Comparison between Three Metaheuristics Applied to Robust Power System Stabilizer Design

Med. MEKHANET, L. MOKRANI and Med. LAHDEB

***Abstract** - This paper presents a PSS (Power System Stabilizer) design using Particle Swarm Optimization (PSO), Genetic Algorithms (GA) and Simulated Annealing (SA) methods. The design is considered for multimachine power systems. The main motivation for this design is to damp electromechanical oscillations and stabilize low-frequency oscillations of power systems. The lead-lag PSS parameters are tuned by formulating an optimization problem which is solved by these three meta-heuristic techniques, to reach optimal global stability.*

This approach has been applied to the WSCC (Western System Coordinating Council), which is a three-machine nine-bus system has, to design optimal PSSs for different operating conditions and disturbances and different loads. A comparison between the three techniques, in terms of performance and calculation time consumption, is carried out through simulation results.

Keywords: Lead-lag PSS, Dynamic Stability, Genetic Algorithms, Particle Swarm Optimization, Simulated Annealing, Multimachine Power Systems.

1. INTRODUCTION

Stability of power systems has mainly depended on the stability of their generators. Automatic Voltage Regulator (AVR) has been introduced to stabilize the voltage, consequently this choice affect the dynamic stability of the power system. A device which is placed in excitation system (Power System Stabilizer: PSS) is used to generate supplementary control signals for the excitation system in order to damp the low frequency local and intra-area oscillations for 0.1 to 3 Hz [1] [2] [3].

The parameters of the CPSS (Conventional Power System Stabilizer) are determined based on a linearized model of the power system around a nominal operating point where they can provide good performance [3]. Because power systems are highly nonlinear systems, with configurations and parameters that change with time, the CPSS design based on the linearized model of the power systems cannot guarantee its performance in a practical operating environment [1]. To improve the performance of the CPSS, numerous techniques have been proposed for their design [1] [3] [4] such as using intelligent optimization methods.

Meta-heuristic techniques are a new family of stochastic algorithms which aim to solve difficult optimization problems. Used to solve various applicative problems, these methods have the advantage to be generally efficient on a large amount of problems.GA and PSO belonged to population approaches but SA algorithms belong to trajectory approaches.

Manuscript received January 11, 2012.

Meta-heuristics are generally used to solve a simplified OPF (Optimal Power Flow) problem such as the classic economic dispatch, security - constrained economic power dispatch, and reactive optimization problem, as well as optimal reconfiguration of an electric distribution network [5].

Genetic algorithms (GAs) were invented by John Holland in the 1960s and were developed by Holland and his students and colleagues at the University of Michigan in the 1960s and the 1970s. In contrast with evolution strategies and evolutionary programming, Holland's original goal was not to design algorithms to solve specific problems, but rather to formally study the phenomenon of adaptation as it occurs in nature and to develop ways in which the mechanisms of natural adaptation might be imported into computer systems [6] [7].

The Particle Swarm Optimization (PSO) strategy is a new class of algorithms proposed to solve continuous optimization problems [8] [9] [10].

The Particle Swarm Optimizer was introduced by James Kennedy and Russell Eberhart in 1995. Inspired by social behavior and movement dynamics of insects, birds and fish, it is also related, however, to evolutionary computation, and has links to both genetic algorithms and evolution strategies.

The origin of the SA algorithm is in statistical mechanics. Kirkpatrick *et al*. originated the idea of using the annealing method in optimization problems [11]. Annealing is the metallurgical process of heating up a solid and then cooling slowly until it crystallizes. Atoms of this material have high energies at very high temperatures. This gives the atoms a great deal of

© 2012 – Mediamira Science Publisher. All rights reserved.

freedom in their ability to restructure themselves. As the temperature is reduced the energy of these atoms decreases, until a state of minimum energy is achieved. In an optimization context SA seeks to emulate this process [12]. Generic probabilistic meta-heuristic for the global optimization problem of applied mathematics, namely locating a good approximation to the global optimum of a given function in a large search space. The core of the SA is laid in a way allowing escaping from local optima in order to find the possibility best global solution; even if the new created solution aggravates the fitness solution [13].

The problem of PSS design is as an optimization problem with constraints. Then, GA, PSO and SA algorithms can be employed to solve this optimization problem. Simulation results have been carried out to compare and assess the effectiveness of these methods under different disturbances and loading conditions [8] [9] [10].

2. POWER SYSTEM MODELING

2.1. Power system model

The complex nonlinear model related to an n–machine interconnected power system [14] [15], can be described by a set of differential-algebraic equations.

For a given operating condition, the multimachine power system model is linearized around the operating point.

The closed loop eigenvalues of the system are then computed and the desired objective functions are formulated using only the unstable or lightly damped electromechanical eigenvalues, keeping the constraints of all the system modes stable under any condition [1] [3]. For a multi-machine power system of n generators the state space model can be written as follows [16] [17] [18]:

$$\frac{d\underline{X}(t)}{dt} = A\underline{X}(t) + B\underline{U}(t) + \Gamma\underline{P}(t) \qquad (1)$$

