01 Oct 2006

Bio-inspired Algorithms for the Design of Multiple Optimal Power System Stabilizers: SPPSO and BFA

Tridib Kumar Das

Ganesh K. Venayagamoorthy
Missouri University of Science and Technology

Recommended Citation

T. K. Das and G. K. Venayagamoorthy, "Bio-inspired Algorithms for the Design of Multiple Optimal Power System Stabilizers: SPPSO and BFA," *Conference Record of the 41st IEEE Industry Applications Conference (2006, Tampa, FL)*, Institute of Electrical and Electronics Engineers (IEEE), Oct 2006.
The definitive version is available at https://doi.org/10.1109/IAS.2006.256593

Bio-inspired Algorithms for the Design of Multiple Optimal Power System Stabilizers: SPPSO and BFA

Tridib K. Das and Ganesh K. Venayagamoorthy
Real-Time Power and Intelligent Systems Laboratory
Department of Electrical and Computer Engineering
University of Missouri-Rolla, Rolla, MO- 65409-0249
tkd7vc@umr.edu & gkumar@ieee.org

Abstract— **Power System Stabilizers (PSSs) provide stabilizing control signals to excitation systems to damp out inter-area and intra-area oscillations. The PSS must be optimally tuned to accommodate the variations in the system dynamics. Designing multiple optimal PSSs is a challenging task for researchers. This paper presents the comparison between two bio-inspired algorithms: a Small Population based Particle Swarm Optimization (SPPSO) and the Bacterial Foraging Algorithm (BFA) for the simultaneous tuning of a number of PSSs in a multi-machine power system. The cost function to be optimized by both algorithms takes into consideration the time domain transient responses. The effectiveness of the algorithms is evaluated and compared for damping the system oscillations during small and large disturbances. The robustness of the optimized PSSs in terms of damping is shown using the Matrix Pencil analysis.**

Keywords – bacteria foraging, matrix pencil analysis, multi-machine power system, particle swarm optimization, PSCAD, power system stabilizers, regeneration, small population.

I. INTRODUCTION

Low frequency oscillations after a disturbance in a power system if not sufficiently damped, can drive the system to instability [1, 2]. The PSSs are used to damp out the system oscillations in the range of 0.2 Hz to 2.5 Hz by providing auxiliary feedback signals to the excitation system of the generators. These oscillations come into existence when rotors of the generators oscillate with respect to each other using transmission line between them to exchange power. These oscillations are usually inter-area and intra-area modes. Depending on their location in the system, some generators participate in only one oscillation mode, while others participate in more than one mode.

The widely used conventional PSSs (CPSSs) are designed using theory of phase compensation in the frequency domain and are introduced as lead-lag compensators [2]. The power system being non-linear, fixed setting of the PSS degrades its performance. To have CPSS provide good damping over a wide range of operating range, its parameters need to be fine tuned in response to the oscillations to fit the system requirements to various modes oscillations. Several PSS design techniques have been reported in literature [3]-[9]. Local optimization techniques like gradient descent method [7], genetic algorithms [8], tabu search [9] and simulated annealing [10] are proposed to eliminate the drawbacks of conventional approach. These approaches failed to provide good optimization results when the function to be optimized is epistatic and the number of parameters involved is many. Efforts have also been taken to design an optimal neural network based PSS [11] but the computations involved in the adaptive critic design are intensive.

To make the PSS design approach immune to the drawbacks mentioned above, two bio-inspired algorithms are presented, a Small Population based Particle Swarm Optimization (SPPSO) and the Bacterial Foraging Algorithm (BFA) [12]. These algorithms take into consideration the time domain transient responses in formulating the cost function to be optimized for the simultaneous multiple PSS design. The SPPSO and BFA algorithms based multiple optimal PSS designs are evaluated and compared on a two-area benchmark system in this paper [2]. The optimal PSSs are further compared in terms of the damping ratios for different frequency modes using the Matrix Pencil method.

This paper is organized as follows: Section II presents the power system considered in this study; Section III describes the bio-inspired algorithms used; Section IV explains how the optimal parameters are determined by formulating the cost function; Section V discusses some of the simulation results obtained; Section VI highlights the benefits of the SPPSO over the BFA. Finally conclusions and future work are given in Section VII.

