K_d}{MT_w} & 0 & \frac{-1}{T_w} \end{bmatrix} \quad (6)$$

7 Objective Function

In this paper a Hybrid PSO-BFA scheme has been used for the optimization of PID-PSS parameters. Just like any other optimization problem, a cost or an objective function needs to be formulated for the optimal PID-PSS design. The objective in the optimal PID-PSS design is to maximize damping; in other words minimize the overshoots and settling time in system oscillations.

The Integral of Squared Error (ISE) is considered as the cost function to be minimized. ISE accounts mainly for error at the beginning of the response and to a lesser degree for the steady state duration[22]. The objective function is given by (7).

$$J=\int_0^{\infty}(\Delta\omega)^2 dt \quad (7)$$

Where $\Delta\omega$ is the speed deviation of the generator obtained from time domain simulation. Therefore, the design problem can be formulated as the following optimization problem.

Minimize J

Subject to

$$z^{min} \le z \le z^{max} \quad (8)$$

Where z is a vector, which consists of the parameters of the PID-PSS.

The proposed approach employs PSO-BFA to solve this optimization problem and search for the optimal set of PID-PSS parameters.

8 Implementation of the PSO-BFA Based PID PSS

The proposed approach is implemented on the power system shown in Fig. 1. The system data is given in the Appendix. The simulation results of the system for the deviations in the speed and the angle for a step disturbance in the mechanical input is shown in Fig. 6 and Fig. 7 with and without classical PID controller. The classical PID parameters are computed using Ziegler-Nichols method and have the following values: K_P=40.4632, K_i=1, and K_d=28.7382. The two figures show that the uncontrolled system is unstable but there is damping of oscillation occurs for the system under the classical PID controllers.

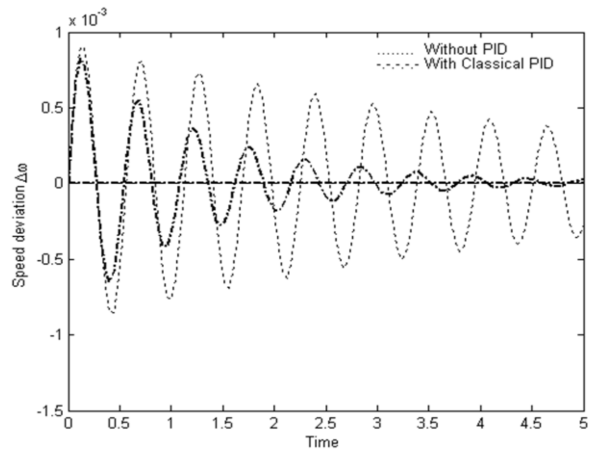


Fig. 6. Speed deviation for disturbance $\Delta P_m = 0.05$ (with and without classical PID controller)

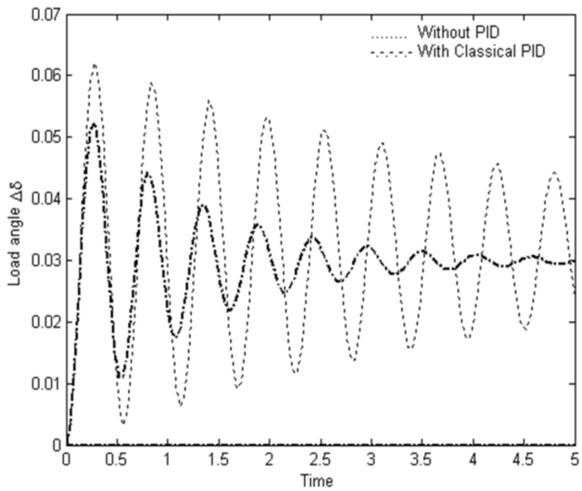


Fig. 7. Load deviation for disturbance $\Delta P_m = 0.05$ (with and without classical PID controller)

To stabilize the system, the PID controller based PSS used and its parameters are tuned using the BFA algorithm. The results of the controlled system under disturbance in the mechanical input $\Delta P_m = 0.05$ is shown in Fig. 8 and Fig. 9 The two figures show that more damping in the oscillation of the system and the system performance is enhanced. The optimized PID parameters for BFA are $K_P = 36.1976$, $K_i = 1.2732$ and $K_d = 5.2715$. It is evident from the figures that the system with the PID based BFA is stable and more enhancement than the classical PID.

