Military Technical College
Kobry El-Kobbah,
Cairo, Egypt

8th International Conference
on Electrical Engineering
ICEENG 2012

Optimal PID Controller for AVR System Using Particle Swarm Optimization

H. F. Abu-Seada[1] W. M. Mansor[2] F. M. Bendary[2] A. A. Emery[3] M. A. Moustafa Hassan[4]
[1] Ministry of Electricity, Cairo, Egypt
[2]Faculty of Engineering "Shoubra" Benha University, Cairo, Egypt
[3]National Energy Control Center, Cairo, Egypt
[4] Elec. Power Dept., Cairo University, Giza Egypt
haniseif@yahoo.com, wagdy_ibrahim2001@yahoo.com,
fahmybendary10@yahoo.com, adelaly188@yahoo.com

Abstract:

This paper presents a method to get the optimal tuning of Proportional Integral Derivative (PID) controller parameters for an AVR system of a synchronous generator using Particle Swarm Optimization (PSO) algorithm. The AVR is not initially robust to variations of the power system parameters. Therefore, it was necessary to use PID controller to increase the stability margin and to improve performance of the system. Tuning of optimum (PID) controller parameter yield high quality solution. A new criterion for time domain performance evaluation was defined. Simulation for comparison between the proposed method and Ziegler-Nichols method is done. The proposed method was indeed more efficient also. The terminal voltage step response for AVR model will be discussed in different cases and the effect of adding rate feed back stabilizer to the model on the terminal voltage response. Then the rate feedback will be compared with the proposed PID controller based on use of (PSO) method to find its coefficients. Different simulation results are presented and discussed.

1. Introduction:

The main function of AVR loop is to control the generator terminal voltage. The total generation must meet the total load requirement of both active and reactive power. The load active demand is voltage and frequency dependent. It is generally increases as voltage or frequency increases (within the safe operational limits). In order to improve the performance of the AVR systems, the PID controller is normally used since it has simple structure. The reason of this acceptability is for its simple structure which can be easily understood and implemented. Easy implementation of hardware and software has helped to gain its popularity.Several approaches have been documented in literatures for determining the PID controller parameters. Most famous methods are Ziegler Nichols tuning [1], neural network [2], fuzzy based approach [3], and Genetic Algorithm [4]. The results of the simulation show that when the PSO method is used the performance of the tuned PID controller is significantly more efficient and the response is better in quality.

2. AVR Model:

The Automatic Voltage Regulator (AVR) of the synchronous generator is responsible for controlling the terminal voltage and reactive power output of the generator and consequently its terminal voltage. A simple (AVR) consists of amplifier, exciter, generator and sensor, The block diagram of AVR with PID Controller is shown in Figure (1). The linear model for each of the AVR elements is given in the following discussion as given in [5].

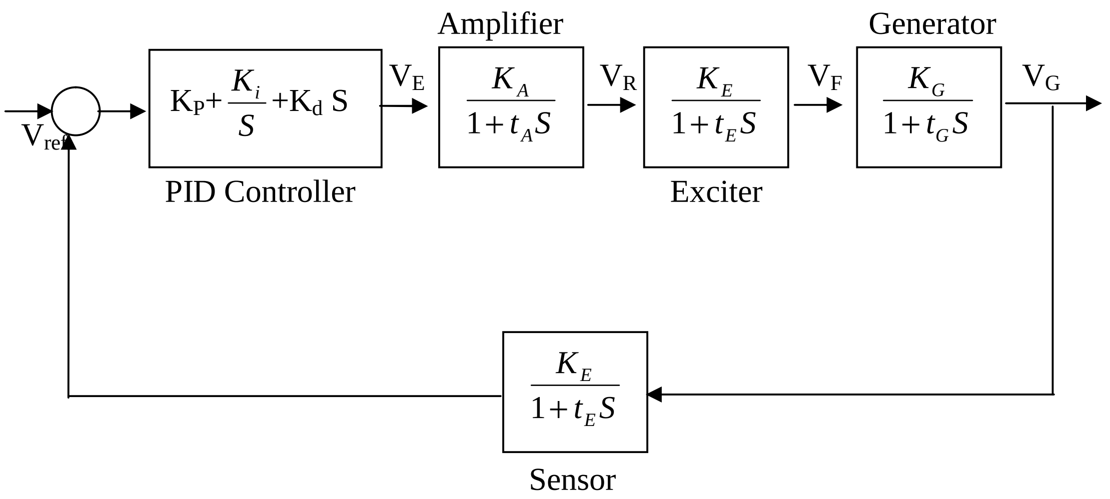

Figure (1): Block diagram of AVR with PID controller.