II. TWO AREA MULTIMACHINE POWER SYSTEM

The two area power systems used in this study is simulated in the PSCAD/EMTDC environment which allows detailed representation of the power system dynamics. The two area power system shown in Fig. 1, consists of two fully symmetrically areas linked together by two transmission lines. Each area is equipped with two identical synchronous generators rated 20kV/900 MVA. All generators are equipped with identical speed governors and turbines, exciters and AVRs and PSSs. The loads in the two areas are such that Area 1 is exporting 413 MW to Area 2. This power network is specifically designed to study low frequency electromechanical oscillations in large interconnected power systems [2].

The support from the National Science Foundation under the grant CAREER ECS #0348221 is gratefully acknowledged by the authors.

1-4244-0365-0/06/$20.00 (c) 2006 IEEE

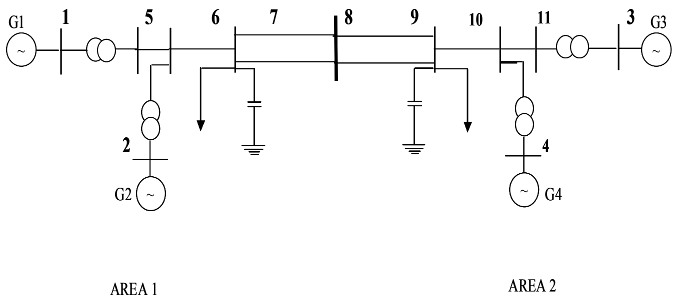

Figure 1. Two-area multi-machine power system.

The PSSs provide additional input signal (V_{pss}) to the voltage regulators/excitation systems to damp out the power oscillations. Some commonly used input signals are rotor speed deviation ($\Delta\omega_r$), accelerating power and frequency. A typical PSS block diagram is shown in Fig. 2. It consists of an amplifier block of gain constant K, a block having a washout time constant T_w and two lead-lag compensators with time constants T_1 to T_4. The gain and the four lead-lag compensator time constants are to be selected for optimal performance over a wide range of operating conditions.

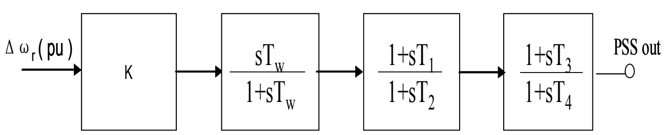

Figure 2. Block diagram of power system stabilizer.

III. BIO-INSPIRED ALGORITHMS

The beauty of PSO lies in its ability to explore and exploit the search space by varying the parameters of the PSO. BFA due to its unique elimination-dispersal events can find favorable regions when the population involved is small. These unique features of the algorithms overcome the premature convergence problem and enhance the search capability. Hence, are suitable optimization tools for PSS design. The two bio-inspired algorithms used in the multiple optimal PSS design are described below.

A. Small Population based Particle Swarm Optimization(SPPSO) algorithms

The SPPSO algorithm is derived from the Particle Swarm Optimization (PSO) algorithm. Particle swarm optimization is a form of evolutionary computation technique (a search method based on natural systems) developed by Kennedy and Eberhart [13]-[14]. PSO like GA is a population (swarm) based optimization tool. However, unlike in GA, individuals are not eliminated from the population from one generation to the next. One major difference between particle swarm and traditional evolutionary computation methods is that particles' velocities are adjusted, while evolutionary individuals' positions are acted upon; it is as if the "fate" is altered rather than the "state" of the particle swarm individuals [15].

The system initially has a population of random solutions. Each potential solution, called *particle*, is given a random velocity and is flown through the problem space. The particles have memory and each particle keeps track of previous best position and corresponding fitness. The previous best value is called the *pbest* of the particle and represented as p_{id}. Thus, p_{id} is related only to a particular particle *i*. The best value of all the particles' pbests in the swarm is called the *gbest* and is represented as p_{gd}. The basic concept of PSO technique lies in accelerating each particle towards its p_{id} and the p_{gd} locations at each time step. The amount of acceleration with respect to both p_{id} and p_{gd} locations is given random weighting.