Where, $\underline{X}(t), \underline{U}(t)$ et $\underline{P}(t)$ are the state variables, the control and the disturbance vectors respectively, with:

$$\underline{X}(t) = \left[\Delta\omega_1\ \Delta\omega_2 \cdots \Delta\omega_n\ \ \Delta\delta_1\ \Delta\delta_2 \cdots \Delta\delta_n\ \ \Delta E'_{q1}\ \Delta E'_{q2} \cdots \Delta E'_{qn}\ \ \Delta E_{fd1}\ \Delta E_{fd2} \cdots \Delta E_{fdn}\right]^T \qquad (2)$$

$$\underline{U}(t) = [\Delta U_1\ \Delta U_2 \cdots\cdots \Delta U_n]^T \qquad (3)$$

$$\underline{P}(t) = [\Delta P_{m1}\ \Delta P_{m2} \cdots\cdots \Delta P_{mn}]^T \qquad (4)$$

For a single machine system n=1 these equations can be written as:

$$\underline{X}(t) = \left[\Delta\omega, \Delta\delta, \Delta E'_q, \Delta E_{fd}\right]^T \qquad (5)$$

Where:

$$A = \begin{bmatrix} -D/_{2H} & -K_1/_{2H} & -K_2/_{2H} & 0 \\ 2\pi f & 0 & 0 & 0 \\ 0 & -K_3/_{T'_{do}} & -K_4/_{T'_{do}} & -1/_{T'_{do}} \\ 0 & -K_A K_5/_{T_A} & -K_A K_6/_{T_A} & -1/_{T_A} \end{bmatrix} \qquad (6)$$

$$B = \begin{bmatrix} 0 & 0 & 0 & K_A/_{T_A} \end{bmatrix}^T \qquad (7)$$

$$\Gamma = \begin{bmatrix} 1/_{2H} & 0 & 0 & 0 \end{bmatrix}^T \qquad (8)$$

The control vector $\underline{U}(t)$ is a vector of stabilizing signals that represents the PSS output at different machines. The dynamic equations of the PSS in state-space form, as obtained from the transfer function block-diagram, are given below [19]:

$$\begin{aligned} \frac{d(\Delta N_{1i}(t))}{dt} &= K_c \frac{d(\Delta\omega_i)}{dt} - \frac{\Delta N_{1i}(t)}{T_{\omega_i}} \\ \frac{d(\Delta N_{2i}(t))}{dt} &= \left[\Delta N_{1i}(t) - \Delta N_{2i}(t) + T_{1i}\frac{d(\Delta N_{1i}(t))}{dt}\right]\frac{1}{T_{2_i}} \\ \frac{d(\Delta U_i(t))}{dt} &= \left[\Delta N_{2i}(t) + T_{3i}\frac{d(\Delta N_{2i}(t))}{dt} - \Delta U_i(t)\right]\frac{1}{T_{4_i}} \end{aligned} \qquad (9)$$

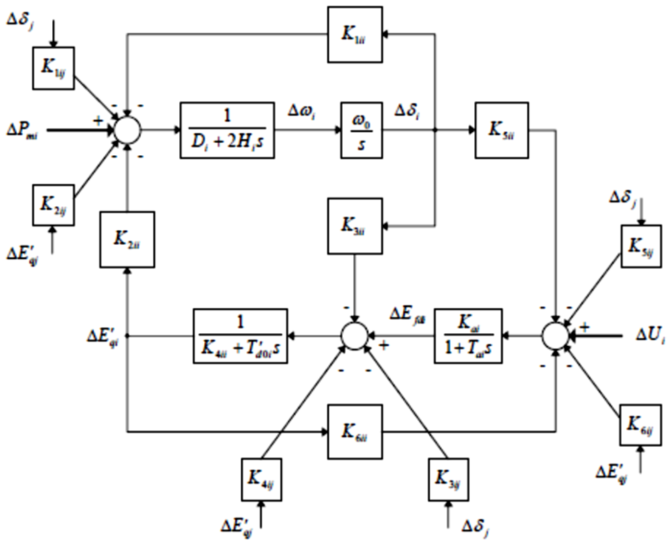

Figure 1. A linear model of a multi-machine power system.

Where ΔN_{1i} and ΔN_{2i} are the state-variables associated with each PSS, $T_{\omega i}$ is the washout time constant, $T_{1i}, \ldots T_{4i}$ are the phase-lead time constants and K_c is the stabilizer gain.

2.2. PSS structure

In the design of PSS, The linearized incremental models around an equilibrium point are usually employed.

The operating function of a PSS is to produce a proper torque on the rotor of the machine involved in such a way that the phase lag between the exciter input and the machine electrical torque is compensated. A widely speed based used conventional PSS is considered throughout the study.

The transfer function of the lead-lag PSS is given by[18] [20]:

$$\frac{u(s)}{\Delta\omega(s)} = K_c \frac{sT_\omega}{1+sT_\omega}\left(\frac{1+sT_1}{1+sT_2}\right)\left(\frac{1+sT_3}{1+sT_4}\right) \qquad (10)$$

T_ω is a filter constant used to block unwanted frequencies below 0.1 Hz. This value is not really critical; it is generally taken between 1 s and 20 s. In this study it is set to 10 s.

K_c is the gain that will mitigate the oscillations of the rotor system. Since the aim of the lag blocks is to establish a sufficient phase lag to damp the system low frequency oscillations, one can take $T_1 = T_3$ and $T_2 = T_4$. Finally, equation) becomes:

$$\frac{u(s)}{\Delta\omega(s)} = K_c \frac{10s}{1+10s}\left(\frac{1+sT_1}{1+sT_2}\right)^2 \quad (11)$$

So the final block-diagram of this kind of lead-lag PSS is as follows [19] [20]:

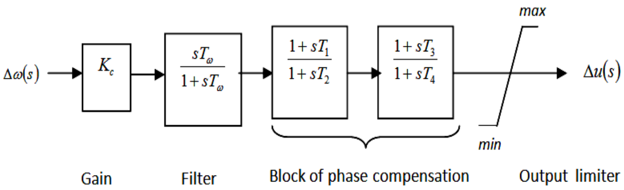

Figure 2. Block diagram of a lead-lag phase PSS.

