

PID Controllers Design for a Power Plant Using Bacteria Foraging Algorithm

Ahmed Bensenouci

Electrical and Computer Engineering Department,
King Abdulaziz University
Jeddah, Kingdom of Saudi Arabia
Bensenouci@ieee.org

Abstract— This paper provides the design for two PID controllers, one for the terminal voltage and the other for the electric power. Both controllers are designed to control a sample power system that comprises a synchronous generator connected to a large network via a step-up transformer and a transmission line. The generator is equipped with two decoupled control-loops, namely, the speed/power (governor) and voltage (exciter) controllers. The gains setting of both PIDs are found using Bacteria Foraging Algorithm (BFA). PID is considered because of its robustness, simple structure and easy implementation. It is also preferred in plants of higher order that cannot be reduced to lower ones. To show the effectiveness of the design, diverse tests, namely, step/tracking in the control variables, and variation in plant parameters, are applied. From the simulation, the results are very encouraging to pursue further this trend.

Keywords-power plant; bacteria foraging algorithm; PID controller design;

I. INTRODUCTION

Adequate control is desirable to improve the synchronous generator performance and overcome limitations in stability boundaries caused by the use of larger generator size with lower specific inertia, and also with longer transmission lines [1,2].

Successful studies using optimal control policy have been described by a number of literatures. One of the faced problems is the difficulty of measuring all the system state variables. Based on this, output feedback is usually preferred over state feedback [3].

PID controller represents the simplest output feedback controller. It is widely used in the industry owing to its simplicity and robustness [3,4] and found to be adequate for most plants to obtain desired performances. Several design techniques for PIDs are found in the literature, starting from Ziegler Nichols method to modern ones (ANN, Fuzzy, evolutionary programming, sliding mode, robust design via some norm, Linear Matrix Inequality technique, etc.) [5-7].

A new evolutionary computation technique, called Bacterial Foraging Algorithm (BFA) [8-12] has been proposed in the literature and explored in this work. The foraging (methods for locating, handling, and ingesting food) behavior of *E. coli* bacteria, present in our intestines, is mimicked. They undergo

different stages such as chemotaxis, swarming, reproduction, and elimination and dispersal. In the chemotaxis stage, each bacterium can tumble or run (swim). In swarming, each bacterium signals other bacteria via attractants to swarm (group) together. In reproduction, the least healthy bacteria die and the healthiest each splits into two bacteria that are then placed in the same location. Further, any bacterium is eliminated from the total set of bacteria just by dispersing it to a random location (elimination and dispersal). Several runs were done and several parameters were tried.

This paper presents the design steps of two decentralized PID controllers, one for the voltage and the other for the speed/power control loops. A local PID controller is designed for each control loop. The PID design is carried out using a bacterial foraging optimization technique scheme. They are designed to drive the synchronous generator connected to an infinite-bus through a step-up transformer and a transmission line. To test the effectiveness of the controlled power plant, diverse tests, namely, step/tracking in the control variables, and variation in the plant parameters are carried out. The results are very encouraging. Simulation is done using Matlab platform. An objective function is formulated for the optimization problem to fulfill the requirements of the design.

II. SYSTEM MODELING

Fig. 1 shows the block diagram of the controlled sample power system that comprises a steam turbine driving a synchronous generator which is connected to an infinite bus via a step-up transformer and a transmission line. The real power P_t and terminal voltage V_t at the generator terminals are measured and fed to the controller. The outputs of the controller (control inputs) are fed into the generator-exciter and governor-valve [13].