Fig. 3 illustrates briefly the concept of PSO, where x_i is current position, x_{i+1} is modified position, v_{ini} is initial velocity, v_{mod} is modified velocity, v_{pid} is velocity considering p_{id} and v_{pgd} is velocity considering p_{gd}. The following steps explain the procedure in the standard PSO algorithm.

(i) Initialize a population of particles with random positions and velocities in *d* dimensions of the problem space.

(ii) For each particle, evaluate the desired optimization fitness function.

(iii) Compare every particle's fitness evaluation with its pbest value, p_{id}. If current value is better than p_{id}, then set p_{id} value equal to the current value and the p_{id} location equal to the current location in *d*-dimensional space.

(iv) Compare the updated pbest values with the population's previous gbest value. If any of pbest values is better than p_{gd}, then update p_{gd} and its parameters.

(v) Compute the new velocities and positions of the particles according to (3) and (4) respectively. v_{id} and x_{id} represent the velocity and position of i^{th} particle in d^{th} dimension respectively and, $rand_1$ and $rand_2$ are two uniform random functions.

$$v_{id} = w \times v_{id} + c_1 \times rand_1 (p_{id} - x_{id}) + c_2 \times rand_2 \times (p_{gd} - x_{id}) \qquad (3)$$

$$x_{id} = v_{id} + x_{id} \qquad (4)$$

(vi) Repeat from step (ii) until a specified terminal condition is met, usually a sufficiently good fitness or a maximum number of iterations.

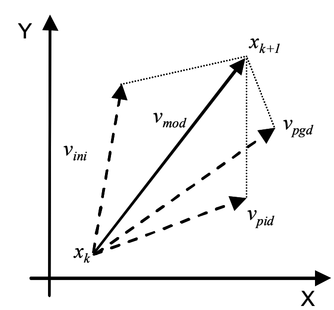

Figure 3. Movement of a PSO particle in two dimensions from one instant i to another instant *i+1*.

The PSO parameters in (3) are: *w* is called the inertia weight, which controls the exploration and exploitation of the search space. Local minima are avoided by small local neighborhood, but faster convergence is obtained by larger global neighborhood and in general, global neighborhood is preferred. Synchronous updates are more costly than the asynchronous updates.

The velocity is restricted to a certain dynamic range. v_{max} is the maximum allowable velocity for the particles i.e. in case the velocity of the particle exceeds v_{max} then it is reduced to v_{max}. Thus, resolution and fitness of search depends on v_{max}. If v_{max} is too high, then particles will move beyond good solution and if v_{max} is too low, then particles will be trapped in local minima. c_1 *and* c_2 termed as cognition and social components respectively are the acceleration constants which changes the velocity of a particle towards p_{id} and p_{gd} (generally somewhere between p_{id} and p_{gd}). Velocity determines the tension in the system. A swarm of particles can be used locally or globally in a search space. In the local version of the PSO, the p_{id} is replaced by the l_{id} and the entire procedure is same.

The SPPSO is an enhanced version of the classical PSO. This algorithm introduces the concept of regeneration to give particles the ability to keep carrying out the search despite a small population. The particles are regenerated after every N iterations retaining their previous g_{best} *(*p_{gd}*)* and p_{best} *(*p_{id}*)* fitness values. Randomizing the position and velocities of each particle every N iteration aids the particle in avoiding local minima and find global minimum. The regeneration concept reduces the number of evaluations and each evaluation is less computational intensive compared to the standard PSO algorithm.

B. Bacterial Foraging Algorithm (BFA)

Natural selection tends to eliminate animals with poor foraging strategies (methods for locating, handling and ingesting food) and favor the propagation of genes of those animals that have successful foraging strategies. Species who have better food searching ability are capable of enjoying reproductive success and the ones with poor search ability are either eliminated or reshaped. The proposed algorithm mimics the foraging behavior of E. coli present in our intestines. It is categorized into four processes: Chemotaxis, Swarming, Reproduction and Elimination [12].

a) Chemotaxis: In this process the bacteria climbs the nutrient concentration, avoid noxious substances, and search for way out of neutral media. The bacterium usually takes a tumble followed by a tumble or a tumble followed by a run. For N_c number of chemotactic steps the direction of movement after a tumble is given by:

$$\theta^i(j+1,k,l)=\theta(j,k,l)+C(i)\times\phi(j) \tag{5}$$

where

$C(i)$ is the step size taken in direction of the tumble.

j is the index for the chemotactic step taken.

k is the index for the number of reproduction step.

l is the index for the number of elimination-dispersal event.

