Selected Papers from: Communications & Information Technology 2008, Circuits, Systems and Signals 2008, Applied Mathematics, Simulation, Modelling 2008, Marathon Beach, Attica, Greece, June 1-3, 2008

Modified Evolutionary Particle Swarm Optimization for AVR-PID tuning

M. L. Amer Hanafy H. H. Hassan Hosam K. M. Youssef

Electrical Power and Machines Department
Faculty of Engineering, Cairo University, Giza, Egypt

Abstract— **this paper, introduces to the power systems society Modified Evolutionary Particle Swarm Optimization (MEPSO), as a tuning algorithm for AVR-PID controller for synchronous generator excitation system, for nominal system parameters and step reference voltage input.**

MEPSO is a hybrid in concepts of Genetic Algorithms (GA) and Particle Swarm Optimization (PSO), the algorithm is adopted to solve the target problem. Simulation is applied on synchronous generator's excitation system model, and a comparison between excitation system performances using the hybrid algorithm (MEPSO) and both of its parent algorithms (GA and PSO) is introduced, to prove that the MEPSO is a competitive algorithm and provides better performance if implemented to tune the AVR-PID controller for synchronous generator excitation system.

Index Terms—**Generator Excitation System, AVR, PID Controller, Genetic Algorithm, Particle Swarm Optimization, Modified Evolutionary Particle Swarm Optimization,**

I. List of Symbols

K_A	Amplifier gain
τ_A	Amplifier time constant
K_E	Exciter gain
τ_E	Exciter time constant
K_G	Generator gain
τ_G	Generator time constant
K_S	Sensor gain
τ_S	Sensor time constant
pbest	personal best position
gbest	global best position
W_{i1}	Weight of the inertia term
W_{i2}	Weight of the memory term
W_{i3}	Weight of the information exchange term
W_{i4}	Weight affecting dispersion around pbest
GP	Gaussian Probability distribution
SH	Shooting factor
t_r	rise time
t_{st}	settling time
SSE	Steady State Error
$e_{max\ V}$	Error at maximum output value
O_{sh}	Maximum percent overshoot

Eng. Moataz Lotfy Amer (e-mail: mlamer@pgesco.com).
Dr. Hanafy H. H. Hassan (e-mail: hanafy_hassan@hotmail.com).
Prof. Dr. Hosam Kamal .M. Youssef (e-mail: hosamkm@yahoo.com).

II. Introduction

Both Genetic Algorithm (GA) and Particle Swarm Optimization (PSO) have been used for AVR-PID tuning in several literatures [1]-[4], but this paper will introduce a hybrid algorithm to combine the advantages of GA and PSO for AVR-PID tuning.

Hybridization of GA and PSO have been done in different ways and for different problems [5],[6], the most famous and competitive hybrid algorithm called Evolutionary Particle Swarm Optimization (EPSO) [7], however it was not implemented to tune AVR-PID controllers for synchronous generator excitation system yet.

This paper will modify EPSO to be adequate and competitive algorithm for AVR_PID tuning, the modified EPSO will be called as MEPSO. The differences between MEPSO and EPSO are also introduced.

III. Evolutionary Particle Swarm Optimization

Evolutionary Particle Swarm Optimization (EPSO) is a hybrid in concepts of Genetic Algorithms (GA) and Particle Swarm Optimization (PSO). EPSO first proposed in [8] and with applications in Power Systems in [9], [10].

EPSO is an Evolutionary Algorithm (close to the family of Evolution Strategies and Evolutionary Programming) where the mutation is only applied to strategic parameters (the parameters that condition the evolution of a given solution) and the recombination is non-conventional (it is, in fact, the movement rule of Particle Swarm Optimization methods).

Recombination is an operation that produces new offspring from some form of combination of parent individuals, chosen in the population.

- Recombination operation in GA is called "**crossover**".
- Recombination operation in PSO is called "**movement rule**" in which a new individual was generated as a weighted combination of parents which are the best ancestor of this individual "pbest" and the best ancestor of the present generation "gbest".

In these types of recombination in GA and PSO, a new individual is formed from a weighted mix of ancestors, and this weighted mix may vary in each space dimension [10].

The recombination rule for EPSO is the following:

Given a particle X_i^K, a new particle X_i^{K+1}, results from

$X_i^{K+1} = X_i^K + V_i^{K+1}$ ……….. (1)

$V_i^{k+1} = W_{i1}^* V_i^k + W_{i2}^* (pbest - X_i^K) + \mathbf{C}\, W_{i3}^* (gbest^* - X_i^K)$……………………….... (2)

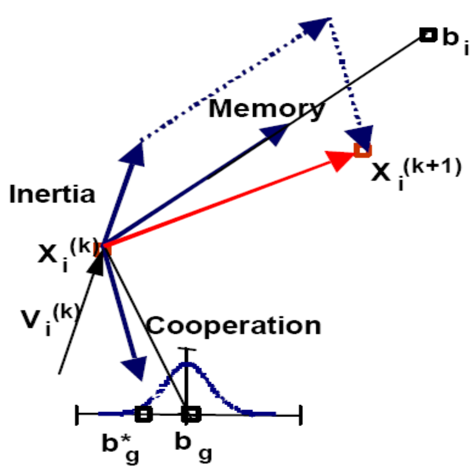

Fig. 1: Illustration of EPSO particle reproduction (a particle $\mathbf{X}_i$ generates an offspring at a location commanded by the movement rule). [8]

Where:
X_i^k: current position,

X_i^{k+1}: modified position,

V_i^k: current velocity,

V_i^{k+1}: modified velocity,

gbest : swarm pest position,
pbest: particle pest position.