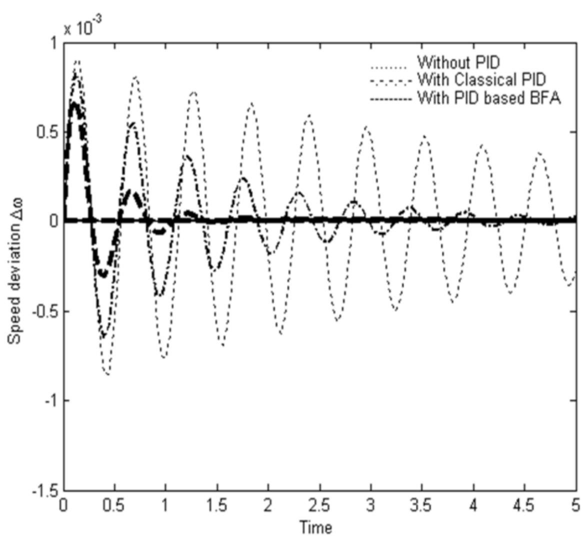


Fig. 8. Speed deviation for disturbance $\Delta P_m = 0.05$ (with and without classical PID controller, and with BFA only)

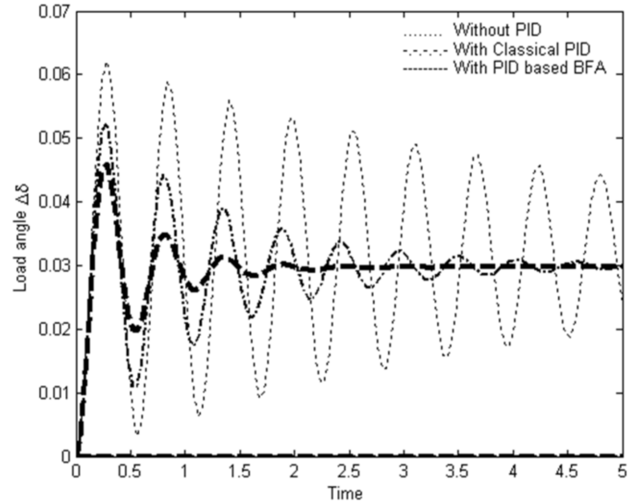


Fig. 9. Load deviation for disturbance $\Delta P_m = 0.05$ (with and without classical PID controller, and with BFA only)

To increase the stability response of the system, hybrid of PSO-BFA is used for tuning the parameters of the PID. The simulation results of the system for the deviations in the speed and the angle under disturbance in the mechanical input $\Delta P_m = 0.05$ is shown in Fig. 10 and Fig.11. The PSO-BFA parameters used for tuning K_P, K_i and K_d are given in Table 1.

Table 1. the parameters of hybrid particle swarm –bacteria foraging optimization

NO. of bacteria in the population (S=6)	BFA parameters
Swimming length (Ns=4)	
NO. of chemotactic steps(N_C=4)	
NO. of reproduction steps(N_{re}=100)	
NO. of elimination-dispersal events (N_{ed}=2)	
The probability of the eliminated and dispersal bacteria(P_{ed}=0.25)	
NO. of bacteria of the best value cost function (S_r=S/2)	
NO. of particles=20	PSO parameters
NO. of swarms (Kp, Ki, Kd)=3	

The optimized PID parameters are $K_P = 190.4623$, $K_i = 2.4753$ and $K_d = 11.2516$. The two figures show that the stability of the system is improved, the system reach to the steady state faster and the oscillation is rapidly damped.

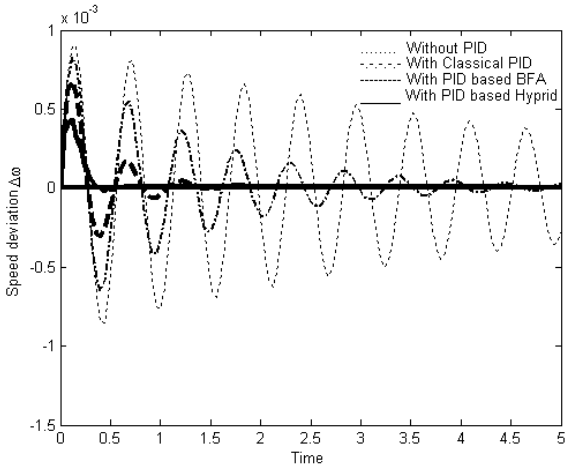


Fig. 10 Speed deviation for disturbance $\Delta P_m = 0.05$ (with and without classical PID controller, with BFA only and with hybrid PSO-BFA)