$\phi(j)$ is the unit length random direction taken at each step.

If the cost at $\theta^i(j+1,k,l)$ is better than the cost at $\theta^i(j,k,l)$ then the bacterium takes another step of size $C(i)$ in that direction. This process will be continued until the number of steps taken is not greater than N_s.

b) Swarming: The bacteria in times of stresses release attractants to signal bacteria to swarm together. It however also releases a repellant to signal others to be at a minimum distance from it. Thus all of them will have a cell to cell attraction via attractant and cell to cell repulsion via repellant. The equation involved in the process is:

$$\begin{aligned} J_{cc}(\theta,P(j,k,l)) &= \sum_{i=1}^{S} J_{cc}{}^{i}(\theta,\theta^{i}(j,k,l)) \\ &= \sum_{i=1}^{S}[-d_{attract}\exp(-w_{attract}\sum_{m=1}^{p}(\theta_m-\theta_m{}^{i})^2)] \\ &+ \sum_{i=1}^{S}[h_{repellant}\exp(-w_{repellant}\sum_{m=1}^{p}(\theta_m-\theta_m{}^{i})^2)] \end{aligned} \tag{6}$$

where

$d_{attarct}$= depth of the attractant .

$w_{attract}$=measure of the width of the attractant.

$h_{repellant}$= $d_{attract}$ = height of the repellant effect.

$w_{repellant}$=measure of the width of the repellant.

p=Number of parameters to be optimized.

S= Number of bacteria.

The bacteria climbing on the nutrient hill can be represented by:

$$J(i,j,k,l)+J_{cc}(\theta,P) \tag{7}$$

where $J(i,j,k,l)$ is the cost function.

c) Reproduction: After all the N_c chemotactic steps have been covered, a reproduction step takes place. The fitness (accumulated cost) of the bacteria are sorted in ascending order. Sr (Sr=S/2) bacteria having higher fitness die and the remaining S_r are allowed to split into two thus keeping the population size constant.

d) Elimination-Dispersal: For each elimination-dispersal event each bacterium is eliminated with a probability of p_{ed}. A low value of Ned 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 bacteria to find favorable regions. The p_{ed} should not be large either or else it should lead to an exhaustive search.

IV. Optimal PSS Design Cost Function

This section describes how the algorithms are used to determine the parameters of the PSSs of the four generating units in Fig. 1. For each of the PSS, the optimal parameters are determined by the SPPSO and the BFA, i.e. 20 parameters in total for the two area power system. The objective of the algorithms implemented is to maximize damping; this means minimize the overshoots and settling time in system oscillations. The time response of the four generators under transient conditions is minimized by the algorithms.

The time response of the generators is used as the fitness function which is to be optimized by the bio-inspired algorithms so as to improve the performance of the system under transient conditions. The optimization is carried by subjecting the system to a 200ms short circuit and a 200ms line outage.

The PSS parameters are determined by optimizing a multi-objective objective function given by (8)

$$J^t = \sum_{n=1}^{N}\sum_{G_n}^{m} J_{G_n} \tag{8}$$

where

$$J_{G_n} = \sum_{j=1}^{NP}\sum_{t=t_0}^{t_2/\Delta t} \left(\Delta\omega(t)\right)\times\left(A\times(t-t_0)\times\Delta t\right) \tag{9}$$

where

NP is the number of operating points for which optimization is carried out .

N is the number of faults for which the optimization is carried out..

A is the weighing factor

m is the number of generators in the system

$\Delta\omega_{Gn}$ is the speed deviation of the generator Gn.

t_0 is the time the fault is cleared .

t_2- t_0 is the transient period considered for area calculation.

Δt is the speed signal sampling period.

t is the simulation time in seconds.

V. SIMULATION RESULTS

The entire simulation is carried out in PSCAD/EMTDC /FORTRAN environment. The challenging task of tuning multiple PSSs using the bio-inspired algorithms in PSCAD is reported in this paper for the first time to the knowledge of the authors. The number of particles used in SPPSO is five and the number of bacteria in BFA is four.