Symbol * indicates that these parameters will be subject to a mutation process

W_{i1}: weight of the inertia term (the new particle is created in the same direction as its previous couple of ancestors)

W_{i2}: weight of the memory term (the new particle is attracted to the best position occupied by its ancestors)

W_{i3}: weight of the cooperation or information exchange term (the new particle is attracted to the overall best position found by the swarm).

W_{i4}: weight affecting dispersion around the best-so-far

C: a diagonal matrix with each element in the main diagonal being a binary variable equal to 1 with a given communication probability ρ, and 0 with probability (1- ρ); in basic models, ρ = 1 but in advanced models ρ = 0.2 has proven to be more effective in assuring the progress of the algorithm, by limiting communication among the particles of the swarm – yet another means of shaping the recombination [10].

The most popular mutation rules for the strategic parameters are the following:

$W_{ik}^* = W_{ik}\, [\log N(0,1)]^{\tau}$ ………………(3)

Where: logN(0,1) is a random variable with lognormal distribution derived from the gaussian distribution N(0,1) of 0 mean "μ=0" and 1 variance "σ2=1" , (See Fig. 2,and Fig. 3) ; "τ" is a learning parameter, it is fixed externally, and controlling the amplitude of the mutations – smaller values of "τ" leads to higher probability of having values close to 1.

Approximations to this scheme are sometimes used by some researchers, such as (4)

$W_{ik}^* = W_{ik}\, [1+ \tau\, N\, (0, 1)]$ …………..... (4)

And they are equivalent provided that "τ" is small and the outcome is controlled so that negative weights are ruled out. This scheme is preferable to additive mutations like (5)

$W_{ik}^{*} = W_{ik} + \tau N(0,1)$ ……………….. (5)

In this case the absolute value of the mutation is insensitive to the value of "W".

As for the global best"$\mathbf{b_g}$", it is randomly disturbed to give (6)

$gbest^{*} = gbest + W_{i4}^{*} N(0,1)$ …………...(6)

Where: W_{i4}, is the forth strategic parameter associated with particle i. It controls the "size" of the neighborhood of gbest where it is more likely to find the real global best solution (assumed not found so far during the process) or, at least, a solution that may be better than the current"$\mathbf{b_g}$".

This weight W_{i4} is mutated (signaled by *) according to the general mutation rule of strategic parameters, allowing the search to focus on a given point, if convenient.

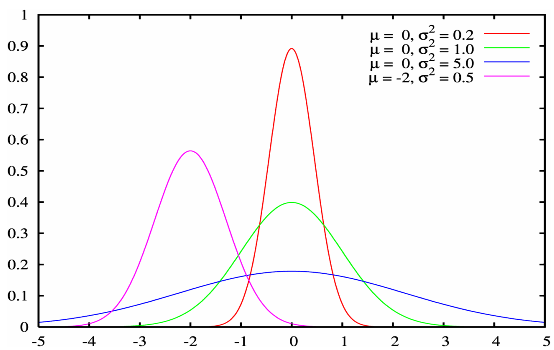

Fig. 2: Gaussian probability distribution curves

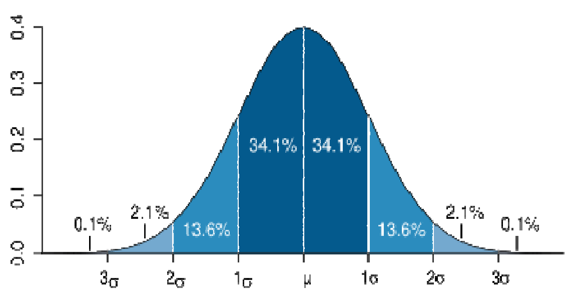

Fig. 3: Illustration of Mean and Variance for Gaussian probability distribution curve

IV. Modified Evolutionary Particle Swarm Optimization

Modifications has been applied to EPSO to be more efficient and more competitive to its parent algorithms (PSO and GA), for solving AVR- PID controller tuning problem, this modifications basically impact both velocity update equation, and the mutation of strategic parameters.

The modified algorithm will be called as Modified Evolutionary Particle Swarm Optimization (MEPSO). The general scheme of MEPSO is the following:

The recombination rule of MEPSO is the following:

Given a particle X_i^K, a new particle X_i^{K+1}, results from

$X_i^{k+1} = X_i^k + V_i^{k+1}$ …………………….. (7)

$$V_i^{k+1} = \underbrace{W_{i1} * V_i^k}_{\text{Current motion influence}} + \underbrace{W_{i2} * \mathbf{\textit{rand}}\,[\mathbf{pbest} - X_i^k]}_{\text{Particle memory influence}} + \underbrace{W_{i3} * \mathbf{\textit{rand}}\,[\mathbf{gbest^{*}} - X_i^k]}_{\text{Swarm influence}}$$

…………………….. (8)

(7) and (8) represents the movement rule in MEPSO; it is most like similar to position and velocity update equations of PSO as illustrated in Fig. 4. In fact, it is a form of a recombination called intermediary recombination, where the value of any variable in the offspring receives a contribution from all parents.

The position of each particle is updated using its velocity vector and its current position as shown in (7) and (8) and depicted in Fig. 4.

