Tuning PID Controller Using Hybrid Genetic Algorithm Particle Swarm Optimization Method for AVR System

Faouzi Aboura
Laboratoire des Systèmes Electriques &Industriels (LSEI)
Université des Sciences et Technologie Houari Boumediéne
Bab Ezzouar 16123 , Algeria
faboura@usthb.dz

***Abstract*—The proportional-integral-derivative (PID) controller is widely used in industrial applications, one of these important application is the Automatic Voltage Regulator (AVR), due to the necessity of using controller to avoid instability of the system. In our paper a comparison between algorithms Particle Swarm Optimization (PSO) and Genetic Algorithm (GA) and Hybrid Genetic Algorithm Particle Swarm Optimization (HGAPSO) is proposed with their characteristics and performance analysis to find an optimum parameters of the PID controller, a new objective function is also proposed to take into account the relation between the performance criteria's .**

Keywords—PID controller, AVR system, objective function, optimization, GA, PSO, HGAPSO.

I. INTRODUCTION

The Proportional-Integral-Derivative (PID) controller has been widely used in industrial process, the evidence of their popularity lies in the fact that even today, 90% of the industry employs PID controller .This kind of controller is popular due to simple structure, reliable in operation and robust in performances. One key factor for their success is that they act in the processes under control in a manner closely similar to human's natural responses to outside stimuli, that is the combined effects of spontaneity (proportional action), post training (integral action) and projection into future (derivative action),[1]. This controller is used in our case in the automatic voltage regulator (AVR) that is used with the synchronous generator in order to maintain constant terminal voltage of the generator under normal operating conditions at various load levels [2]. The AVR controls the terminal voltage by adjusting the exciter voltage of the generator. The tuning parameters of PID controller is not easy especially with a commonly Ziegler-Nichols method that is the most common practical control method,[3].It is difficult to achieve the best performance of the system by using Ziegler-Nichols method, and the designer has to depend on his experience for obtaining the best performance. For the past three decades lots of research has been reported on in the intelligent controllers, evolutionary based approaches have received the increasing attention of engineers dealing with problems. GA (Genetic Algorithm) and PSO (Particle Swarm Optimization) have been used in [4]-[8], to enhance optimal tuning of PID controller for AVR (Automatic Voltage Regulator), other methods have been applied recently [9-10]. GA mainly depends on the concept of survival of the fittest. The PSO is motivated by the social behavior of bird flocking and fish schooling. In our paper we used the two previous methods and a new method called based Hybrid Genetic Algorithm Particle Swarm Optimization (HGAPSO) to integrate the advantages of GA and PSO algorithms to find optimal parameters for controller in AVR system, we also analyze performances for each of the three optimization methods.

II. LINEARIZED MODEL OF AN AVR SYSTEM

The AVR system is composed by four main components, namely amplifier, exciter, generator, and sensor as shown in Fig.1, each of these have its own transfer function given in TABLE I.

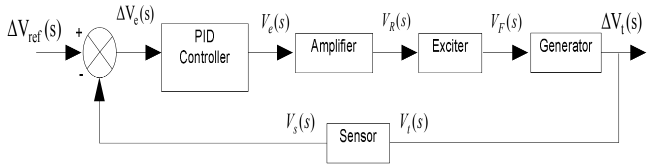


Fig. 1. Block diagram of AVR system with PID controller.

TABLE I. TRANSFER FUNCTION OF AVR COMPONENTS

Component	Transfer Function
Amplifier	$\frac{10}{1+0.1s}$
Exciter	$\frac{10}{1+0.5s}$
Generator	$\frac{0.7}{1+s}$
Sensor	$\frac{1}{1+0.001s}$

The PID controller is in parallel type and his transfer function is:

$$C(s) = K_p + \frac{K_i}{s} + K_d s \quad (1)$$

-The transfer function of the amplifier model is:

XXX-X-XXXX-XXXX-X/XX/$XX.00 ©20XX IEEE
978-1-5386-7687-5/19/$31.00 ©2019 IEEE

$$\frac{V_R(s)}{V_e(s)} = \frac{K_A}{1+\tau_A s} = \frac{10}{1+0.1s} \quad (2)$$

Where K_A are in the range of 10 to 400, and in our work is taken equal to 10, the amplifier time constant τ_A is very small ranging from 0.02 to 0.1 s, and in our work is taken equal to 0.1.