The performance of the PSS optimized by the bio-inspired algorithms is tested under small and large disturbances. Results are presented for four cases as described below.

Case 1

In this case there are no PSSs connected to the system.

Case 2

The PSSs parameters in this case are the Kundur's parameters [16]. These parameters are as follows: K= 20.0, T_1=0.05, T_2=0.02s, T_3=3.0s and T_4=5.4s respectively.

Case 3

The PSSs parameters used in this case are the BFA optimized parameters.

Case 4

The parameters used in this case are the parameters optimized by using the SPPSO algorithm.

The three tests are carried out and the responses are studied for the above mentioned cases in the two area power system

A. *Test 1*

A three phase short circuit test of 200ms duration is applied at bus 8 in Fig.1. The speed responses of the generators for Cases 1-4 are shown in Figs. 4 and 5.

B. *Test 2*

A 200ms line outage is applied between buses 8 and 9 of Fig.1. The speed responses of the generators for Cases 1-4 are shown in Figs. 6 and 7.

C. *Test 3*

A 100ms short circuit at bus 8 immediately followed by a 100ms line outage between buses 8 and 9 in Fig.1. The speed responses of the generators for Cases 1-4 are shown in Figs. 8 and 9.

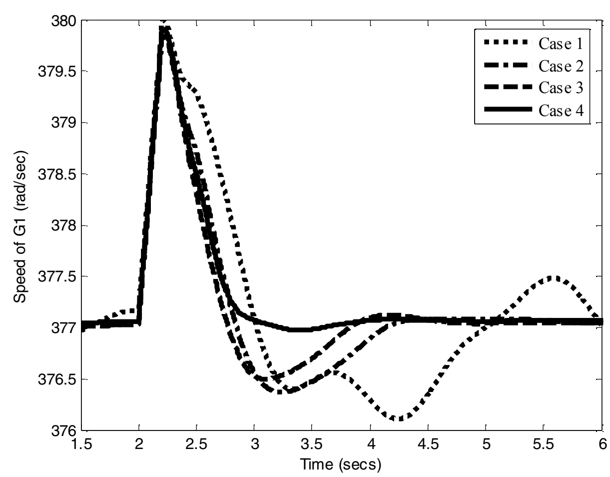

Figure 4. Speed response of generator G1 for a 3 phase 200ms short circuit at applied at bus 8.

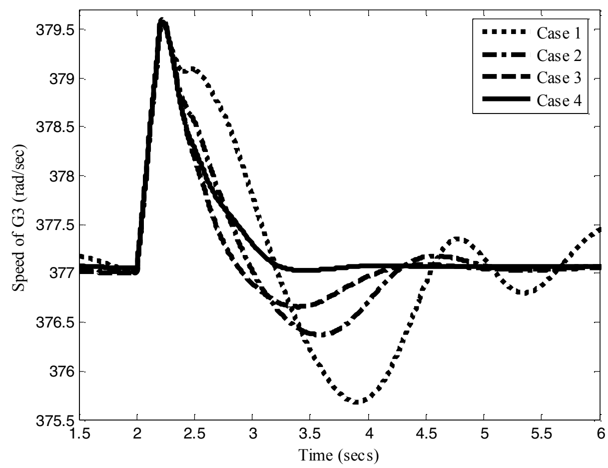

Figure 5. Speed response of generator G3 for a 3 phase 200ms short circuit at applied at bus 8.

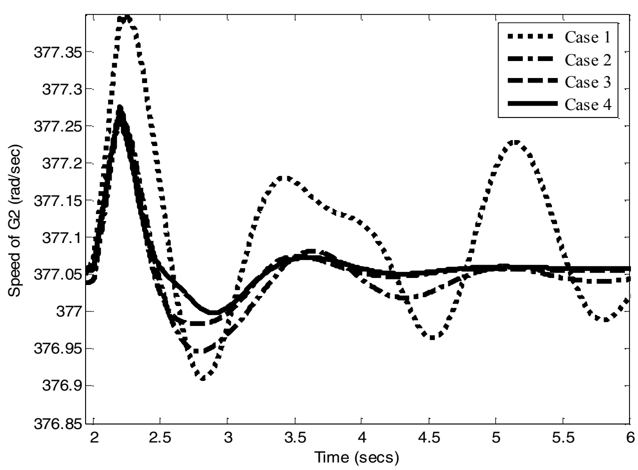

Figure 6. Speed response of generator G2 for a 200ms line outage applied between buses 8 and 9.