The linear Multi-Input Multi-output (MIMO) model of the system is required to design such controllers. It is derived from the system nonlinear model for a specific operating point. The variables shown represent small displacements around the selected operating point. The system can be written as

$$\begin{aligned}\dot{x} &= Ax + Bu \\ y &= Cx\end{aligned}\tag{1}$$

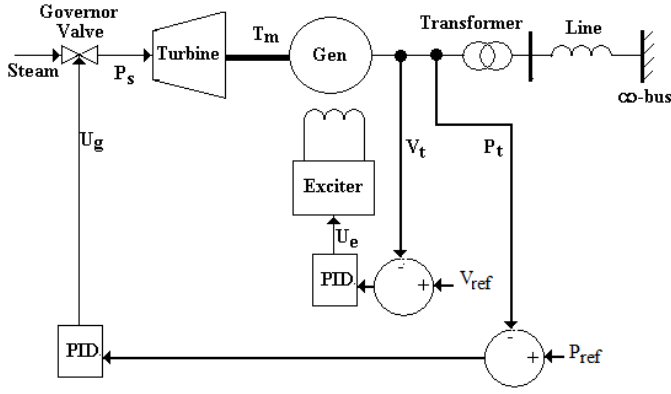


Fig. 1 Power system block diagram

The matrices A, B, and C have the form:

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ K_8 & \frac{-(K_d + T_d)\omega_0}{2H} & K_9 & 0 & 0 & \frac{\omega_0}{2H} \\ K_{10} & 0 & K_3 & \frac{\omega_0 r_{fd}}{x_{ad}} & 0 & 0 \\ 0 & 0 & 0 & \frac{-1}{T_e} & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{-1}{T_e} & 0 \\ 0 & 0 & 0 & 0 & \frac{-1}{T_b} & \frac{-1}{T_b} \end{bmatrix}$$

$$B^t = \begin{bmatrix} 0 & 0 & 0 & \frac{1}{T_e} & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{Kg}{T_g} & 0 \end{bmatrix}$$

$$C = \begin{bmatrix} K_{11} & 0 & K_{12} & 0 & 0 & 0 \\ K_{13} & 0 & K_{14} & 0 & 0 & 0 \end{bmatrix}$$

Where

$$x = [\delta, \dot{\delta}, \psi_{fd}, E_{fd}, P_s, T_m]^t : \text{state vector}$$

$$u = [U_e \ U_g]^t : \text{input vector}$$

$$y = [P_t \ V_t]^t : \text{output measurement vector}$$

And

$$P_t = K_{11}x_1 + K_{12}x_3 : \text{output power}$$

$$V_t = K_{13}x_1 + K_{14}x_3 : \text{terminal voltage}$$

With the inclusion of both PID controllers, the system becomes:

$$\dot{x}_{cl} = A_{cl}x_{cl} + B_{cl}w$$

$$y_{cl} = C_{cl}x_{cl} + D_{cl}w \quad (2)$$

Where

$$A_{cl} = \begin{bmatrix} WA & WB_1 & WB_2 \\ -K_{I1}C_1 & 0 & 0 \\ -K_{I2}C_2 & 0 & 0 \end{bmatrix}$$

$$B_{cl} = \begin{bmatrix} WB_1K_{P1} & WB_2K_{P2} \\ K_{I1} & 0 \\ 0 & K_{I2} \end{bmatrix}$$

$$B_1 = \begin{bmatrix} 0 & 0 & 0 & \frac{1}{T_e} & 0 & 0 \end{bmatrix}^t \quad B_2 = \begin{bmatrix} 0 & 0 & 0 & 0 & \frac{Kg}{T_g} & 0 \end{bmatrix}^t$$

$$C_{cl} = \begin{bmatrix} K_{11} & 0 & K_{12} & 0 & 0 & 0 & 0 \\ K_{13} & 0 & K_{14} & 0 & 0 & 0 & 0 \end{bmatrix} \quad D_{cl} = 0_{2 \times 2}$$

With

$$W = [1 + B_1K_{D1}C_1 + B_2K_{D2}C_2]^{-1}$$

$$x_{cl} = [\delta \ \dot{\delta} \ \psi_{fd} \ E_{fd} \ P_s \ T_m \ x_7 \ x_8]^t$$

$$w = [V_{ref} \ P_{ref}]^t$$

And K_{Pi} , K_{fi} and K_{Di} ($i=1,2$) represent the PID gains.