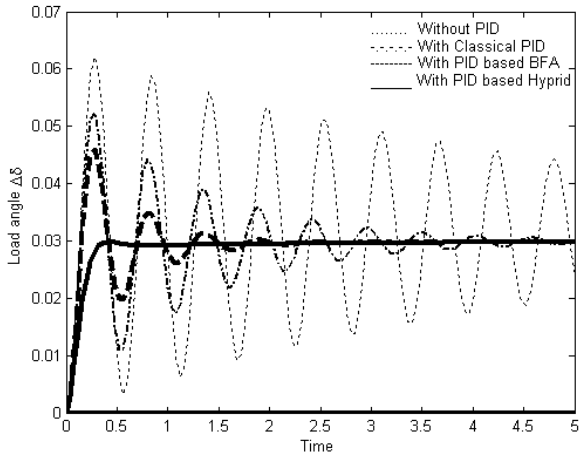


Fig. 11 Load deviation for disturbance $\Delta P_m = 0.05$ (with and without classical PID controller, with BFA only and with hybrid PSO-BFA)

9 Conclusion

In this paper, the design of a PID power system stabilizer using hybrid PSO-BFA had been investigated. The design was applied to a typical single machine infinite bus power system. The simulation results of the system for the deviations in the speed and the angle demonstrated that the designed optimal PID-PSS based hybrid PSO-BFA optimization method is capable of guaranteeing the stability and performance of the power system better than the classical PID controllers and the PID-PSS based BFA only.

References:

[1] P. K. Modi, S. P. Singh, J. D. Sharma and P. K. Pradhan, Stability Improvement of Power System by Decentralized Energy, *Advances in Energy Research*, 2006, pp. 65-70.

[2] A. A. El-Dib, H. K. M. Youssef, M. M. El-Metwally, Z. Osman, Maximum Loadability of Power System Using Hybrid Particle Swarm Optimization, *Electric Power System Research*, 76, 2006, pp. 485-492,.

[3] M .A Pai, *Power System Stability*, New York: North-Holland, 1981.

[4] H. Saadat, *Power System Analysis*, McGraw-Hill International Editions, 1999.

[5] C. Bayliss, B. Hardy, *Transmission and Distribution Electrical Engineering*, 3rd edition, Newnes, 2007, pp. 28.

[6] P. W. Sauer, M. A. Pai, *Power System Dynamic and Stability*, Prentice Hall, 1997.

[7] B. Sumanbabu, S. Mishra, B. K. Panigrahi, G. K. Venayagamoorthy, Robust Tuning of Modern Power System Stabilizers Using Bacterial Foraging Algorithm, *IEEE Congress on Evolutionary computation*, 2007, pp. 2317-2324.

[8] Y.L. Abdel-Magid, M. Bettayeb, M.M. Dawoud, Simultaneous stabilization of power systems using genetic algorithms, *IEE Proceedings Generation Transmission Distribution*, Vol.144, No.1, 1997, pp. 39-44.

[9] M.A. Abido, A novel approach to conventional power stabilizer design using tabu search, *Int. J. Electric Power Energy Sys.*, Vol.21, 1999, pp. 443- 454.

[10] M.A. Abido, Robust design of multimachine power system stabilizers using simulated annealing, *IEEE Trans. Energy conversion*, Vol.15, 2000, pp. 297-304.

[11] 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, 2008, pp.153-159.

[12] K.M. Passino, Biomimicry of bacterial foraging for distributed optimization and control, *IEEE Control Syst.*, 2002,pp. 52–67.

[13] Ziegler, J.G, Nichols, N.B., Optimum settings for automatic controllers, *Trans. ASME*, Vol.64, 1942, pp. 759-768.

[14]Kundur, P., *Power system stability and control*, McGraw-Hill, New York, 1993, pp.700-822.