-The transfer function of the exciter model is:

$$\frac{V_F(s)}{V_R(s)} = \frac{K_E}{1+\tau_E s} = \frac{10}{1+0.5s} \quad (3)$$

Where the gain K_E are in the range of 10 to 400, and in our work is taken equal to 10, the single time constant τ_E is from 0.5 to 1.0 s, and in our work is taken equal to 0.5.

-The transfer function of the generator model is:

$$\frac{V_t(s)}{V_F(s)} = \frac{K_G}{1+\tau_G s} = \frac{0.7}{1+s} \quad (4)$$

Where K_G are in the range of 0.7 to 1.0, and in our work is taken equal to 0.7 and constant τ_G vary in function of load from 1.0 to 2.0 s, and in our work is taken equal to 1.0.

-The transfer function of the sensor model is:

$$\frac{V_s(s)}{V_t(s)} = \frac{K_R}{1+\tau_R s} = \frac{1}{1+0.001s} \quad (5)$$

The sensor is modeled by a simple first-order transfer function, where $K_R = 1$ and $\tau_R = 0.001$.

III. Performance Estimation of PID Controller

In our paper we propose a new objective function to integrate the integrated of time-weighted-squared-error (ITSE) in the objective function given in [8] ,the new objective function integrate performance criteria in the time domain and include, the overshoot M_p ,rise time t_r ,settling time t_s and steady-state error E_{ss}.

This new form a objective or fitness function is based on the fact to combine to similar parameter in term of magnitude and nature of parameter.

The new objective function used is:

$$f = \sqrt{((M_p * 100)^2 + (t_s + t_r)^2 + (ITSE + E_{ss})^2} \quad (6)$$

IV. Description and Implementation of Particle Swarm Optimization

Particle swarm optimization (PSO) is firstly introduced by Kennedy and Eberhart [11] in 1995. This algorithm searches the space of an objective function by adjusting the trajectories of individual agents, called particles, as these trajectories form piecewise paths in a quasi-stochastic manner, [12]. PSO is initialized with a group of random particles (solutions), k_p, k_i, k_d and then searches for optima by updating generations.

The basic flowchart of particle swarm optimization is presented in Fig. 2.

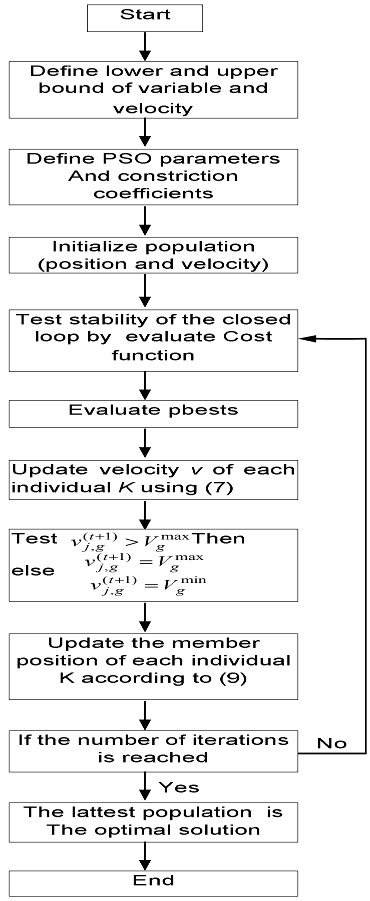


Fig. 2. Algorithm of particle swarm optimization.

The simulation step of the particle swarm optimization (PSO) is shown as follows.

Step 1) Specify the lower and upper bounds of the three controller parameters and initialize randomly the individuals of the population including searching points, velocities, *pbests*, and *gbest* .Acceleration constants c_1 and c_2 were set to be 2.05. These constants represent the weighting of the stochastic acceleration terms that pull each particle toward and positions.