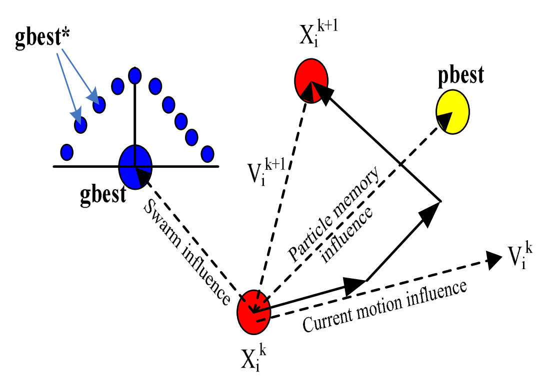

Fig. 4: Illustration of MEPSO particle reproduction: A particle Xi generates an offspring at a location commanded by the movement rule.

The velocity update formula includes random parameters, represented by the uniformly distributed variables, "***rand***", to ensure good coverage of the design space, and to avoid entrapment in local optima. The three values that affect the new search direction, namely, current motion, particle own memory, and swarm influence, are incorporated via a summation approach as shown in (8), with four weighting factors (Strategic Parameters), namely, Weight of the inertia term (w_{i1}), weight of the memory term (w_{i2}), weight of the cooperation or information exchange term (w_{i3}), and weight affecting dispersion around the best (w_{i4}).

Mutation rules are different than EPSO; however they are also applied to strategic parameters before implementation for velocity update equation as follows:

Weight of the inertia term:

$W_{i1}* = W_{i1} + K*GP$ ……………………….. (9)

Where: "GP" Gaussian Probability distribution

Weight of the memory term:

$W_{i2}* = W_{i2}(1+SH*GP)$…………………... (10)

Where: "SH" Shooting factor (either 0 or 1, based on shooting probability) .SH is inspired by the fact that the Gaussian Probability Distribution may not be representative for actual mutation process (i.e. practical mutation is not straight forward and follows GP distribution curve (See Fig. 2, 3)).

For AVR_PID controller a typical value of shooting probability is 30%, however it may take any other value based on the problem in hand. 30% shooting probability means that the parameter SH will be "1" for 30% of iterations (Generations), in other words not all particles memories are following GP curve however some of it do.

Weight of the cooperation or information exchange term:

$W_{i3}* = W_{i3}(1+SH*GP)$…………………... (11)

Weight affecting dispersion around the pbest (See Fig. 4):

$gbest^* = gbest \times W_{i4}^*$……………………..(12)

$W_{i4}* = W_{i4} + SH*GP$……………………... (13)

After recombination and position update, each generation has its fitness evaluated according to Objective Function. The objective function used with MEPSO is detailed in simulation settings (Section VI).

After fitness evaluation of each particle in the population, selection process starts. Selection is the process of choosing the next population from the current population. The standard selection technique in Modified Evolutionary Particle Swarm Optimization is stochastic sampling with replacement (roulette wheel). This technique is by no means the most popular method employed in evolutionary algorithms; it is simply the most commonly employed.

Roulette Wheel Selection is known as fitness proportionate, it operates on the concept that the proportionate fitness of each particle should be reflected on that particle's incidence in the mating pool.

Thus, each particle in the swarm has a probability of selection for recombination based on its relative fitness. Roulette Wheel provides the greatest probability of selection to the fittest members of the population.

<u>Summary of Roulette Wheel Selection:</u>

I. Sum the fitness of each member of the population.
II. Determine the relative fitness of each member of the population.
III. Generate a random number (SPIN) between zero and some predefined maximum value (MAX).
IV. Select next individual.
V. From SPIN, subtract the individual's relative proportion of MAX (i.e., relative fitness times MAX).
VI. Repeat steps IV and V until SPIN is less than or equal to zero.
VII. Repeat steps III to VI until mating pool is full.

After selection process complete, repeat velocity and position update and so on, until termination condition is achieved.

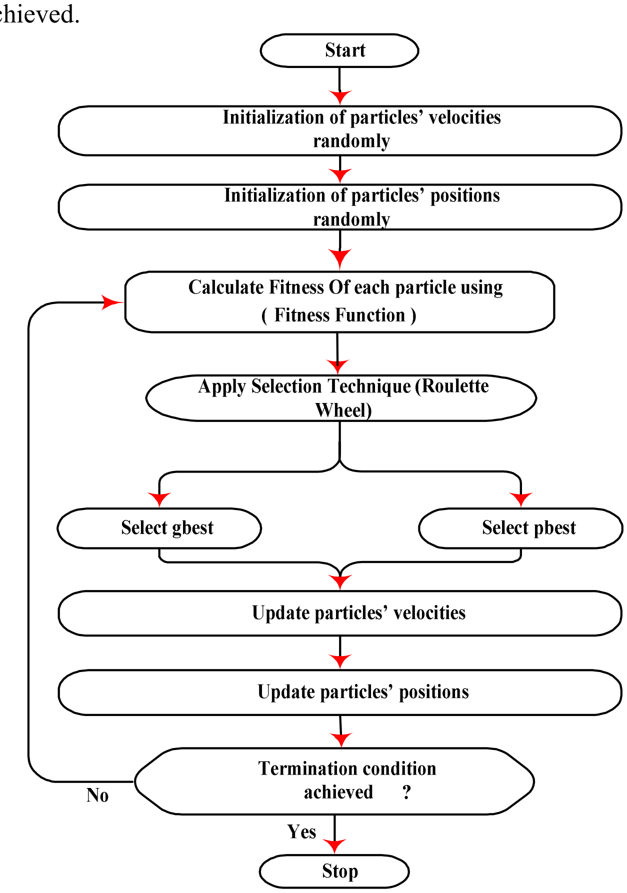

Fig. 5: MEPSO flow-chart.