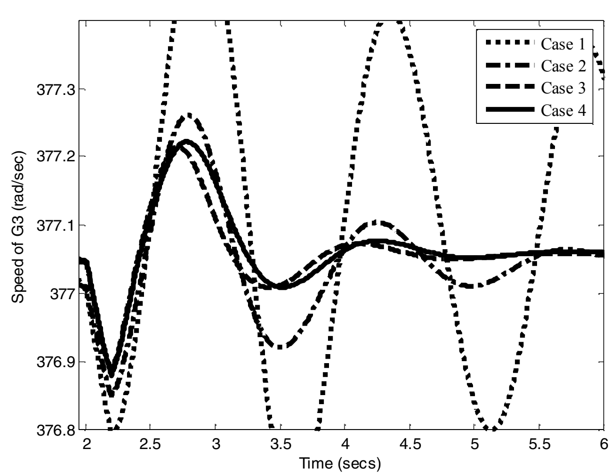

Figure 7. Speed response of generator G3 for a 200ms line outage applied between buses 8 and 9.

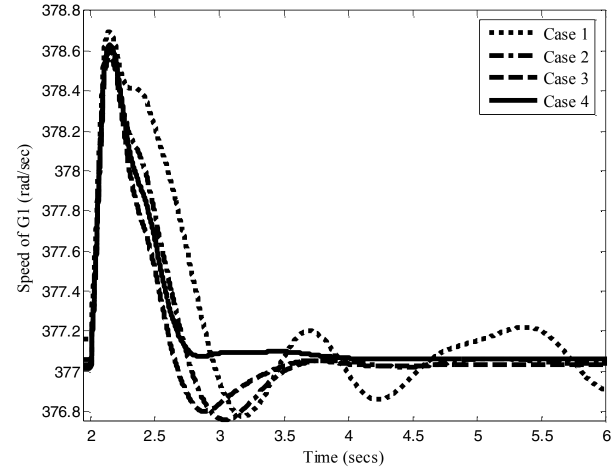

Figure 8. Speed response of G1 for 3 phase 100ms short circuit applied at bus 8, followed by immediate 100ms line outage between buses 8 and 9.

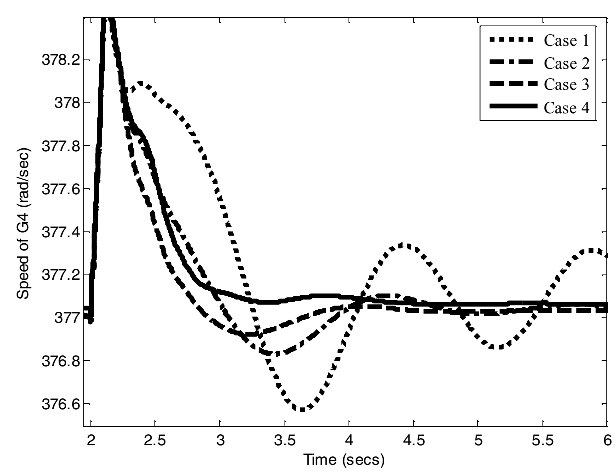

Figure 9. Speed response of G1 for 3 phase 100ms short circuit applied at bus 8, followed by immediate 100ms line outage between buses 8 and 9.

VI. EVALUATIONS OF DAMPING PERFORMANCE

The number of fitness evaluations in case of BFA is more than that of SPPSO. In BFA, for each particle the fitness is evaluated a number of times. The number of processes involved makes the algorithm computationally intensive. In BFA, a fitness evaluation is done after all the chemotactic steps are covered, hence for each fitness evaluation $S \times N_c$ evaluations are needed. In the case of SPPSO, a single fitness evaluation is carried out after covering S particles. The number of factors involved in BFA makes it more dependent. These factors need to be properly chosen for the algorithm to perform better. This can be a serious handicap of the algorithm. The fitness of the best particle with the number of iterations is shown in Fig. 10. It shows that the fitness of SPPSO converges faster and to a lower value compared to BFA.