Step 2) For each initial individual of the population, test the closed-loop system stability and calculate the values of the four performance criteria in the time domain, namely M_p, E_{ss}, t_r, and t_s.

Step 3) Calculate the evaluation value of each individual in the population using the evaluation function given by (6).

Step 4) Compare each individual's evaluation value with its pbests. The best evaluation value among the *pbests* is denoted as *gbest*.

Step 5) Modify the member velocity v of each individual K according to (7)

$$v_{j,g}^{(t+1)} = w.v_{j,g}^{(t)} + c_1^* rand()^* (pbest_{j,g} - x_{j,g}^{(t)}) + c_2^* rand()^* (gbest_{j,g} - x_{j,g}^{(t)}) \quad (7)$$
$$j = 1,2,..........,n$$
$$g = 1,2,3$$

Where the value of w that represent inertia weight is set by (8).

$$w = w_{\max} - \frac{w_{\max} - w_{\min}}{iter_{\max}} \times \text{iter} \quad (8)$$

Step 6) If $v_{j,g}^{(t+1)} > V_g^{\max}$,then $v_{j,g}^{(t+1)} = V_g^{\max}$ and if $v_{j,g}^{(t+1)} < V_g^{\min}$,then $v_{j,g}^{(t+1)} = V_g^{\min}$

Step 7) Modify the member position of each individual K according to (9)

$$k_{j,g}^{(t+1)} = k_{j,g}^{(t)} + v_{j,g}^{(t+1)}$$
$$k_g^{\min} \le k_{j,g}^{(t+1)} \le k_g^{\max} \quad (9)$$

Where $k_g^{\min}$ and $k_g^{\max}$ represent the lower and upper bounds, respectively, of member, g of the individual K .

Step 8) If the number of iterations reaches the maximum, then go to Step 9. Otherwise, go to Step 2.

Step 9) The individual that generates the latest $gbest$ is the optimal controller parameter.

V. DESCRIPTION AND IMPLEMENTATION OF GENETIC ALGORITHM

The Genetic Algorithm (GA) is firstly introduced by John Holland in 1975 [13]. GA is a class of stochastic algorithm based on the biological evolution in the natural world. The technological process employs three operators: 1) selection and reproduction, 2) crossover and 3) mutation.
The basic flowchart of genetic algorithm is presented in Fig. 3.

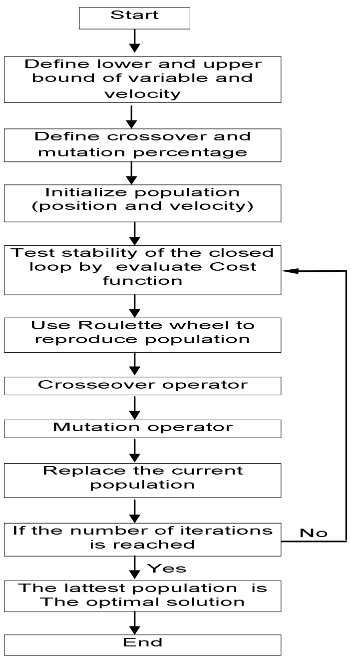


Fig. 3. Algorithm of Genetic Algorithm optimization.

The simulation step of the genetic algorithm optimization (GA) is shown as follows.

Step 1) Specify the lower and upper bounds of the three controller parameters and define crossover and mutation percentage and initialize randomly the individuals of the population. In our case crossover percentage $P_c = 0.70$, mutation percentage $P_m = 0.10$.

Step 2) For each initial individual of the population, test the closed-loop system stability and calculate the values of the four performance criteria in the time domain, namely M_p, E_{ss}, t_r, and t_s.

Step 3) Calculate the evaluation value of each individual in the population using the evaluation function given by (6).

Step 4) Reproduce population using a probabilistic roulette wheel method.

Step 5) Implement crossover operation on the reproduced chromosomes and execute mutation operation.