EPSO and MEPSO are self-adaptive algorithms because the weights that regulate recombination are taken as strategic parameters and are mutated and allowed to evolve. Mutation acts on the recombination weights and, from generation to generation, a better (adaptive) recombination operation evolves [11].

V. MEPSO, POS, AND GA FOR AVR-PID TUNING

There is a solid theoretical background giving insight on why GA achieve convergence and how a near optimal progress rate is achieved [12]. In GA, the generation of offspring is regulated by operations of mutation and crossover. However, these reproduction mechanisms do not provide a positive push towards the optimum – this is the role of the selection.

On the other hand, in Classical PSO there is a reproduction scheme but selection is trivial – each parent has one child and each of these survives to its parent. However, the movement (reproduction) rule, by itself, assures the progress to the optimum, meaning that, on average, each generation will be better than the preceding one.

In MEPSO we have two mechanisms acting in sequence, each one with its own probability of producing not only better individuals, but also an average better group. Selection acts on a generation that is already on average better than the preceding, so the effects are additive.

The fact that MEPSO is self-adaptive adds another interest to the method: it avoids in a large scale the need for fine tuning the parameters of the algorithm, because the procedure will hopefully learn (in the evolutionary sense) the characteristics of the search space, and will self-tune the weights in order to produce an adequate rate of progress towards the optimum.

In AVR-PID controller tuning for synchronous generator excitation system, MEPSO has been showing better performance than other meta-heuristics such as Genetic Algorithms or the classical Particle Swarm Optimization algorithm [13], [14]. It tends to escape from local optima and more robust, i.e., generates results with a narrow variance in a series of runs for a problem with random initialization.

VI. SYNCHRONOUS GENERATOR EXCITATION SYSTEM MODEL

It is well known that a change in the real power demand affects essentially the frequency, whereas a change in the reactive power affects mainly the voltage magnitude. The sources of reactive power are generators, capacitors, and reactors. The generator reactive power is controlled by field excitation. Other supplementary methods of improving the voltage profile on electric transmission systems are transformer load-tap changers, switched capacitors, step-voltage regulators, and static VAR control equipment.

The generator excitation system maintains generator voltage and controls the reactive power flow. The generator excitation system may be provided through slip rings and brushes by means of DC generators mounted on the same shaft of the rotor of the synchronous machine, or through AC generators with rotating rectifiers, and are known as brushless excitation.

Recently Static Excitation System is increasingly used. Static rectifier, supplies the excitation current directly to the field of the main alternator through its slip rings. The supply of power to the rectifiers is from the main generator or the station auxiliary bus through a transformer to step down the voltage to an appropriate level.

The primary means of generator reactive power control is generator excitation control using automatic voltage regulator (AVR). The role of an (AVR) is to hold the terminal voltage magnitude of a synchronous generator at a specified level. The schematic diagram of a simplified AVR is shown in Fig.5.

A drop in the terminal voltage magnitude accompanies an increase in the reactive power load of the generator. The voltage magnitude is sensed through a potential transformer on one phase. This voltage is rectified and compared to a DC set point signal. The amplified error signal controls the exciter field and increases the exciter terminal voltage. Thus, the generator field current is increased, which results in an increase in the generated emf. The reactive power generation is increased to a new equilibrium, raising the terminal voltage to the desired value.

For comparison between Excitation systems performances with the three different algorithms, MEPSO, PSO, and GA, linearized model of excitation system has been used to simplify the comparison between the algorithms. We will look briefly at the linearized models of the component involved in the AVR system (See fig.6 and 7).

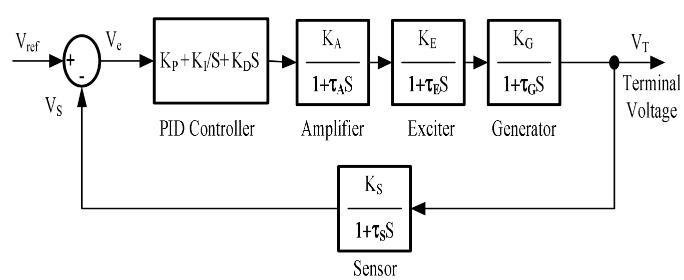

Fig. 6: Block diagram of AVR system with PID controller.

Amplifier Model

The excitation system amplifier may be a magnetic amplifier, rotating amplifier, or modern electronic amplifier. The amplifier is represented by a gain "K_A" and a time constant "τ_A", and the transfer function is

$$\frac{K_S}{1+\tau_S S}$$

Typical values of KA are in the range of 10 to 400. The amplifier time constant is very small, in the range of 0.02 to 0.1 second, and often is neglected.

Exciter Model

There is a variety of different excitation types. However, modern excitation systems uses ac power source through solid-state rectifiers such as SCR. The output voltage of the exciter is a nonlinear function of the field voltage because of the saturation effects in the magnetic circuit. Thus, there is no simple relationship between the terminal voltage and the field voltage of the exciter.

Many models with various degrees of sophistication have been developed and are available in the IEEE recommendation publications. A reasonable model of a modern exciter is a linearized model, which takes into account the major time constant and ignores the saturation or other nonlinearities. In the simplest form, the transfer function of a modern exciter may be represented by a single time constant "τ_E" and a gain "K_E", i.e.

$$\frac{K_E}{1+\tau_E S}$$

The time constant of modern exciters are very small.