III. FORMULATION OF OBJECTIVE FUNCTION

In any optimization problem, a cost or an objective function needs to be formulated. The objectives in the optimal PID design are to maximize damping; in other words minimize the overshoots and settling time in system oscillations. To achieve this, the errors in V_t , P_t and ω_r , and the placement of the dominant pole λ_d to a specific location are used in the cost function J as follows:

$$J = (\text{Real}(\lambda_d) - a)^2 + (\text{Imag}(\lambda_d) - b)^2 +$$

$$\int_{t=0}^{t=t_f} t \left[(V_{ref} - V_t(t))^2 + (P_{ref} - P_t)^2 + \omega_r(t)^2 \right] \quad (5)$$

where t is the simulation time in seconds, t_f is the final simulation time, $\omega_r(t)$ is the speed deviation of the generator. The selected dominant eigenvalue components are chosen through the values of "a" and "b".

The Integral of Time Squared Error (ITSE) is considered as a part of the cost function to be minimized by the bio-inspired algorithm. ITSE is a better criterion which keeps account of errors at the beginning but also emphasizes the steady state [4]. Therefore, the design problem can be formulated as the following optimization problem.

$$\begin{aligned} & \text{Minimize } J \\ & \text{Subject to} \\ & K_{i,min} < K_i < K_{i,max} \end{aligned}$$

where K_i ($i=1-6$) consists of the parameters of the PIDs. The proposed approach employs BFA to solve this optimization problem and search for the optimal set of PID parameters.

IV. BACTERIAL FORAGING ALGORITHM (BFA)

Recently, bacterial foraging algorithm (BFA) has emerged as a powerful technique for the solving optimization problems. BFA mimics the foraging strategy of *E. coli* bacteria which try to maximize the energy intake per unit time. From the very early days it has drawn attention of researchers due to its effectiveness in the optimization domain. So as to improve its performance, a large number of modifications have already been undertaken. The bacterial foraging system consists of four principal mechanisms, namely chemotaxis, swarming, reproduction and elimination-dispersal. A brief description of each of these processes along with the pseudo-code of the complete algorithm is described below.

Chemotaxis: This process simulates the movement of an *E.coli* cell through swimming and tumbling via flagella. Biologically an *E.coli* bacterium can move in two different ways. It can swim for a period of time in the same direction or it may tumble, and alternate between these two modes of operation for the entire lifetime. Suppose $\theta^i(j,k,l)$ represents i^{th} bacterium at j^{th} chemotactic, k^{th} reproductive and l^{th} elimination-dispersal step. $C(i)$ is the size of the step taken in the random direction specified by the tumble (run length unit). Then in computational chemotaxis the movement of the bacterium may be represented by

$$\theta^i(j+1,k,l) = \theta^i(j,k,l) + C(i) \frac{\Delta(i)}{\sqrt{\Delta^t(i)\Delta(i)}} \quad (3)$$

Where Δ indicates a vector in the random direction whose elements lie in $[-1, 1]$.

Swarming: An interesting group behavior has been observed where a group of *E.coli* cells arrange themselves in a traveling ring by moving up the nutrient gradient when placed amidst a semisolid matrix with a single nutrient chemoeffector. The cells, when stimulated by a high level of *succinate*, release an attractant *aspartate*, which helps them to aggregate into groups and thus move as concentric patterns of swarms with high bacterial density. The cell-to-cell signaling in *E. coli* swarm may be represented by the following function.

$$J_{cc}(\theta, P(j,k,l)) = \sum_{i=1}^S J_{cc}(\theta, \theta^i(j,k,l))$$

$$J_{cc} = \sum_{i=1}^S \left[-d_{attrac} \tan t^e \left(-w_{attrac} \tan t \sum_{m=1}^p (\theta_m - \theta_m^i)^2 \right) \right] + \sum_{i=1}^S \left[h_{repellant}^e \left(-w_{repellant} \sum_{m=1}^p (\theta_m - \theta_m^i)^2 \right) \right] \quad (4)$$

where $J_{cc}(\theta, P(j,k,l))$ is the objective function value to be added to the actual objective function (to be minimized) to present a time varying objective function, S is the total number of bacteria, p is the number of variables to be optimized, which are present in each bacterium and $\theta = [\theta_1, \theta_2, \dots, \theta_p]^t$ is a point in the p dimensional search domain.