a) Amplifier model: $$\frac{V_R}{V_E} = \frac{K_A}{1+\tau_A S} \quad (1)$$

b) Exciter Model: $$\frac{V_F}{V_R} = \frac{K_E}{1+\tau_E S} \quad (2)$$

c) Generator Model: $$\frac{V_t}{V_F} = \frac{K_G}{1+\tau_G S} \quad (3)$$

d) Sensor Model: $$\frac{V_S}{V_t} = \frac{K_R}{1+\tau_R S} \quad (4)$$

3. Particle Swarm Optimization:

A Particle Swarm Optimization (PSO) is a population-based stochastic optimization algorithm modeled after the simulation of the social behavior of bird and fish school. The particle swarm optimizer was first introducing by Kennedy and Eberhart [6]. PSO is basically developed through simulation of bird flocking in two-dimension space. The position of each agent is represented by XY axis position and also the velocity is expressed by V_x (the velocity of X axis) and V_y (the velocity of Y axis). Modification of the agent position is realized by the position and velocity information. In this thesis PSO will be used therefore, a details description of PSO will be presented. Bird flocking optimizes a certain objective function. Each agent knows its best value so far (pbest) and its XY position. This information is analogy of personal experiences of each agent. Moreover, each agent knows the best value so far in the group (gbest) among pbests. This information is analogy of knowledge of how the other agents around them have performed. Namely, each agent tries to modify its position using the following information:

- The current positions (x,y),
- The current velocities (Vx , Vy),
- The distance between the current position and pbest
- The distance between the current position and gbest

This modification can be represented by the concept of velocity. Velocity of each agent can be modified by the Equation (5):

$$v_i^{k+1} = wv_i^k + c_1 rand_1 \times (pbest_i - x_i^k) + c_2 rand_2 \times (gbest - x_i^k) \quad (5)$$

The inertia weighting function w that has been mentioned in equation (5) is a linearly decreasing function. The parameter selection of this function is examined by Shi and Eberhart. According to their examination, the parameters are ranged from about 0.9 to 0.4 during iterations procedure [7, 8]. The values of these parameters are appropriate for power system problems [9, 10]. This function can be calculated from the Equation (6):

$$w = w_{\max} - \frac{w_{\max} - w_{\min}}{iter_{\max}} \times iter \tag{6}$$

The current position (searching point in the solution space) of each agent can be modified by the Equation (7):

$$x_i^{k+1} = x_i^k + v_i^{k+1} \tag{7}$$

Recent work done by Clerc [11] indicates that use of a constriction factor may be necessary for the convergence of the particle swarm optimization technique. The particle velocity equation using constriction factor will be:

$$v_i^{k+1} = K * [v_i^k + c_1 rand_1 \times (pbest_i - x_i^k) + c_2 rand_2 \times (gbest - x_i^k)] \tag{8}$$

$$K = \frac{2}{\left|2 - \varphi - \sqrt{\varphi^2 - 4\varphi}\right|} \tag{9}$$

$$where \quad \varphi = c_1 + c_2 \,, \ \varphi > 4$$

As φ increases above 4, as K decreases below 1. The using of constriction factor results in convergence of the particle over time. The particle swarm optimization technique using constriction factor controls the system behavior which ensures the convergence of the system in a real value region which means that the particle swarm optimization technique using constriction factor generates higher quality solutions than the particle swarm optimization technique using an inertia weight [12].

The general steps of PSO can be described as follows:

Step 1: Generation of initial condition of each agent.

Step 2: Evaluation of searching point of each agent (the objective function value calculated for each agent).

Step 3: Modification of each searching point.

Step 4: Checking the exit condition.

4. Stochastic Particle Swarm Optimization Technique

The Particle Swarm Optimization Technique (PSOT) was first introduced by Kennedy and Eberhart [6]. The particles are "flown" through the search space by updating the position of the i^{th} particle at time step t according to the equation (7). The velocity updates are governed by the Equation (10):

$$v_i(t+1) = \omega v_i(t) + c_1 rand_1 \times (pbest_i - x_i(t)) + c_2 rand_2 \times (gbest - x_i(t)) \quad (10)$$