Generator Model

The synchronous machine generated emf is a function of the machine magnetization curve, and its terminal voltage is dependent on the generator load. In the linearized model, the transfer function relating the generator terminal voltage to its field voltage can be represented by a gain "K_G" and a time constant "τ_G" and the transfer function is

$$\frac{K_G}{1+\tau_G S}$$

These constants are load-dependent, "K_G" may vary between 0.7 to 1, and "τ_G" between 1.0 and 2.0 seconds from full-load to no-load.

Sensor Model

The voltage is sensed through a potential transformer and, in one form, it is rectified through a bridge rectifier. The sensor is modeled by a simple first order transfer function, given by

$$\frac{K_A}{1+\tau_A S}$$

"τ_S" is very small, and we may assume a range of 0.01 to 0.06 second. Utilizing the above models the AVR block diagram is shown in Fig. 7.

VII. Simulation Settings

Simulation results are obtained by MATLAB 7.2 software on a 3.2 GHz, 500 MB of RAM, P4 computer.

AVR-PID controller based on MEPSO algorithm has been simulated on MATLAB-SIMULINK as shown in Fig. 8. The controller subjected to unit step function of the following parameters:

Step time=0 sec, Initial Value=0 per-unit, Final value=1 per unit, and Sample time=0 sec.

Augmented Deviation Objective Function (ADOF) is used as objective functions for MEPSO algorithm to evaluate the performance of AVR-PID controller.

ADOF= (sum (e) + t_r)*($e_{\max V}$+1)…. (14)

The right hand side is a multiplication of two terms, the first term (sum (e) + t_r) consists of sum (e) which is summation of output error from the point at which process output reaches 95% of it's reference value, and t_r which is the rise time or the time taken by output value to rise form 0.05% to 95% of steady state value in seconds. The second term is the error at maximum output value ($e_{\max V}$).

Block diagram parameters are chosen to be as follows [15]:

K_A = 10, K_E = K_S = 1.0,
τ_A = 0.1 sec, τ_E = 0.4 sec, τ_S = 0.01 sec
"K_G"and "τ_G "are variables to illustrate the performance of the controllers at different generator cases.

MEPSO Algorithm parameters are chosen to be as follows:
Maximum population size is applied to be 50, maximum iteration cycles = 20, Weighting factors used are same as mentioned in section "V"

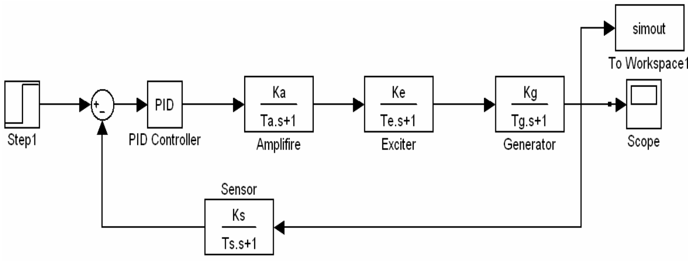

Fig. 7: MATLAB-SIMULINK based block diagram of AVR system with intelligent PID controller.

VIII. RESULTS AND DISCUSSION

Table 1 list representative sets of results for two separately conducted tests on the AVR system with GA, PSO, and MEPSO based PID controller.

Simulation results of excitation control system using AVR-PID controller at different sets of generator parameters and for 20 iterations for each algorithm (i.e. GA, PSO, and MEPSO).

Simulation results of excitation control system using AVR-PID controller at different sets of generator parameters are shown in table 1 and for 20 iterations, and for each of competitive algorithms which are GA, PSO, and MEPSO.

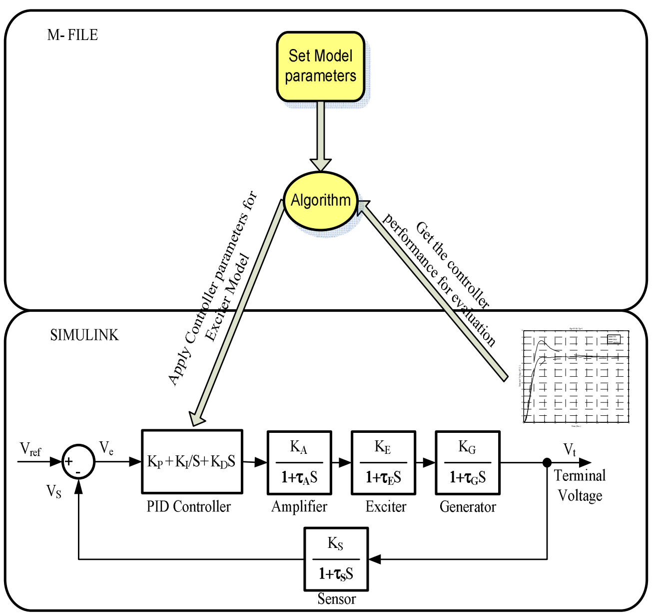

Fig. 8: The flow chart of simulation procedures

The table indicates at each set of generator gain and time constant and each algorithm of the competitive algorithms, the step response of the controller and its performance in terms of Steady State Error (SSE), Over Shoot, Rise Time, and Settling Time, which are the performance parameters required for comparison between controllers' performances .The table also indicates the PID controller gains at each case.

Fig. 8 through 13 represents graphical illustration of table 1, for simplifying comparison between controllers' performances

Simulation results indicate the precedence of MEPSO over its parent algorithms (GA and PSO), except figures 12 and 13 it was found that PSO and GA take the precedence respectively.