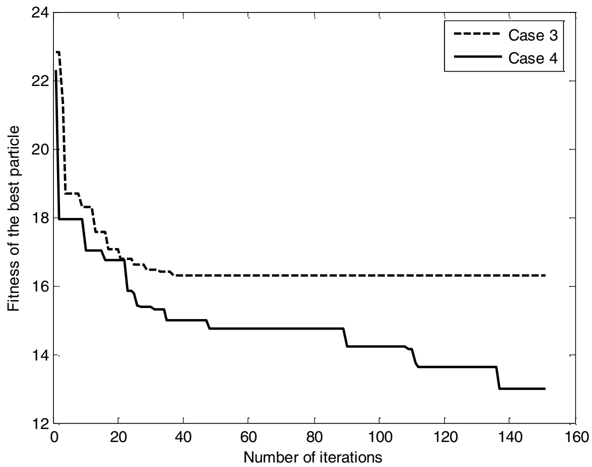

Figure 10. Fitness of the best particle/bacteria.

A brief comparison of the two algorithms based on the transient energy calculations is shown in Tables I- III. The transient energy of each of the generator for the first 5 seconds of the fault has been calculated using equation (10)

$$TE_{Gen_i} = \frac{1}{2} H_{Gen_i} \int_{t_{flt}}^{t_{flt}+5} \Delta\omega_i^2 dt \quad (10)$$

where i is the generator number and t_{flt} is the time the fault is triggered. Tables I, II and III present the normalized transient energies of generators G1, G2, G3 and G4 due to short circuit of 200ms at bus 8, line outage of 200ms between buses 8 and 9 and a short circuit and line outage combined respectively. The results show that the normalized transient energy is the least in Case 4 which are obtained when the system has SPPSO optimized PSSs. This corroborates the superiority of the SPPSO algorithm over the BFA for same operating conditions. Table IV shows the best parameters obtained by the BFA and SPPSO algorithms over 10 trials.

TABLE I. NORMALIZED TRANSIENT ENERGY DURING SHORT CIRCUIT

Generator	Case 1	Case 2	Case 3	Case 4
G1	1.0	0.6243	0.5279	0.5111
G2	1.0	0.6206	0.5247	0.5081
G3	1.0	0.6073	0.5135	0.4972
G4	1.0	0.6147	0.5198	0.5032

TABLE II. NORMALIZED TRANSIENT ENERGY DURING LINE OUTAGE

Generator	Case 1	Case 2	Case 3	Case 4
G1	1.0	0.2588	0.2082	0.1792
G2	1.0	0.4451	0.3588	0.3083
G3	1.0	0.0810	0.06519	0.0561
G4	1.0	0.0969	0.0780	0.0671

TABLE III. NORMALIZED TRANSIENT ENERGY DURING SHORT CIRCUIT AND LINE OUTAGE

Generator	Case 1	Case 2	Case 3	Case 4
G1	1.0	0.6363	0.5176	0.5061
G2	1.0	0.6472	0.5264	0.5148
G3	1.0	0.677	0.5507	0.5385
G4	1.0	0.6945	0.5650	0.5525

TABLE IV. TWO AREA POWER SYSTEM PSS PARAMETERS

Generator	BFA optimized parameters	SPPSO optimized parameters
G1	K = 30.0, T_1 = 0.5, T_2 = 0.5, T_3 = 10.0, T_4 = 13.61	K = 30.0, T_1 = 2.0, T_2 = 0.7097, T_3 = 3.5332, T_4 = 13.995
G2	K = 30.0, T_1 = 0.5, T_2 = 0.5, T_3 = 10.0, T_4 = 13.506	K =23.4297, T_1 = 2.0, T_2 = 0.398, T_3 = 3.947, T_4 =15.0
G3	K = 30.0, T_1 = 0.5, T_2 = 0.5, T_3 = 10.0, T_4 = 13.7.	K = 9.604, T_1 = 0.9955, T_2 = 0.9711, T_3 = 10.0, T_4 = 4.9313
G4	K = 29.881, T_1 = 0.5, T_2 = 0.5, T_3 = 10.0, T_4 = 15.0	K = 30.0, T_1 = .2447, T_2 = 0.7836, T_3 = 3.58, T_4 = 13.785