Reproduction: The least healthy bacteria eventually die while each of the healthier bacteria (those yielding lower value of the objective function) asexually split into two bacteria, which are then placed in the same location. This keeps the swarm size constant.

Elimination and Dispersal: Gradual or sudden changes in the local environment where a bacterium population lives may occur due to various reasons e.g. a significant local rise of temperature may kill a group of bacteria that are currently in a region with a high concentration of nutrient gradients. Events can take place in such a fashion that all the bacteria in a region are killed or a group is dispersed into a new location. Some guidelines for BFA Parameter Choices are [13]:

Size of population ‘S’: Increasing S can significantly increase the computational complexity of the algorithm. However, for larger values of S , it is more likely at least some bacteria near an optimum point should be started, and over time, it is then more likely that many bacterium will be in that region, due to either chemotaxis or reproduction.

Length of chemotactic step ‘C(i)’: If $C(i)$ are too large, then if the optimum value lies in a valley with steep edges, the search will tend to jump out of the valley, or it may simply miss possible local minima by swimming through them without stopping. On the other hand, if $C(i)$ are too small, convergence can be slow, but if the search finds a local minimum it will typically not deviate too far from it. $c(i)$ is a sort of a “step size” for the algorithm.

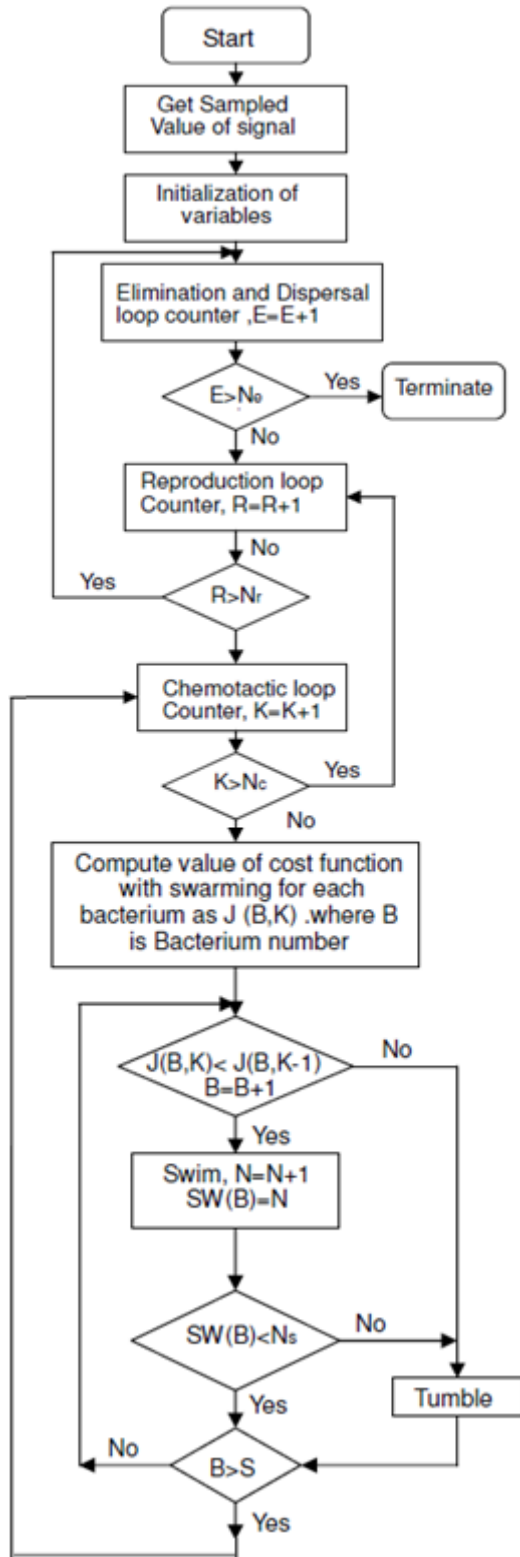


Fig. 2 Flowchart of BFA

Chemotactic step ‘ N_c ’: If the size of N_c is chosen to be too short, the algorithm will generally rely more on luck and reproduction, and in some cases, it could more easily get trapped in a local minimum (premature convergence). N_s creates a bias in the random walk (which would not occur if