The reasons behind MEPSO failure to peat PSO and GA at these above mentioned cases is that it needs more iterations to be applied using the algorithm, and it is also required to take in considerations that the intelligent and optimization algorithms are based on random generations of initial solutions, that random generation may be so far or so close to the optimum solution, that is function of the processor random generator which is uncontrollable, however in almost all cases it is clear that Modified Particle Swarm Optimization is more reliable and provide better performance for Excitation Control System of Synchronous Machines.

The presented successful algorithm "MEPSO" can be modified to enhance the performance of PID controllers for many applications rather than the applied one.

Over than that the execution time for 20 iterations is 12 Seconds for MEPSO and PSO while it is 13 seconds for GA, which means that MEPSO gets the advantage of less execution time form PSO.

IX. CONCLUSION

Proposed Modified Evolutionary Particle Swarm Optimization (a hybrid algorithm based on Genetic Algorithm and Particle Swarm Optimization), proved its precedence on its parent algorithms to solve AVR-PID tuning problem for synchronous generator excitation system.

MEPSO is a step forward to enhance the performance of PID controllers' performance in general and specifically AVR-PID controllers for synchronous generators.

Table 1: Simulation results of excitation control system using AVR-PID controller at different sets of generator parameters and for 20 iterations for each algorithm

K_G	T_G	Algorithm	Controller Performance					Controller Parameters		
			e max V	O_{sh}	SSE	t_r	t_{st}	K_P	K_I	K_D
0.4	1	GA	0.0063	0.0063	0	0.6795	0.8553	1.6332	0.5444	0.3976
		PSO	0.1441	0.1441	0.044	0.3737	1.187	4.3225	0.083	0.6498
		MEPSO	0.0085	0.0085	0	0.571	0.7081	2.1438	0.6115	0.4989
0.4	1.5	GA	0.0326	0.0326	0.005	0.7413	0.91	1.9003	0.4623	0.427
		PSO	0.0217	0.0217	0.05	0.5025	0.9	3.509	0.1503	0.8201
		MEPSO	0.0279	0.0279	0.0066	0.7414	0.91	1.8765	0.4466	0.4248
0.4	2	GA	0.007	0.007	0.003	0.9803	1.289	1.9143	0.4279	0.5604
		PSO	0.0803	0	0.08	1.005	1.342	2.6802	0.0031	0.9298
		MEPSO	0.0164	0.0164	0.001	0.5426	0.6986	4.1625	0.6185	1.0257
0.5	1	GA	0.2477	0.2477	0	0.5439	2.656	1.9003	1.8615	0.3957
		PSO	0.0226	0	0.03	0.8764	1.164	3.2277	0.227	0.9582
		MEPSO	0.000171	0.000171	0	0.5705	0.7754	1.8086	0.4862	0.442
0.5	1.5	GA	0.0653	0.0653	0.002	0.6789	1.323	1.9003	0.4623	0.3976
		PSO	0.03	0	0.03	0.5727	1.261	3.0312	0.1744	0.8015
		MEPSO	0.0417	0.0417	0.001	0.5407	1.006	2.6791	0.5371	0.5995
0.5	2	GA	0.0805	0.0805	0	0.7768	1.682	1.9003	0.4623	0.3976
		PSO	0.0623	0	0.063	0.7728	0.8776	2.2494	0.0504	0.6657
		MEPSO	0.002	0	0.002	0.936	1.449	2.1542	0.339	0.6882
0.6	1	GA	0.0157	0.0157	0.002	0.471	1.205	1.8068	0.4279	0.3976
		PSO	0.0899	0	0.03	1.2209	3.85	2.3208	1.0033	1.0266
		MEPSO	0.0172	0.0172	0.004	0.6079	0.76	1.2456	0.3613	0.2776
0.6	1.5	GA	0.1119	0.1119	0	0.5724	1.247	1.9003	0.4623	0.354
		PSO	0.0333	0.0333	0	0.3954	1.133	3.5238	0.4881	0.7623
		MEPSO	0.0225	0.0225	0	0.542	0.9059	2.1769	0.425	0.5124
0.6	2	GA	0.0746	0.0746	0.009	0.7148	1.626	1.7293	0.4623	0.3976
		PSO	0.0461	0.0461	0.007	0.5393	1.037	2.9946	0.5442	0.697
		MEPSO	0.003	0.003	0.001	0.648	0.8816	2.1033	0.3567	0.5656
0.7	1	GA	0.0381	0.0381	0	0.4	1.106	1.9075	0.4538	0.3976
		PSO	0.0116	0.0116	0.004	0.7383	0.89	2.9962	0.5505	0.6871
		MEPSO	0.0087	0.0087	0.002	0.4718	0.65	1.4962	0.3797	0.34
0.7	1.5	GA	0.063	0.063	0.005	0.5108	1.081	1.9592	0.4623	0.427
		PSO	0.0078	0.0078	0	0.4007	1.097	2.9674	0.417	0.6778
		MEPSO	0.0091	0.0091	0	0.509	0.645	2.0567	0.3397	0.4886
0.7	2	GA	0.1112	0.1112	0.007	0.6389	1.381	1.9767	0.4623	0.3976
		PSO	0.059	0.059	0.007	0.5047	1.001	2.847	0.5094	0.6373
		MEPSO	0.0105	0.0105	0	0.636	0.785	2.0158	0.3148	0.514
0.8	1.5	GA	7.4E-05	0	7.4E-05	1.4019	1.978	1.9003	0.2082	0.8373
		PSO	0.018	0.018	0.007	0.3713	0.9938	2.9381	0.5154	0.6628
		MEPSO	0.0259	0	0.0259	1.253	1.715	2.7391	0.0988	1.1299
0.8	2	GA	0.0029	0	0.002	0.7729	1.027	1.3926	0.2514	0.3976
		PSO	0.0571	0.0571	0.007	0.4323	0.7901	3.1932	0.505	0.6974
		MEPSO	0.0304	0.0304	0.002	0.3294	0.9866	4.8008	0.3813	1.0239