[15] K.M. Passino, Biomimicry of bacterial foraging for distributed optimization and control, *IEEE Control Syst.*, 2002, pp. 52–67.
[16] S. Mishra, A hybrid least square-fuzzy bacteria foraging strategy for harmonic estimation, *IEEE Trans Evolutionary Comput.,* Vol.9, 2005, pp.61-73.
[17] J. Kennedy, R. Eberhart, Particle Swarm Optimization, *Proc. IEEE Int. Conf. Neural Network*, 1995, pp. 1942-1948.
[18] Y. Fukuyama, et al., Practical distribution state estimation using hybrid particle swarm optimization, *Proceedings of IEEE Power Engineering Society Winter Meeting, Columbus*, 2001, pp. 815-820.
[19] G. Ciuprina, D. Ioan, I. Munteanu, Use of intelligent-particle swarm optimization in electromagnetics, *IEEE Trans.*, 2002, pp.1037-1040.
[20] J. Kennedy, The particle swarm optimization: social adaptation of knowledge, *International Conference of Evolutionary Computation, Indianapolis*, April 1997, pp.303-308.
[21] A. T. AL-Awami, Y. L. Abdel-Magid, M. A. Abido, A particle-swarm-based approach of power system stability enhancement with unified power flow controller, *Electrical power and energy system* 29, 2007, pp.251-259.
[22] H.X. Li, H.B. Gatland, Enhanced Methods of Fuzzy Logic Control, *IEEE, transactions on systems*, 1995, pp.331-336.

List of symbols:

E_t	sending voltage
E_b	receiving voltage
r_e	transmission line resistance
X_e	transmission line reactance
ω_0	synchronous speed
$K_1 - K_6$	contents of synchronous generator
T_{d0}	d- axis open circuit field time constant
M,H	inertia coefficient, constant (M=2H)
D	Damping coefficient
i_d, i_q	armature current direct and quadrature axis components
v_d, v_q	armature voltage direct and quadrature axis components
X'_d, X_d, X_q	direct axis transient, direct axis and quadrature axis reactance
P_m	mechanical power input to machine
P	electric power output from machine
Q	reactive power output from machine
δ	torque angle
Ω	angular velocity
E_{fd}	field voltage
E'_q	q- axis generator internal voltage
$\Delta E'_q$	the change of q- axis internal voltage
E	infinite bus voltage
V_{ref}	reference input voltage
V_t	terminal voltage
A,B	system and control matrices
x, u	state and control vectors
K,T	stabilizer gain and time constant
J	objective function
K_e,T_e	exciter gain and time constant
s^k	current searching point
s^{k+1}	modified searching point
V^k	current velocity
V^{k+1}	modified velocity
V_{pbest}	velocity based on pbest
V_{gbest}	velocity based on gbest

Appendix

- Generator parameters in (pu)

$x_d = 1.6; x'_d = 0.32;\ x_q = 1.55; V_{t0} = 1.05;$

$\omega_0 = 100\pi$ rad/s; T'_{d0} =6 s; $D = 0;\ M = 10$

- *Exciter parameters for the generator*

$K_e = 50;\quad T_e = 0.05$ s

- *Transmission line parameters in (pu)*

$r_e = 0; \quad x_e = 0.4$

Washout filter parameter:-

T_w =5 s

The equations of K1-K6 parameters used in the equation (1), (2), (3) and (4) are obtained by:

$$K_1 = \frac{x_q - x'_d}{x_e + x'_d} i_{q0} E_0 \sin\delta_0 + \frac{E_{q0} E_0 \cos\delta_0}{x_e + x_q}$$

$$K_2 = \frac{E_0 \sin\delta_0}{x_e + x'_d}$$

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

$$K_4 = \frac{x_d - x'_d}{x_e + x'_d} E_0 \sin\delta_0$$

$$K_5 = \frac{x_q}{x_e + x_q} \frac{v_{d0}}{v_{t0}} E_0 \cos\delta_0 - \frac{x'_d}{x_e + x'_d} \frac{v_{q0}}{v_{t0}} E_0 \sin\delta_0$$

$$K_6 = \frac{x_e}{x_e + x'_d} \frac{v_{d0}}{v_{t0}}$$

The parameters i_{q0}, v_{d0}, v_{q0}, i_{d0} , E_{q0}, E_0, δ_0 can be computed from:

$$i_{q0} = \frac{P_0 v_{t0}}{\sqrt{(P_0 x_q)^2 + (v_{t0}{}^2 + Q_0 x_q)^2}}$$

$$v_{d0} = i_{q0} x_q$$

$$v_{q0} = \sqrt{v_{t0}{}^2 - v_{d0}{}^2}$$

$$i_{d0} = \frac{Q_0 + x_q i_{q0}{}^2}{v_{q0}}$$

$$E_{q0} = v_{q0} + i_{d0} x_q$$

$$E_0 = \sqrt{(v_{d0} + i_{q0} x_e)^2 + (v_{q0} - i_{d0} x_e)^2}$$

$$\delta_0 = \tan^{-1}\left(\frac{v_{d0} + i_{q0} x_e}{v_{q0} - i_{d0} x_e}\right)$$