2.3. Objective Function

Normally, the PSS stabilizes the system regardless of its operating point. The parameters of the PSS will be tuned, for each operating point considering a linear model of the electrical system.

This is formulated as a multivariable and nonlinear optimization problem in order to maximize a constrained cost function.

First of all, one may classify the damping factors of the k^{th} operating point in the vector $\aleph_k$.The objective is tune the PSSs parameters in limited search space, ensuring an acceptable damping for all operating points. To do this we must ensure that at the end all the damping factors are higher than an acceptable value.

Then, the optimization problem to be solved by the AG, PSO or SA algorithms can be written in the following from:

Maximize:

$$F(K_{cj}, T_{1j}, T_{2j}) = \min(min(\aleph_k)) \quad k = 1, \ldots m \quad (12)$$

Subject to the following constraints:

$$0.001 \leq K_{cj} \leq 50$$

$$0.001s \leq T_{1j} \leq 2s$$

$$0.001s \leq T_{2j} \leq 2s$$

Where:

- m is the total number of the operating points;
- j is the PSS index

The problem thus defined is a complex optimization problem because the objective function depends on the eigenvalues of a large matrix. It is difficult to solve it using conventional methods [21].

The damping coefficients ξ_i are calculated from the eigenvalues as:

$$\lambda_i = \alpha_i \pm j\beta_i \quad (12)$$

$$\xi_i = \frac{-\alpha_i}{\sqrt{\alpha_i^2 + \beta_i^2}} \quad (13)$$

Where *"i "* is the index of the mode.

In this work different meta-heuristic methods are used to solve this optimization problem and search for an optimal set of PSS parameters, K_c,T_1 and T_2.

3. ALGORITHMS

3.1. Genetic Algorithm

Genetic Algorithms are global search techniques providing a powerful tool for optimization problems by miming the mechanisms of natural selection and genetics [19].

The sequential steps for searching the optimal solution of PSS parameters using GA are shown in the following steps:

1. Set parameters of GA:
   - Number of bits on which we can encode a parameter
   - Number of parameters to optimize for each PSS (K_c,T_1, T_2)
   - Number of PSSs
   - Number of possible combinations (for the choice of the initial population)
   - Number of individuals (must be multiple of the combinations number)
   - Probability of crossover
   - Number of crossing points
   - Mutation probability
2. Choose the number of generations
   - Number of generations "*MaxGen*"
3. Initialize the population, encoding and decoding
4. Repeat the following, WHILE the generation number is below "*MaxGen*":
   - Evaluation of individuals (fitness function)
   - Selection (by class)
   - Crossing
   - Mutation
5. Output optimal solution.

3.2. PSO Algorithm

The particle swarm optimization is based on a set of individuals originally randomly arranged and homogeneous. Therefore we call it particles, which move in the hyperspace of research and are each a potential solution. Each particle has a memory about his best seen as the ability to communicate with the particles forming around it. From this information, the particle will follow a trend made, from one side, willingness to return to its optimal solution, and from the other side, his mimicry in relation to the solutions found in its vicinity. From local optima and empirical, all particles will normally converge to the global optimum solution of the addressed problem [22].

The process of finding the particles is based on two rules:

1) Each particle has a memory that can store the best point by which it has already passed and it tends to return to this point;
2) Each particle is informed of the best known point in its neighborhood and it will tend to move towards this point.

Each particle moves according to a compromise between the 3 following trends:

- Repeat its previous motion;
- Move towards its best previous position;
- Move towards the best position (past) of its group of informants.

Each agent tries to modify its position based on the following information [23]:

- Current positions (x, y)
- Current velocities (v_x, v_y)
- Distance between the current position and ***pbest***,
- Distance between the current position and *gbest*.

Thus, the velocity of the particle i is updated using the following equation (18):

$$v_i^{k+1} = wv_i^k + c_1 rand_1 (pbest - S_i^k) + c_2 rand_2 (gbest - S_i^k) \quad (14)$$

Where w is the inertia weight, c_1 and c_2 are the acceleration constants ($c_1 + c_2 \leq 4$, [24]), $rand_1$ and $rand_2$ are random numbers in the interval [0 1].

Then the position of the particle S_i^k is modified from the current position and a new speed is calculated v_i^{k+1} :

$$S_i^{k+1} = S_i^k + v_i^{k+1} \quad (15)$$

The weight w is given by the following equation:

$$w = w_{max} - \frac{w_{max} - w_{min}}{iter_{max}} iter \quad (16)$$

w= [0.4 - 0.9] during the search procedure gives better results [23].

The right choice of parameters will allow the rapid convergence and minimizes the computation time, details of choice is quoted, in [25]:

The first rule stipulates that c_1 must have an absolute value less than 1in practice, this coefficient should be neither too small on PSO recommended that it be equalized to 0.7 or 0.8.

The parameter c_2 should not be too large, a value of about 1.5 to 1.7 being regarded as effective in the majority of cases. The pairs of values (0.7 1.47) and (0.8 1.62) are indeed correct. The following figure shows the general flowchart of PSO.