While empirical evidence has accumulated that the algorithm "works", e.g., it is a useful tool for optimization, and there has thus far been little insight into how it works. Ozcan and Mohan have published the first mathematical analysises regarding the trajectory of a PSO particle [13, 14]. From theoretical analysis [15], the trajectory of the particle $x_i(t)$ converges on to a weighted mean of p_i and p_g .It is important to note at this stage that if the trajectory of the particle converges, then it will do so towards a value derived from the line between its personal best position and the global best particle's position. Due to update equation, the personal best position of the particle will gradually move closer to the global best position, so that the particle will eventually converge on the position of the global best particle. At this point, the algorithm will not be able to improve its solution, since the particle will stop moving. This has no bearing on whether the algorithm has actually discovered the minimum of the function in fact, there's no guarantee that the position on which the particle has converged is even a local minimum. The stochastic nature of the particle swarm optimizer makes it more difficult to prove (or disprove) like global convergence. F.Soils and R.Wets [16] have studied the convergence of stochastic search algorithms, most notably that of pure random search algorithms, providing criteria under which algorithms can be considered to be global search algorithms, or merely local search algorithms. Frans Van Den Bergh [17] used their definitions extensively in the study of the convergence characteristics of the PSO and the guaranteed convergence PSO (GCPSO), he proved the PSO is not even guaranteed to be local extreme, and GCPSO can converge on a local extreme. Due to demerits of the basic particle swarm optimization technique [18], a new stochastic particle swarm optimizer is introduced as follow: Let ω equal zero, the update for equations (7) and (10) can be combined as follow:

$$x_i(t+1) = x_i(t) + c_1 rand_1 \times (pbest_i - x_i(t)) + c_2 rand_2 \times (gbest - x_i(t))] \quad (11)$$

This formula reduces the global search capability, but increases the local search capability. To improve the global search capability, we conserve the current best position of the swarm P_g and the j's best position P_j , then mainly using Tabu Search (TS) [19] to give a new particle j's position $x_j(t+1)$, and other particles are manipulated according to (12). The new particle j's position $x_j(t+1)$ can be calculated as follow:

$$\begin{aligned} x_j(t+1) &= G_1(x_j(t)), \text{ if } (\text{random} < \text{P}_{\text{select}}) \\ x_j(t+1) &= G_2(x_j(t)), \text{ other wise} \end{aligned} \quad (12)$$

Where Pselect is a parameter within (0.01, 0.1), and random is uniform random sequences sampled from U (0, 1).$G_1(x)$ is a function which uniformly sample from the domain, and $G_2(x)$ is a TS technique.

5. Simulation Results

In order to evaluate the performance quality of the proposed (AVR) as tuned by (PSO) method, it is compared with that obtained using Ziegler tuning method [1].The block diagram of the employed AVR system is shown in Figure (1).Two AVR systems have the following specifications at Table (1). These parameters are according to the block diagram given in Figure (1).

Table (1): *Parameters of the AVR Systems (1), (2) of the Generator:*

System (1) [5]	**GAIN**	**Time Constant**	System (2) [20]	**GAIN**	**Time Constant**
Amplifier	K_A=10	τ_A =0.1	**Amplifier**	K_A=40	τ_A =0.01
Exciter	K_E=1	τ_E =0.4	**Exciter**	K_E=0.2	τ_E = 4
Generator	K_G=1	τ_G =1	**Generator**	K_G=1	τ_G =1
Sensor	K_R=1	τ_R =0.05	**Sensor**	K_R=1	τ_R =0.01

From the results shown in Figure (2), it is seen that for amplifier gain (K_A=10), the response is highly oscillatory, with a very large overshoot and a long settling time.

Furthermore, the steady-state error is over 9 percent. We couldn't have a small steady-state error and satisfactory transient response at the same time.

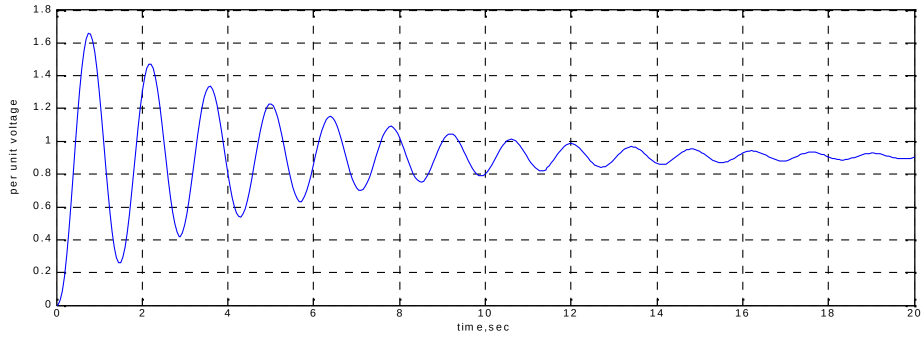

Figure (2): Terminal Voltage Step Response of System (1) without Controller

Table (2): The time-domain performance specifications for system (1) without controller:

Peak time	0.791 sec
percent overshoot	82.46%
Rise time	0.247 sec
Settling time	19.04 sec

Table (3) illustrates the PID gains using Ziegler – Nichols:

Table (3): Table PID Gains Using Ziegler-Nichols [18]

General PID gains	K_{pr}	K_i	K_D
ZN PID gains	$0.6K_{cr}$	$\frac{K_{\Pr}}{0.5P_{cr}}$	$\frac{K_{\Pr}P_{cr}}{8}$

-The controller coefficients according to Zeigler method for AVR system (1) were found to be as shown in Table (4).