Step 6) If the number of iterations reaches the maximum, then go to Step 7. Otherwise, go to Step 2.

Step 7) The individual that generates the latest population is the optimal controller parameter solution.

VI. DESCRIPTION AND IMPLEMENTATION OF HYBRID GENETIC ALGORITHM PARTICLE SWARM OPTIMIZATION

The HGAPSO algorithm combines the advantages of swarm intelligence of the PSO algorithm and the natural selection mechanism of the genetic algorithm in order to increase the number of highly evaluated agents at each iteration step.
The basic flowchart of HGAPSO used in this study is given in Fig. 4.

First, multiple solutions are generated randomly as initial population and objective function values are evaluated for each solution. After the evaluation is done, the population is divided into two subpopulations. One of these subpopulations is updated by the GA operation, while the other is updated by the PSO operation.
New solutions created by each operation are combined in the next generation, and non-dominated solutions in the combined population are archived. The archive data is shared between the GA and PSO, i.e., non-dominated solutions created by the PSO can be used as parents in GA, while non-dominated solutions created by GA can be used as global guides in PSO.

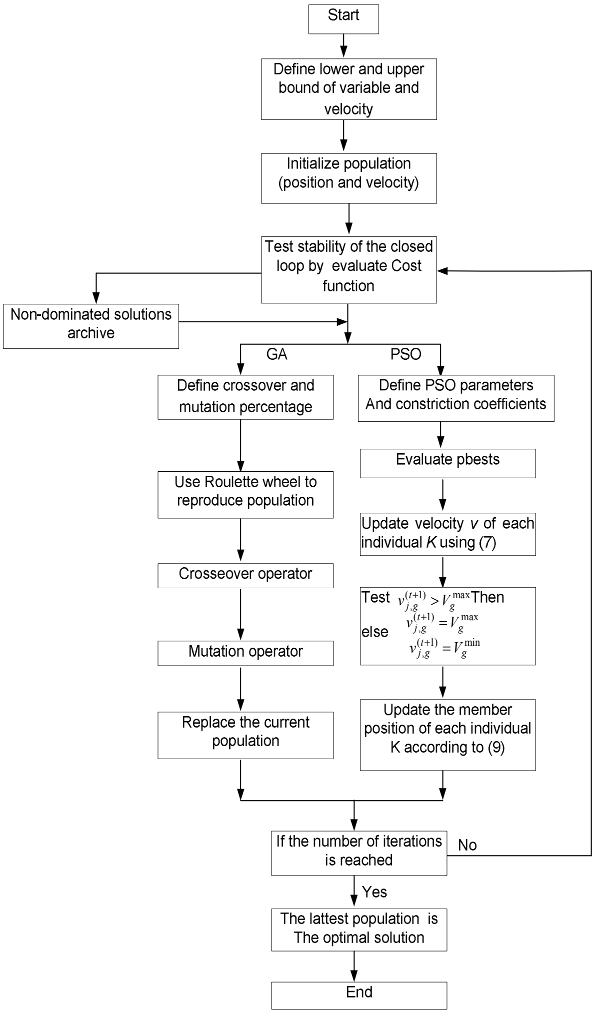
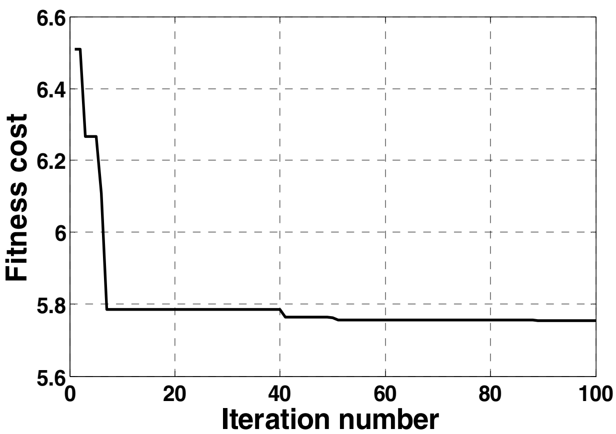