VII. CONCLUSION

The successful implementation of the two bio-inspired algorithms for simultaneous tuning of the multiple PSSs has been presented in this paper. Both of the algorithms give robust damping performance for various operating conditions and severity of disturbances. The SPPSO owing to its regeneration concept is shown to have faster convergence and requires less number of fitness evaluations than the standard PSO. BFA owing to its unique processes involved can find the good optimal solutions. The SPPSO however is found to be superior to the BFA both in number of fitness evaluations, the convergence speed and damping performances.

The paper has presented these algorithms as an optimization tool in the PSCAD/EMTDC environment. This is a first step towards online optimization and future work can involve developing these algorithms further for real-time optimization in power systems.

REFERENCES

[1] G. K. Venayagamoorthy, R. G. Harley, “Two separate continually online trained neurocontrollers for excitation and turbine control of a turbogeneraotr,” vol. 13, Issue 3, pp. 764-773, May 2002.

[2] P. Kundur, M. Klein, G. J Rogrers and M. S. Zywno, “Application of power sytem stabilizers for enhancement of overall stability”, IEEE Transactions on Power System, vol. 4, pp. 614-626, 1989.

[3] D. Xia and G.T. Heydt, “Self tuning control for generator excitation control, ” IEEE Transactions on Power Application Syste, pp. 1877-1885, 1983.

[4] Y. Cao, L. Jiang, S. Cheng, D.Chen, O.P. Mallik and G. S. Hope, “A non-linear variable structure stabilizer for power system stability,” IEEE Trasactions on Energy Conversion, vol. 9, pp. 489-495, Sept. 1994.

[5] S. Abe and A. Doi, “A neew power syetm stabiizer synthesis in mulimachne power systems,” IEEE Transactions on Power System, vol. PAS-102, pp. 3910-3918, 1983.

[6] J. M. Arredono, “Results of a study on location and tuning of power system stabilizers,” International. Journal on Electric Power Energy System, vol. 19, pp. 563-567, 1997.

[7] V. A. MAslennikov and S. M. Ustinov, “The optimization method for coordinated tuning of power system regulators,” in Proceedings on 12th Power System Computer Confernce, PSCC, Dresden, Germany, 1996, pp. 70-75.

[8] Y. L. Abdel-Magid, M. A Abido, S. Al-Baiyat and H. Manatawy, “Simultaneous stabilization of multimachine power system via geneteic algorithms,” IEEE Transactions on Power System, vol. 145, pp. 1428-1439, Nov 1999.

[9] M. A. Abido, “A novel approach to conventional power syetm stabilizer design using tabu search,” International Journal Electric Power Energy System, vol. 21, pp. 443-454, June 1999.

[10] – “Robust design of multimachine power system stabilizers using simulated annealing,” IEEE Transactions on Energy Conversion, vol. 15, pp.297-304, Sept. 2000.

[11] W. Lui, G. K. Venayagamoorthy, D. Wunsch II, “A heuristic-dynamic based power system stabilizer for a turbogenerator in a sinlge machine power system,” IEEE Transactions on Industry Applications, vol. 41, Issue 5, pp. 1377-1385, Sept-Oct. 2005.

[12] K. M. Passino, “Biomimicry of bacterial foraging for distributed optimization and control,” Control System Magazine, IEEE, vol. 22, Issue 3, pp. 52-67, June 2002.

[13] J. Kennedy and R. Eberhart, “ Particle Swarm Optimization ,” IEEE International Conference on Neural Networks, Perth, Australia. vol. 4, pp. 1942-1948, Dec 1995.

[14] Y. Shi and R. Eberhart, “Empirical study of particel swarm optimization,” Proceedings of the 1999 Congress on Evolutionary Computation, CEC 99, vol. 3, 1999.

[15] J. Kennedy, R. C. Eberhart and Y. Shi, Swarm Intelligence, Morgan Kaufmann Publishers, 2001.

[16] P. Kundur, Power System Stability and Control, McGrawHill, New York: 1974, pp. 814.