$N_s = 0$), with large values tending to bias the walk more in the direction of climbing down the hill.

Reproduction number ‘ N_{re} ’: If N_{re} is too small, the algorithm may converge prematurely; however, larger values of N_{re} clearly increase computational complexity.

Elimination and dispersal number ‘ N_{ed} ’: A low value for N_{ed} dictates that the algorithm will not rely on random elimination-dispersal events to try to find favorable regions. A high value increases computational complexity but allows the bacteria to look in more regions to find good nutrient concentrations. Clearly, if N_{ed} is large, the algorithm can degrade to random exhaustive search. If, however, it is chosen appropriately, it can help the algorithm jump out of local optima and into a global optimum.

Parameters defining cell-to-cell attractant functions ‘ J_{cc} ’: If the attractant width is high and very deep, the cells will have a strong tendency to swarm (they may even avoid going after nutrients and favor swarming). On the other hand, if the attractant width is small and the depth shallow, there will be little tendency to swarm and each cell will search on its own. Social versus independent foraging is then dictated by the balance between the strengths of the cell-to-cell attractant signals and nutrient concentrations.

V. SIMULATION RESULTS

To demonstrate the effectiveness of the PID controllers, several tests are carried out and the results are presented. The simulation results are obtained using MATLAB package.

A. System Parameters:

Initial condition (operating point):

$$X_{c10} = [0.775 \quad 0 \quad 1.434 \quad -0.0016 \quad 0.8 \quad 0.8 \quad 0 \quad 0]$$

BFA parameters selection:

- $p=6$ Dimension of the search space
- $S=50$ Number of bacteria in the population
- $N_c=100$ Number of chemotactic steps per bacteria lifetime
- $N_s=4$ Limits the length of a swim when it is on a gradient
- $N_{re}=4$ Number of reproduction steps
- $S_r=S/2$ Number of bacteria reproductions (splits) per generation (this choice keeps the number of bacteria constant)
- $N_{ed}=2$ Number of elimination-dispersal events
- $ped=0.25$ Probability that each bacteria will be eliminated / dispersed
- $C=0.1$ Basic run length step

$d_{attractant}=0.1$ Magnitude of secretion of attractant by a cell

$w_{attractant}=0.2$ How the chemical cohesion signal diffuses (smaller makes it diffuse more)

$h_{repellant}=0.1$ Sets repellant (tendency to avoid nearby cell)
 $w_{repellant}=10$ Makes small area where cell is relative to diffusion of chemical signal

Obtained PID controller gains from BFA:

$$K_{P1} = 0.30194, K_{I1} = 0.26444, K_{D1} = 0.04041$$

$$K_{P2} = 0.32691, K_{I2} = 0.45424, K_{D2} = 0.05928$$

Closed-loop eigenvalues:

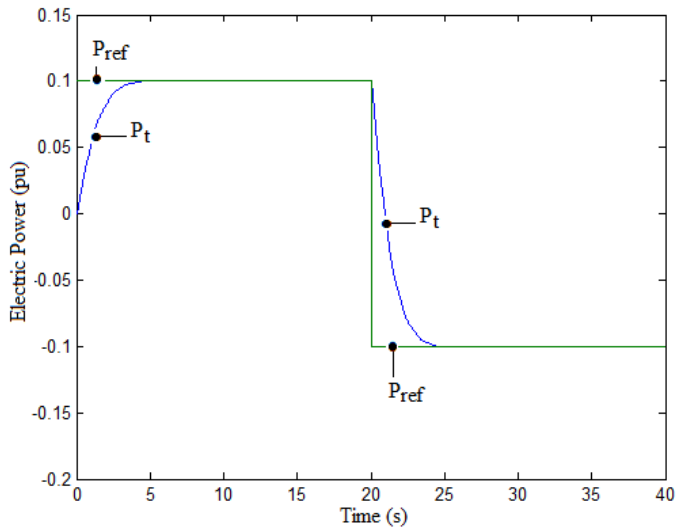
$$\lambda_{cl} = [-10.3 \quad -10.2 \quad -0.887 \pm j0.46 \quad -0.462 \pm j6.23 \quad -0.390 \pm j0.514]^T$$