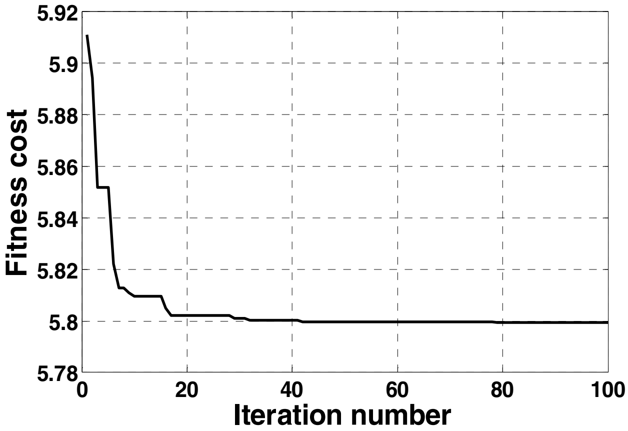
Fig. 4. Algorithm of Hybrid Genetic Algorithm Particle Swarm Optimization .

VII. Numerical Results and Discussion

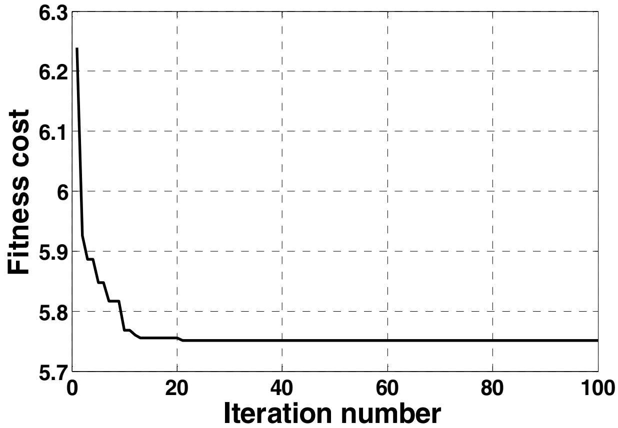
In our paper the maximum number of iterations for the three methods is 100, and the number of population is 100.
The curves corresponding to the fitness cost as a function of iterations number obtained for the three optimization techniques are presented in Fig. 5(a)–(c).



(a)



(b)



(c)

Fig. 5. Fitness cost as a function of iterations number. (a) Genetic Algorithm (GA). (b) Particle Swarm Optimization (PSO). (c) Hybrid Genetic Algorithm Particle Swarm Optimization (HGAPSO).

From Fig. 5(a)–(c) ,first we compare GA and PSO ,we can deduce that the fitness cost of GA is lower than PSO ,but the PSO is faster than the GA and reach the minimum cost at the 40 iteration number, when the GA reach its minimum cost at the 50 iteration number.
The HGAPSO have the advantages described above of the time to reaching the minimum cost value and also the value of this cost value. This algorithm is fastest method and has minimum cost; all the values of the controller parameters and performances are summarized to the Table.2.

TABLE II. Parameters of PID controller and performances for each optimization methods.

Methods	GA	PSO	HGAPSO
Fitness cost	5.7553	5.7994	5.7507
Kp	3.1613	4.5920	3.3047
Ki	0.0189	0.2020	0.1053
Kd	0.6246	0.9507	0.6548
Ess	0.0170	0.0154	0.0108
Mp	0.0563	0.0018	0.0062
ts	2.9315	0.9238	0.7385
tr	4.9984	4.9984	4.9983

The figure.6 represents the terminal voltage step response of the AVR system with the three controllers parameters obtained by the GA, PSO and HGAPSO.