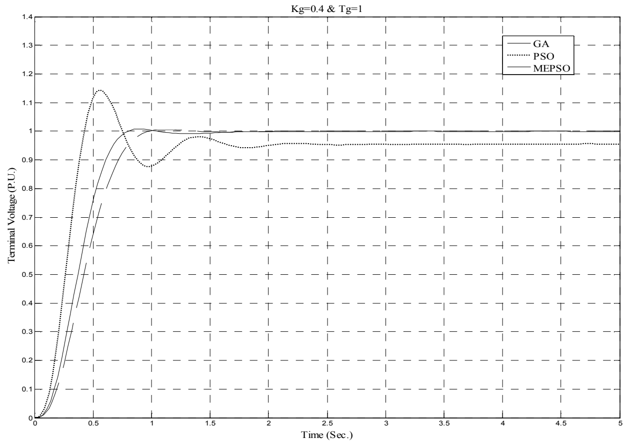

Fig. 8: Step response of excitation system controller using AVR-PID, for comparison between controllers' performances that are based on GA, PSO, and MEPSO, for 20 iterations at generator gain of 0.4 and time constant of 1 Second.

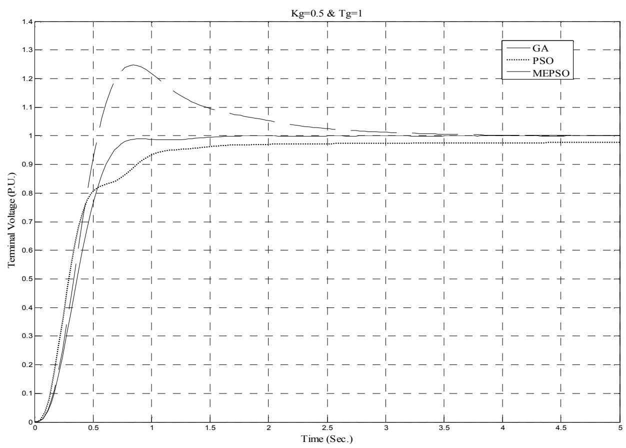

Fig. 9: Step response of excitation system controller using AVR-PID, for comparison between controllers' performances that are based on GA, PSO, and MEPSO, for 20 iterations at generator gain of 0.5 and time constant of 1 Second.

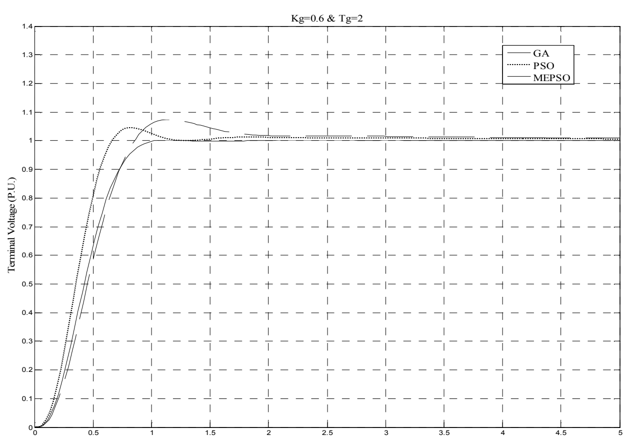

Fig. 10: Step response of excitation system controller using AVR-PID, for comparison between controllers' performances that are based on GA, PSO, and MEPSO, for 20 iterations at generator gain of 0.6 and time constant of 2 Second.

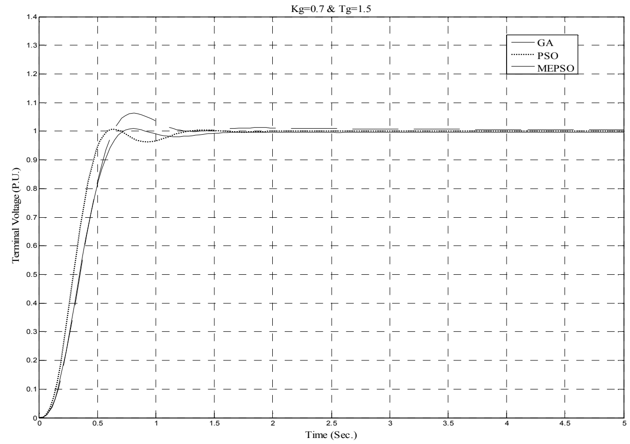

Fig. 11: Step response of excitation system controller using AVR-PID, for comparison between controllers' performances that are based on GA, PSO, and MEPSO, for 20 iterations at generator gain of 0.7 and time constant of 1.5 Second.