A. Test 1: Step-response (regulation)

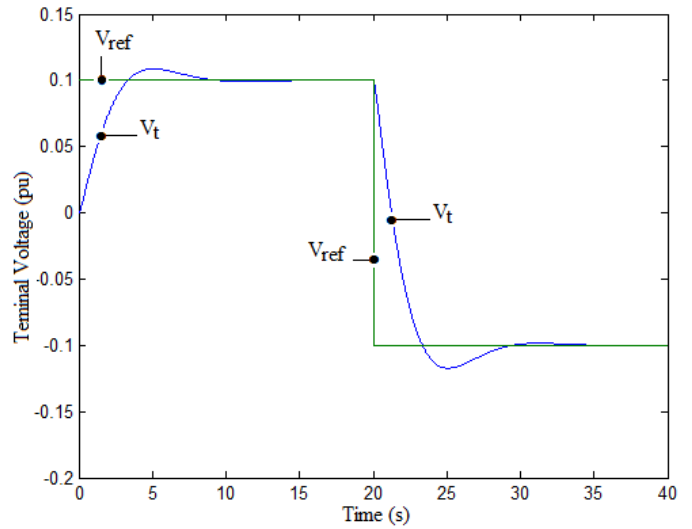
To test the effectiveness of the system equipped with the designed PID controllers, the system is subjected to an increase by 10% in both P_{ref} and V_{ref} then a decrease by the same amount. The time responses of the electric power P_t and the terminal voltage V_t are shown in Fig. 3. For P_t -response, no overshoot is shown while a small overshoot is seen in V_t . In summary, good performance (lower overshoot and acceptable settling time) is obtained.

B. Test 2: Reference tracking response (tracking)

To test the effectiveness of the system to tracking the reference control values P_{ref} and V_{ref} , the system is subjected to an increasing ramp variation of both P_{ref} and V_{ref} from 0% (nominal operating point) to +10% then steady and finally a decreasing ramp from +10% to 0%, as shown in Fig. 4. The controlled values V_t and P_t follow closely their desired values with a very small time-shift. As in Test1, the PIDs show good tracking performance.