The steps involved in the optimization algorithmic of the particle swarm are as follows [22] [23] [26]:

Step1: Select several parameters of PSO;

Step2: Initialize a population of particles with random positions and velocities in the problem search space;

Step3: Evaluate the ability of optimization for a desired personal touch to each particle;

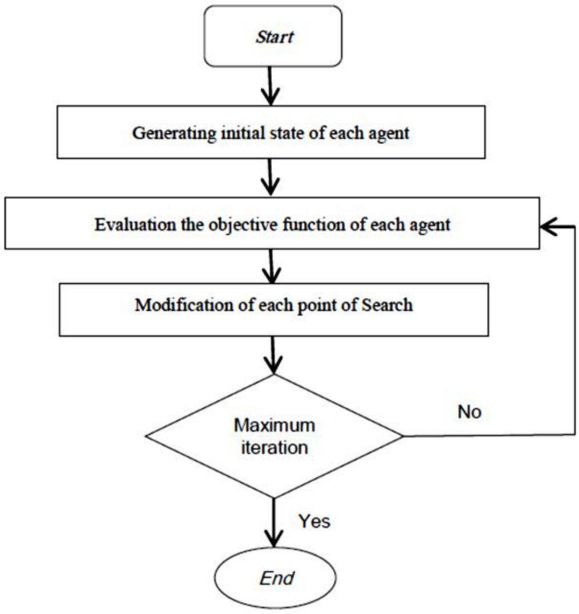

Figure 3. General flowchart of the PSO algorithm.

Step4: For each individual particle, compare the value of the particle with its ability *pbest*. If the current value is better than the evaluated *pbest*, then set this as *pbest* for the agent *i*;

Step5: Identify the particle that has the best value of its fitness function which will be identified as *gbest;*

Step6: Calculate the new speed and position of particles using equations 15 and 16.

Step7: Repeat steps 3-6 until the stopping criterion is met.

3.3. SA Algorithm

Simulated annealing is an optimization method for analog simulation of a process encountered in metallurgy, annealing of metals. Annealing is a physical process of heating. Thus, when heating a solid metal, it becomes liquid at a certain temperature, in which case the atoms that compose saw their degree of freedom increases, conversely when the temperature is lowered the degree of freedom decreases to a solid lead.

In a SA algorithm, it is attempted to avoid becoming trapped in a local optimum by sometimes accepting a neighborhood move which increases the value of the fitness function [13] [27] .

Now, depending on how the temperature is reduced, one may get different solid:

1. A sudden drop in temperature (tempering), produces an amorphous structure, a glass, and then we have a local minimum of energy;
2. A gradual decline in temperature will permit to reach the global minimum of energy. One may obtain in this case a crystal (best thermodynamic equilibrium).

The state of a material depends on the temperature, at which it is worn. One can achieve a balance heat at each temperature. This requires a large number of transitions that must occur at each temperature. This thermal equilibrium is characterized by the Boltzmann distribution. The probability *P* (*x*) to visit a state *X* in terms of its energy *E*(*x*) and temperature *T*, is given by the following relationship (15):

$$P(x) = \frac{e^{E(X)/KT}}{N(T)} \quad (17)$$

Where *K* is the Boltzmann constant ($1.3806503 \times 10^{-23}$ $m^2.kg.s^{-2}.K^{-1}$) and *N*(*T*) is the partition function.

The analogy between a physical annealing process and a combinatorial optimization problem is based on the following:

- The solutions obtained for the optimization problem are equivalent to configurations of a physical system;
- The cost of a solution is equivalent to the energy of a configuration.

The basic algorithm of the method of simulated annealing is described as follows:

1. Initialization:

- Choose an initial solution
- Choose an initial temperature

2. As long as no stop criterion is satisfied do:

- Generate a random neighborhood
- Choose a neighborhood
- Select the neighboring competitor
- Update the current configuration
- Update the temperature

3. End if stop criterion is satisfied.

4. SIMULATION RESULTS

In this part of the study, a three-machine nine-bus power system of Figure 4 (which shows the single line diagram of the network WSCC (Western System Coordinating Council) three machine nine-bus system [19]) is considered. It operates at different loading conditions. Details of the system data and operating conditions are given in [18] [19] [28] and depicted in Tables 1 and 2.

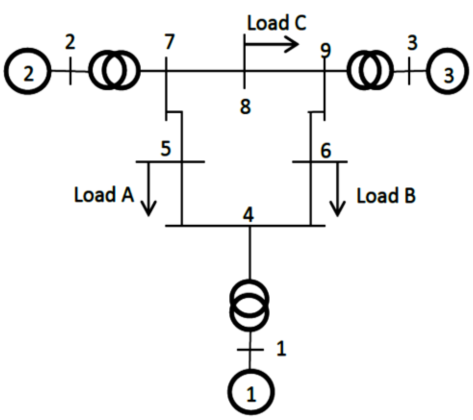

Figure 4. Three-machine nine-bus system.