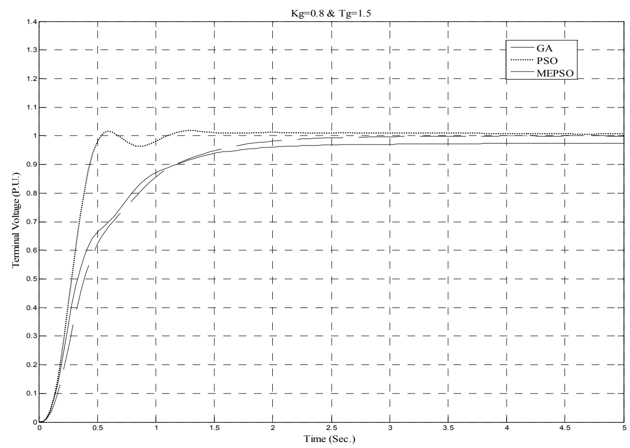

Fig. 12: Step response of excitation system controller using AVR-PID, for comparison between controllers' performances that are based on GA, PSO, and MEPSO, for 20 iterations at generator gain of 0.8 and time constant of 1.5 Second.

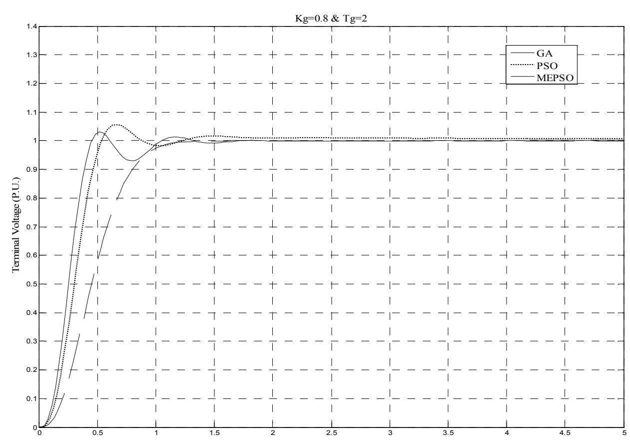

Fig. 13: Step response of excitation system controller using AVR-PID, for comparison between controllers' performances that are based on GA, PSO, and MEPSO, for 20 iterations at generator gain of 0.8 and time constant of 2 Second.

X. REFERENCES

[1] Viveros, E.R.C.; Taranto, G.N.; Falcao, D.M." Coordinated tuning of AVRs and PSSs by multiobjective genetic algorithms"Intelligent Systems Application to Power Systems, 2005. Proceedings of the 13th International Conference on Volume , Issue , 6-10 Nov. 2005 Page(s): 6 pp.

[2] V. Mukherjee, S.P. Ghoshal, Intelligent particle swarm optimized fuzzy PID controller for AVR system, Electric Power System Research (2007), doi:10.1016/j.epsr. 2006.12.04.

[3] S.P. Ghoshal, Optimizations of PID gains by particle swarm optimization in fuzzy based automatic generation control, Electr. Power Syst. Res. 72 (2004) 203–212.

[4] Z. L. Gaing, "A particle swarm optimization approach for optimum design of PID controller in AVR system," IEEE Trans. Energy Conversion, vol. 19, no. 2, pp. 384-391, June 2004.

[5] Lovberg, M. et al. (2001), "Hybrid Particle Swarm Optimizer with Breeding and Sub-populations", Proceedings of GECCO2001 - Genetic and Evolutionary Computation Conference 2001(pp. 409), July 7-11 2001, San Francisco CA, USA

[6] Mirko Todorovski and Dragoslav Rajiˇcic´,"An Initialization Procedure in Solving Optimal Power Flow by Genetic Algorithm" IEEE Trans. Power Systems, vol. 21, no. 2, pp. 480-487, May 2006.

[7] Vladimiro Miranda, Nuno Fonseca, "EPSO Evolutionary self-adapting Particle Swarm optimization", internal report INESC Porto, July 2001 (obtainable from the authors by request)

[8] Miranda, V., and Fonseca, N.: "EPSO – Best-of-two-worlds Meta-heuristic Applied to Power System problems". In Proc. of the IEEE Congress on Evolutionary Computation, Vol. 2, pp. 1080 – 1085 (2002).

[9] V. Miranda and N. Fonseca, "EPSO - Evolutionary Particle Swarm Optimization, a New Algorithm with Applications in Power Systems", Proceedings of the IEEE Transmission and Distribution Asia-Pacific Conference 2002, Yokohama, Japan, Oct 2002

[10] V. Miranda, Fellow IEEE, C. Cerqueira and C. Monteiro "Training a FIS with EPSO under an Entropy Criterion for Wind Power prediction" 9th International Conference on Probabilistic Methods Applied to Power Systems KTH, Stockholm, Sweden – June 11-15, 2006

[11] B. Mozafari, A. et , "A hybrid of particle swarm and ant colony optimization algorithms for reactive power market simulation", IOS Press Journal of Intelligent and Fuzzy Systems, Volume 17, Number 6/2006, PP. 557-574.

[12] Clerc, M., "The Swarm and the Queen: Towards a Deterministic and Adaptive Particle Swarm Optimization", Proceedings of the 1999 Congress of Evolutionary Computation, vol. 3, 1951-1957, IEEE Press

[13] H. Mori and Y. Komatsu, "A Hybrid Method of Optimal Data Mining And Artificial Neural Network for Voltage Stability Assessment", Proceedings of IEEE St. Petersburg PowerTech Conference, Russia, June 2005

[14] N. W. Oo and V. Miranda, "Multi-energy Retail Market Simulation with Intelligent Agents", Proceedings of IEEE St. Petersburg PowerTech Conference, Russia, June 2005

[15] C.H. Lee and C.C. Teng, "Tuning of PID Controllers for Stable and Unstable Processes based on gain and phase margin specifications", International Journal of Fuzzy Systems, Vol. 3, No. 1, pp. 346-355. 2001.