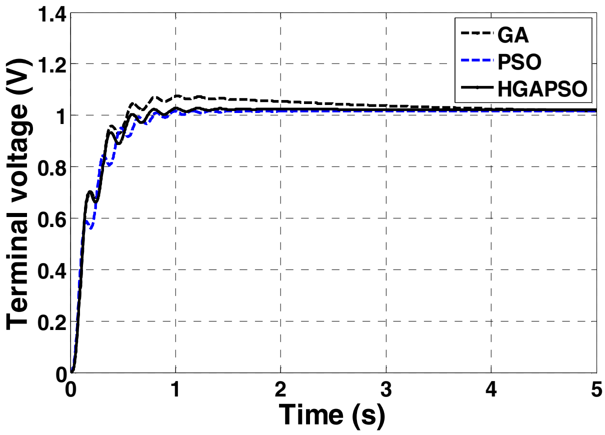


Fig. 6. Terminal voltage step response of the AVR system with the three optimization methods.

VIII. CONCLUSION

This paper study tuning the proportional-integral-derivative (PID) controller with three optimization methods GA, PSO, and a Hybrid version of these two algorithms called HGAPSO. We developed also a new objective function based in the combination of the similar parameter in term of magnitude and nature.

We obtained in our work for AVR system one of the most benefit of the GA method that consist have fitness cost lower than PSO and also that PSO method is faster than the GA, these two characteristics are included in hybrid HGAPSO. Although HGAPSO algorithm is more stable and effective, we prepare further analysis in its internal parameters to study their effect on efficiency.

REFERENCES

[1] K. M. Elbayomy, J. Zongxia and Z. Huaqing, "PID controller optimization by GA and its performances on the electro-hydraulic servo control system," Chinese Journal of Aeronautics, 21(4), 378–384, 2008.

[2] A. H. M. S. Ula and A. R. Hasan, "Design and implementation of a personal computer based automatic voltage regulator for a synchronous generator," IEEE Trans. Energy Convers., vol. 7, no. 1, pp. 125–131, Mar. 1992.

[3] J. G. Ziegler and N. B. Nichols, "Optimum Settings for Automatic Controllers," Transactions of the ASME, Vol. 64, 1942, pp. 759-768.

[4] R. A. Krohling and J. P. Rey, "Design of optimal disturbance rejection PID controllers using genetic algorithms," IEEE Trans. Evol. Comput., vol. 5, no. 1, pp. 78–82, Feb. 2001.

[5] J. Zhang, J. Zhuang, H. Du, and S. Wang, "Self-organizing genetic algorithm based tuning of PID controllers," Inform. Sci., vol. 179, no.7, pp. 1007–1018, Mar. 2009.

[6] D. Devaraj and B. Selvabala, "Real-coded genetic algorithm and fuzzy logic approach for real-time tuning of proportional-integral-derivative controller in automatic voltage regulator system," IET Generation Transmission Distribution, vol. 3, no. 7, pp. 641–649, 2009.

[7] V. Mukherjee and S. P. Ghoshal, "Intelligent particle swarm optimized fuzzy PID controller for AVR system," Electr. Power Syst. Res., vol. 77, no. 12, pp. 1689–1698, 2007.

[8] Z. Gaing , "A particle swarm optimization approach for optimum design of PID controller in AVR system," IEEE Trans. Energy Convers., 2004, 19, (2), pp. 384–391.

[9] S. Ekinci and B. Baran Hekimoğlu, "Improved Kidney-Inspired Algorithm Approach for Tuning of PID Controller in AVR System," IEEE Access., 2019, 7, pp. 39935-39947.

[10] N. Nahas, M. Abouheaf, A.M. Sharaf and W. Gueaieb, "A Self-Adjusting Adaptive AVR-LFC Scheme for Synchronous Generators," IEEE Trans.on Power Systems (Early Access)., 2019, pp. 1-1.

[11] J. Kennedy and R. Eberhart, "Particle swarm optimization," in Proc. IEEE Int. Conf. Neural Networks, 1995, pp.1942-1948.

[12] X-She .Yang, "Engineering optimization, an introduction with metaheuristic applications", Published by John Wiley & Sons, Inc., Hoboken, New Jersey, (2010).

[13] J.H.Holland, "Adaptation in Natural Artificial System", Published by Ann Arbor, the University of Michigan Press, 1975.