Three operating points are considered to cover a wide range of load variation:

Table 1. Operating points of the WSCC

	Operating conditions (in per unit)					
	Light		Nominal		Heavy	
	P	***Q***	***P***	***Q***	***P***	***Q***
G1	**0.362**	**0.162**	**0.716**	**0.271**	**2.207**	**1.088**
G2	**0.800**	**-0.109**	**1.630**	**0.067**	**1.920**	**0.564**
G3	**0.450**	**-0.204**	**0.850**	**-0.107**	**1.280**	**0.359**

Table 2. Loading conditions

load	Loading conditions (in per unit)					
	Light		Nominal		Heavy	
	P	***Q***	***P***	***Q***	***P***	***Q***
A	**0.65**	**0.55**	**1.25**	**0.50**	**2.00**	**0.80**
B	**0.45**	**0.35**	**0.90**	**0.30**	**1.80**	**0.60**
C	**0.50**	**0.25**	**1.00**	**0.35**	**1.50**	**0.60**

A linear system is obtained after linearization. Then, the study of the system eigenvalues for the nominal operating point is carried out. Figure **5** shows that the system is stable but very poorly damped.

To verify this in the time domain, the dynamic response of the system (see Figure 6) is carried out for the three mentioned cases of operating conditions. The following scenario has been considered:

- A three-phase short circuit near the highest power generator (node 7) is created at *t* = 1 second;
- The line (7-5) is opened at *t* = 1.09 seconds;
- Then this line is closed at *t* = 1.1 seconds.

The simulation time is about 10 seconds.

We find that, indeed, the system is poorly damped because it takes a long time to return to its stable state.

Elsewhere, the method of participation factors was applied and the results are shown in the Table 3.These results indicate that it will be interesting if the generators ***2*** and ***3*** are equipped with PSSs.

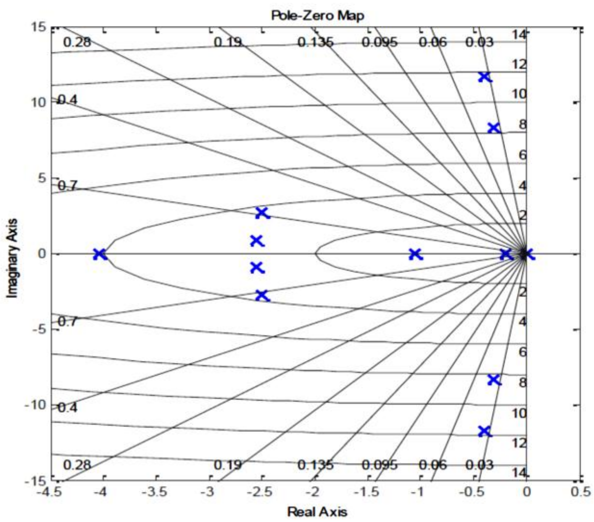

Figure 5. Eigenvalues of electromechanical modes without PSSs.

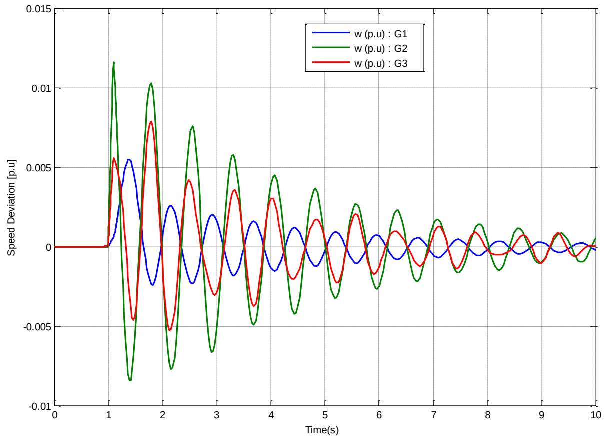

Figure 6. Dynamic response of the system for a Nominal Load without PSSs.

Table 3. Participation factors of the system generators

Load	Mode	*G1*	*G2*	*G3*
Light	**-0.4009 ±9.7572i**	**0.0138**	***0.1318***	***0.8349***
	-0.3346 ±7.5394i	**0.2269**	***0.6819***	***0.0694***
Nominal	**-0.4063±11.7451i**	**0.0085**	***0.1520***	***0.8291***
	-0.3080±08.3057i	**0.2561**	***0.6559***	***0.0838***
Heavy	**-0.3643 ±11.9296i**	**0.0097**	***0.1466***	***0.8389***
	-0.3079 ±8.2038i	**0.2371**	***0.6777***	***0.0831***

Now let us optimize the PSSs parameters of the two generators *G2* and *G3* using the three techniques (GA, PSO and SA) and compare the results.

The PSSs parameters obtained by the use of different algorithms are given in (Table 4, Table 5 and Table 6). In order to increase the search space, the following operating conditions are considered:

Table 4. Settings of the genetic algorithm

AG parameters	
Number of bits (binary encoding)	8
Number of individuals	128
Probability of crossover	0.9
Number of crossing points	2
Mutation probability	(0.09, 0.09]
Iteration Max	50
Number of combination	64

Table 5. Settings of the PSO algorithm

PSO parameters	
w_{max} : Initial weight	0.9
w_{min}: Final weight	0.75
c_1	0.7
c_2	1.47
Population size	10
Iteration Max	10

Table 6. Settings of the SA algorithm

SA parameters	
Initial temperature	100
Final temperature	0.001
Cooling Coefficient	0.8
Number of iterations	10

Moreover, Table 7 shows the obtained PSSs parameters by GA, PSO and SA optimization.