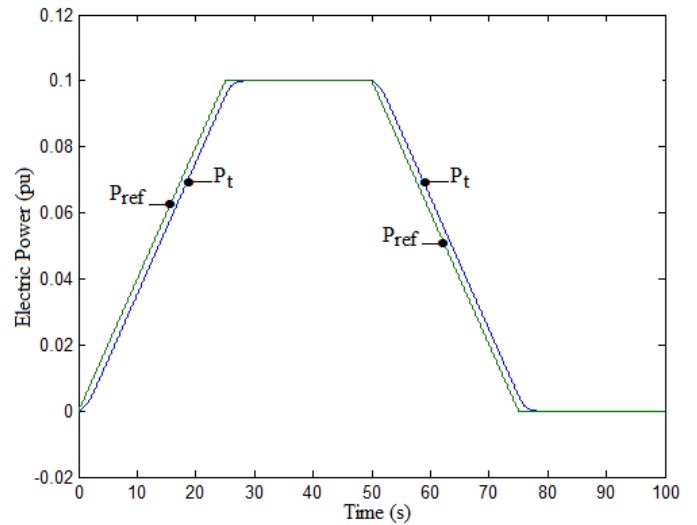


(a) Electric power output

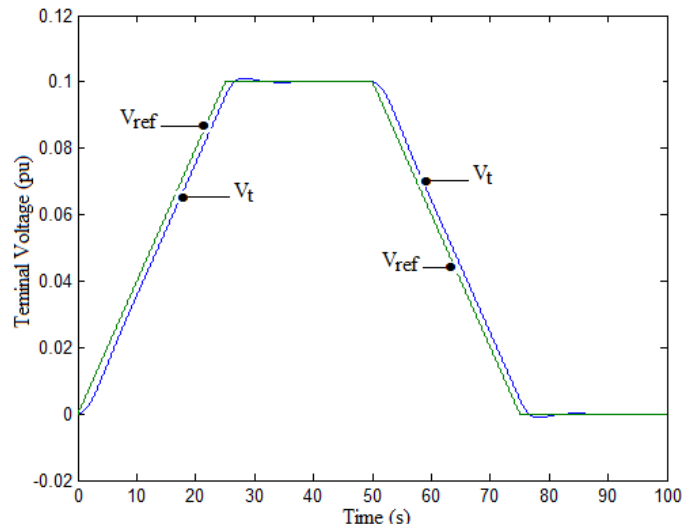


(b) Terminal voltage

Fig. 3 Step-response following $P_{ref}=V_{ref}=10\%$



(a) Electric power output

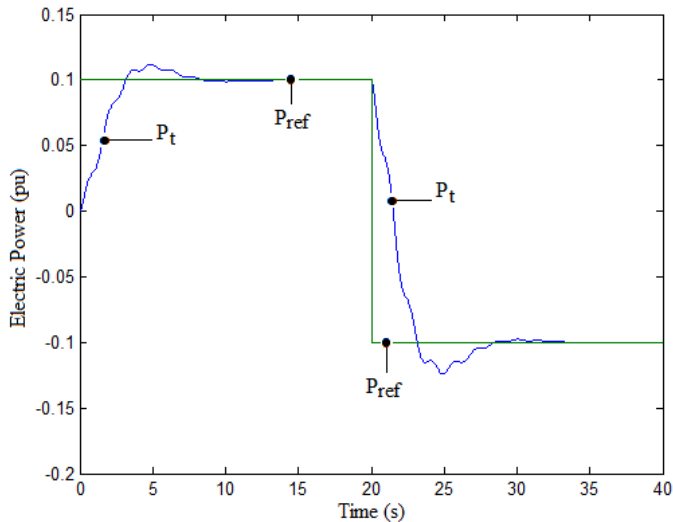


(b) Terminal voltage

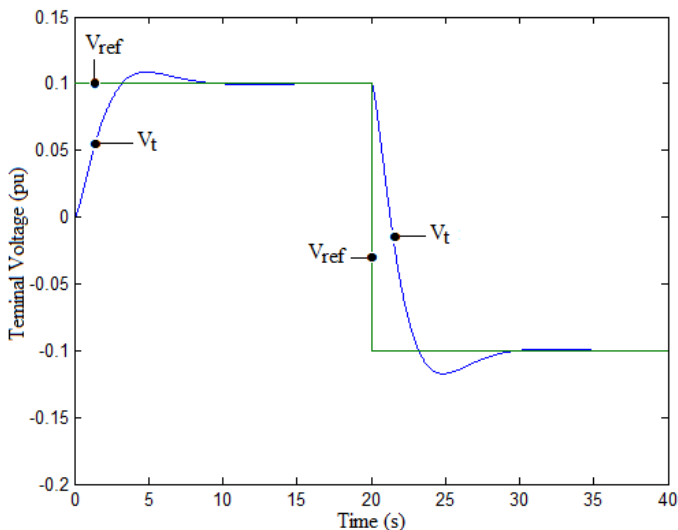
Fig. 4 System responses with H_z /PID due to reference tracking (test 2)

C. Test 3: Parameters variation

To test the robustness to parameters change, a large increase by 100% is applied to the exciter T_e , turbine T_b , and governor T_g time constants, and the inertia constant H . Even though not realistic, but this will demonstrate the controller's robustness to an extremely large variation. Fig. 5 shows the system response following an increase by 10% then a decrease by the same amount in V_{ref} and T_{ref} while experiencing the described parameters change. The parameters change induces some high-frequency oscillations with an acceptable overshoot in P_t response. V_t seems to be unaffected by such a change but a somehow higher delay can be noticed.