Table 7. Optimization results by the different techniques

		GA	*PSO*	*SA*
PSS1	K_{C1}	***45.8824***	***35.4298***	***47.3142***
	T_{11}	***0.2772***	***0.1680***	***0.2101***
	T_{21}	***0.0277***	***0.0047***	***0.0246***
PSS2	K_{C2}	***14.5098***	***35.4298***	***49.4164***
	T_{12}	***0.2618***	***0.1680***	***0.2431***
	T_{22}	***0.0269***	***0.0047***	***0.0264***
	$F_{OBJ}\,\%$	***25,20***	***17,10***	***15,90***
	Search space	$0.001 \leq K_{cj} \leq 50$ $0.001s \leq T_{1j} \leq 2s$ $0.001s \leq T_{2j} \leq 2s$		

Lire phonétiquement

Dictionnaire - Afficher le dictionnaire
pronom
them
article
the

4.1. Dynamic response of the stabilzed system

To evaluate the effectiveness and compare the different techniques applied to optimize the PSSs, a dynamic study using is carried out under different load conditions and a significant disturbance with the same scenario already mentioned. The simulation results for all the operating conditions are shown in Figure 7, Figure 8 and Figure 9.

The three meta-heuristic techniques can be compared, as shown in the Table 8 and Table 9. The comparison of these different techniques was made by taking into account two criteria, the amplitude of the first peak and the attenuation time (see Table 10). We can see that the meta-heuristic techniques used to the mitigation electromechanical oscillations, by optimizing the parameters of PSSs, have proved their effectiveness in general.

Table 8. Comparison between the three techniques (Point of view first peak amplitude)

	Load		
Generator	Light	Nominal	Heavy
G1	*SA* 0.0017 *p.u*	*GA* 0.0031 *p.u*	*GA* 0.0032 *p.u*
G2	*GA/PSO* 0.0049 *p.u*	*GA* 0.0049 *p.u*	*PSO* 0.0049 *p.u*
G3	*SA* 0.0011 *p.u*	*SA* 0.0051 *p.u*	*PSO* 0.0054 *p.u*

Table 9. Comparison between the three techniques (Point of view of time attenuation)

	Load		
Generator	Light	Nominal	Heavy
G1	*SA* 3.75s	*GA/PSO/SA* 4.84s	*PSO* 4.19s
G2	*SA* 4.33s	*GA/PSO/SA* 3.26s	*PSO* 3.81s
G3	*PSO/SA* 4.68s	*GA/PSO/SA* 5.03	*PSO* 3.70s

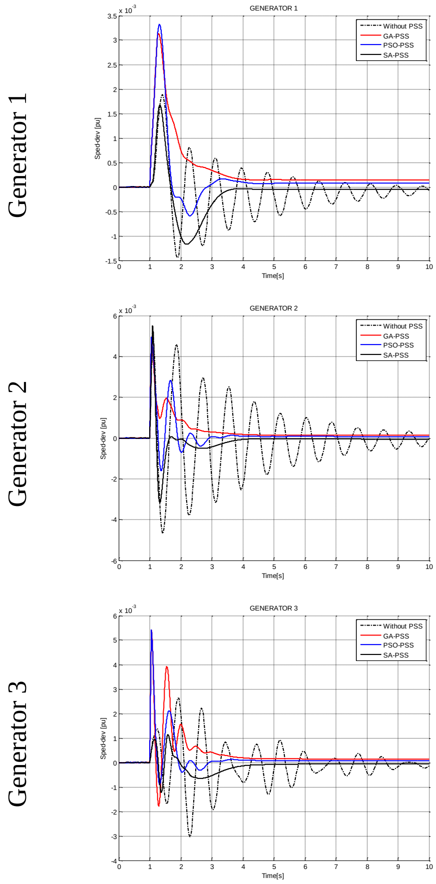

Figure 7. Speed deviations for light load condition.

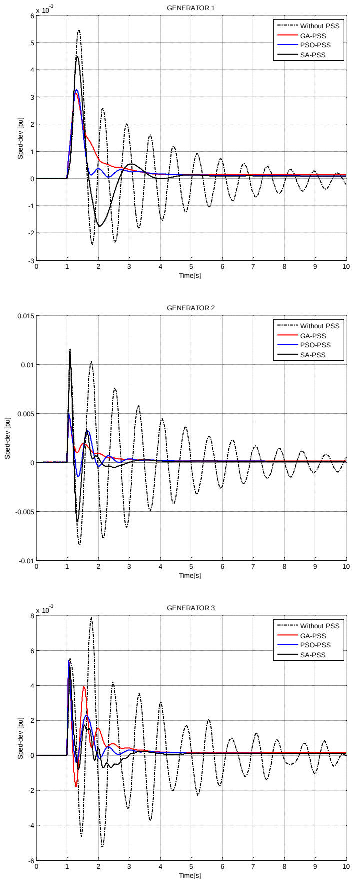

Figure 8. Speed deviations for nominal load condition

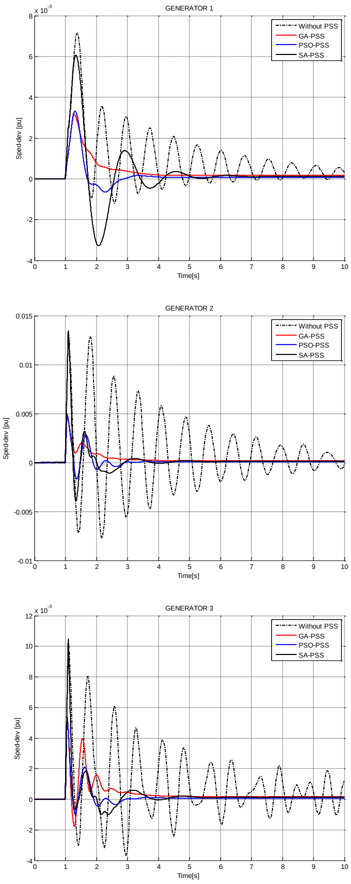

Figure 9. Speed deviations for heavy load condition

Table 10. The best technique in both cases

Generator	Load		
	Light	Nominal	Heavy
G1	*SA*	*GA*	*GA/PSO*
G2	*SA*	*GA*	*PSO*
G3	*SA*	*SA*	*PSO*

To compare accurately the three techniques in terms of execution time, the population size (characterizing the GA and PSO only) and the number of generations that characterizes the three methods, have been varied.

One can see from Figure 10 the difference in terms of execution time between GA and PSO when we vary the number of generations for different population size of 64 and 128 individuals. It is clear that PSO takes less time to converge compared to GA. Note that the mean calculation time of each technique is the average value of several executions for the same generation.

The results are carried out using a T 6600 (2 duo CPU, 2.2 GHz) PC and the Matlab R2010a software.

To compare SA with GA and PSO in terms of execution time, it should be noted that the simulated annealing depends on several parameter settings that increase considerably the convergence time (such as the cooling coefficient, the maximum number of temperature and the difference between the initial and the final temperature). In this comparison study, the same parameters of

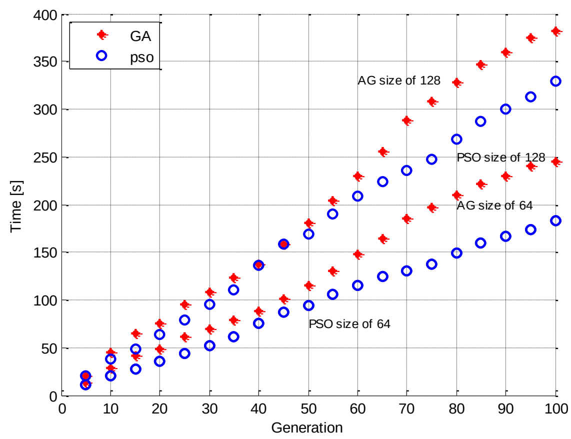

Figure 10. Comparison between GA and PSO in terms of mean calculation time for two different sizes of population.

Table **6** have been considered. But the number of generations has been varied for two different values of the cooling coefficient (0.95 and 0.8).

From Figure 11, one can note a great difference in terms of execution time in favor of the simulated annealing for different cooling coefficient values (0.8 and 0.95 in this case). One can conclude also that the PSO technique is more rapid then the GA.

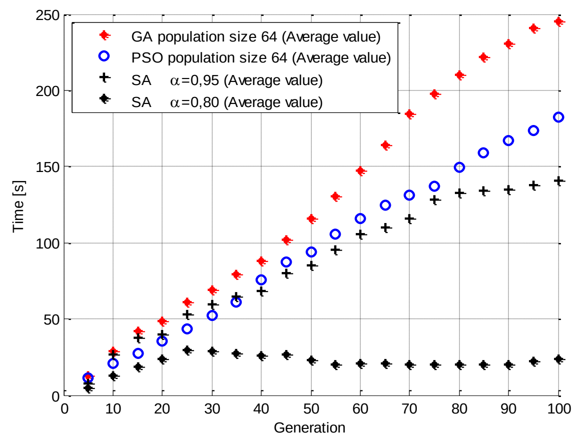

Figure 11. Comparison of three techniques (SA, GA and PSO) in terms of execution mean time.

5. CONCLUSION

In this study, three techniques has been presented and applied to optimal design of PSSs for multimachine power systems. In fact, the optimal parameters of the PSSs are globally tuned by GA, PSO and SA algorithms. Simulation results of the system dynamic response for different operating points has shown the effectiveness and the robustness of these algorithms in terms of damping characteristics and dynamic stability of the power system.

A comparison of the three techniques applied to the PSS optimal design leads to the following remarks:

There is no an absolutely better algorithm in terms of PSSs performance, since this depends on several parameters. In terms of time convergence, PSO has the best score. Elsewhere, SA algorithm is simpler to implant. But, the genetic algorithm usually gives the best result in terms of the objective function value, since its convergence is sure but asymptotic. It was found also that the results of these algorithms are more or less good in terms of the first peak limitation and the time of rotor oscillation damping. It depends on the system operating point and the PSS placement. The major drawback of meta-heuristic techniques is that their settings are adapted empirically to the problem for each operating point, and the power system changes its configuration continuously.