(a) Electric power output



(b) Terminal voltage

Fig. 5 System responses with H_{∞} /PID due to parameter change (test 3)

VI. CONCLUSION

Two PID controllers were designed using BFA technique; one for the electric power and the other for the terminal voltage. The PIDs were designed for a sample power system comprising a steam turbine driving a synchronous generator connected to an infinite bus via a step-up transformer and transmission. BFA technique that was explored as a candidate to optimally tune the PID gains represents an optimization technique mimic to the search of food for the bacteria. This is a promising technique that can be used in complex problems. From the simulation results, the system, when driven by such BFA-tuned PIDs, shows a good performance during regulation, reference tracking and parameters change tests. It can also be noted that no PSS is needed since no significant oscillations exist in the rotor speed (not shown here due to space limitations). As an extension to this work, multimachine and nonlinearities will be looked into them.

REFERENCES

- [1] P.M. Anderson and A.A. Fouad, Power system control and stability. IEEE Press, 1993.
- [2] O. I. Elgerd, Electric energy system Theory, An Introduction. McGraw-Hill, 1982.
- [3] J. Ogata, Modern Control Engineering. 3rd Edition, Prentice Hall, 1996.
- [4] K.J. Astrom, T. Hagglund, C. C. Zhang, and W.K. Ho, "Automatic tuning and adaptation for PID controllers –A survey," IFAC Journal Control Eng. Practice, vol. 1, No. 4, pp. 699-714, 1993.
- [5] Feng Zheng, Qing-Guo Wang, and Tong Heng Lee, "On the design of multivariable PID controllers via LMI approach," Automatica, vol. 38, pp. 517-526, 2002.
- [6] A. Bensenouci and A.M. Abdelghany, "Simulated annealing optimized and neural networks self-tuned PID voltage regulator for a single-machine power system," Proc. 32nd IEEE/IECON2006, Paris, France, Nov. 7-10, 2006, pp. 241-246.
- [7] A.M. Abdelghany and A. Bensenouci, "PID controllers for a one-machine infinite-bus power system using generalized Ziegler-Nichols method", Proc. 4th Saudi Technical Conference & Exhibition, STCEX'2006, Riyadh, Saudi Arabia, Dec. 2-6, 2006.
- [8] K. M. Passino, "Biomimicry of bacterial foraging for distributed optimization and control," IEEE. Control Syst. Mag., pp. 52-67, June. 2002.
- [9] B. Sumanbabu, S. Mishra, B.K. Panigrahi, and G.K. Venayagamoorthy, "Robust tuning of modern power system stabilizers using bacterial foraging algorithm," IEEE Congress on Evolutionary Computation, CEC 2007.
- [10] D.H. Kim, A. Abraham, and J.H. Cho, "A hybrid genetic algorithm and bacterial foraging approach for global optimization," Information Sciences, vol. 177, no. 18, pp. 3918-3937, 2007.
- [11] S. Dasgupta, S. Das, A. Abraham, and A. Biswas, "Adaptive computational chemotaxis in bacterial foraging optimization: an analysis," IEEE Transactions on Evolutionary Computing, (in press), 2008.
- [12] R. Allamneni, "Bacterial foraging based channel equalizers," Master of Technology, Dept. Electronics and Communication Eng., National Institute of Technology, Rourkela, Orissa, Spain, May-2006.
- [13] P.A.W. Walker and O.H. Abdallah, "Discrete control of an A.C. turbogenerator by output feedback," Proc. IEE, Control & Science, vol. 125, no. 9, Oct. 1978, pp. 1031-38.