REFERENCES

1. **Wenxin Liu, Ganesh K. Venayagamoorthy, Donald C wunsch II.** Adaptive Neural Network Based Power System Stabilizer. *USA: IEEE.* 2003, 7803-7898-9/03.
2. **Larsen, E.V and Swann, DA.** Applying power system stabilizer. *IEEE Transaction on Power Apparatus and Systems.* s.l, 1981. 6, pp.3017-3041.
3. **Jenica Ileana Corcau, Eleonor Stoenescu.** Fuzzy logic controller as a power system stabilizer Systems and signal preceding. *International Journal of circuits.* 3, 2007, Vol. I.
4. **Manisha Dubey, Pankaj Gupta.** Design of Genetic-Algorithm Based Robust Power System Stabilizer. *International Journal of computational Intelligence.* 1, 2006, Vol. II.
5. **Zhu Jizhong.** Optimization of power system operation. USA: IEEE Press Editorial Board WILEY, 2009. 978-0-470-29888-6.
6. **Dhubkarya D.C., Deepak Nagariya, Jay Kumar,** Function Optimization Using Genetic Algorithm by VHDL. *Global Journal of Computer Science and Technology.* Issue 9 Ver. 1.0, 2010, Vol. 10.
7. **Melanie Mitchell.** *An Introduction to Genetic Algorithms.* Cambridge, Massachusetts London, England: A Bradford Book The MIT Press, 1999. 0−262−13316−4 (HB), 0−262−63185−7 (PB)
8. **Anthony Carlisle, Gerry Dozier.** Tracking Changing Extrema with Adaptive Particle Swarm Optimizer. *WAC Proceedings.* Orlando, Florida: http://wacong.com, June 2002.
9. **F. Schutte,** The Particle Swarm Optimization Algorithm. *Structural Optimization.* 2005. EGM 6365.
10. **COOREN.Yann.** *Perfectionnement d'un algorithme adaptatif d'Optimisation par Essaim Particulaire Applications en génie médical et en électronique.* L'université Paris 12 Val De Marne : Thèse Doctorat, 2008
11. **Sonmez, A H Ertas and F O.** Design optimization of composite structures for maximum strength using direct simulated annealing. *Journal of Mechanical Engineering Science.* Proceedings of the Institution of Mechanical Engineers, 2011, Vol. C, 10.1243/09544062JMES2105.
12. **Chibante. Rui.** *Simulated Annealing Theory with Applications.* India: Sciyo, September 2010. 978-953-307-134-3.
13. **Alireza Bloori Araban,** Applying Simulated Annealing Algorithm for Cross-Docking Scheduling. 2009, Vol. II.
14. **H. Shayeghi, A. Safari, H.A. Shayanfar.** Multimachine Power System Stabilizers Design Using PSO Algorithm. *International Journal of Electrical Power and Energy Systems.* Iran: s.n., 2008. Vol. 1, 4.
15. **A. Jeevanandham,** Optimization of Power System Stabilizers Relying on Particle Swarm Optimizers. *University (GCT Campus): ICGT-ACSE Journal.* 2008. Vol. 8, II. 1687-4811.
16. **Ermanu A. Hakim, Adi Soeprijanto, and Mauridhi H.P.** Fuzzy PID based PSS Design Using Genetic Algorithm. *International Journal of Electrical Power and Energy Systems Engineering.* Indonesia: s.n., 2010. Vol. 3, 1.
17. **Rusejla. S.** Single Machine Infinite Bus System. *Rapport Technique.* Zurich : KTH, 2003.
18. **M. Mekhanet, L. Mokrani, A. Choucha.** Stabilisateurs de puissance (PSSs) intelligents génétiques et neuro-flou-génétiques. Revue DIRASSAT *2nd International Conference on Electrical and Electronics Engineering.* LAGHOUAT/ ALGERIE: Revue DIRASSAT, 21-23 April 2008. 1112-4652.
19. **ANDREOIU.** On Power System Stabilizers: Genetic Algorithm Based Tuning and Economic. *THESIS FOR THE DEGREE OF DOCTOR OF PHILOSOPHY.* Göteborg, Sweden: Department of Electric Power Engineering Chalmers, 2004.
20. **M.A. Abido,** Ahybrid Neuro-fuzzy Power System stabilizer For Multimachine Power System. *IEEE Transaction on Power Systems.* 1998, Vol. Vol 13, 4.
21. **A.L.B. do Bomfim, G.N. Taranto and D.M. Falcão.** Simultaneous Tuning of Power System Damping Controllers Using Genetic Algorithms. *IEEE Transactions on Power Systems.* 1, February 2000, Vol. 15, pp. 163-169.
22. **Guillaume. Cala.** Optimisation par essaim de particules. France. *EPITA.* SCIA, 2009.
23. **Lahdeb. Mohamed.***Théorie et Applications de Méthodes d'hybridations Métaheuristiques dans les Réseaux Electriques .* LAGHOUAT, ALGERIE : UNIVERSITE AMAR TELIDJI., 2007.
24. **Antoine Dutot, Damien Olivier.** Optimisation par essaim de particules Application au problème des n-Reines. *http://www-lih.univ-lehavre.fr/~olivier/enseignement/Maitrise/TD/OptimisationParEssaim.pdf.* (Online] (Cited: 23 07 2011.]
25. **Maurice, .***Particle Swarm Optimization.* France: British Library Cataloguing-in-Publication Data, 2006. 13: 978-1-905209-04-0.
26. **Abido, M. A.** Particle Swarm Optimization for Multimachine Power System Stabilizer Design. *IEEE Power Engineering Society Summer Meeting.* Vancouver, Canada, July 15-19, 2001, pp. 1346-1351.

27. **EGLESE,** Simulated Annealing: A tool for Operational Research. *European Journal of Operational Research 46 (1990) 271-281*. North-Holland, 1990, 271-281.
28. **Abdelmadjid, TOLBA M.** Commande Des Systèmes D’energie De Puissance Par Des Approches Heuristiques Modernes. *Thèse de Magister.* El-Harrach, Alger : Ecole Nationale Polytechnique, 2005.

Med. MEKHANET
L. MOKRANI
Med. LAHDEB

Electrical Engineering Department
AMAR TELIDJI University of LAGHOUAT
ALGERIA
E-mails: m.mekhanet@mail.lagh-univ.dz,
l.mokrani@mail.lagh-univ.dz,
m.lahdeb@mail.lagh-univ.dz