Table (4): Controller Coefficients According to Ziegler-Nichols method for System (1):

$\mathbf{K_p}$	0.7296
$\mathbf{K_i}$	1.118
$\mathbf{K_d}$	0.119

From Figure (3), It can be seen that for the same amplifier gain (KA=10), and when we add PID tuned PID controller by Zeigler- Nichols method the response oscillatory, with 60% overshoot and (4.4 s) settling time. But we can see that the PID controller eliminate the steady-state error. The eigen values for system (1) with Zeigler-Nichols: W = -24.671, -2.2165±7.1548*i* , -2.1979±3.0433*i*

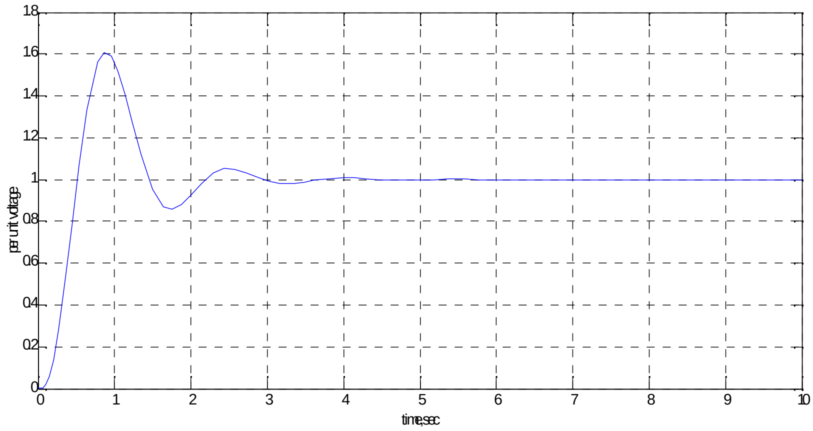

Figure (3): AVR Response Using Ziegler-Nichols for Tuned PID Controller, System (1)

Table (5): *Performance of the Zeigler- Nichols Tuned PID Controller for system (1):*

Parameter	**Value**
Max over shoot	60%
Settling time	4.4 sec
Rise time	0.31sec

The controller coefficients according to the (PSO) are found to be as shown in Table (6):

Table (6): *Controller Coefficients According to (PSO) for system (1):*

$\mathbf{K_p}$	0.3680
$\mathbf{K_i}$	0.3413
$\mathbf{K_d}$	0.0927

From Figure (4) we can see that the G_{best} value is = -2.9329 which mean that the system is more stable as from the eigen values for the AVR system when we used Zeigler-Nichols for tuning PID controller the max real part at the negative direction is (-2.2165), which shows us that when we used (PSO) in tuning PID controller, the AVR system is more stable and has a satisfactory response. The eigen values for system (1) with (PSO): W = -21.768, $-2.9329 \pm 0.61729i$,$-2.9068 \pm 0.36128i$

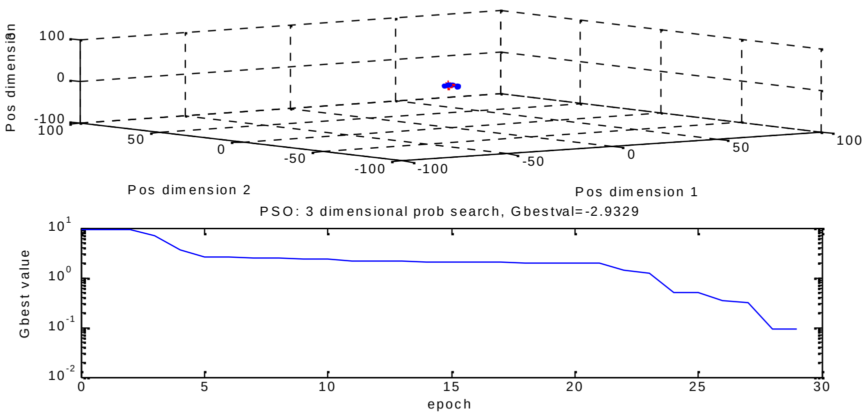

Figure (4): Search Process of Optimal Parameter Values of AVR system (1) by Using PSO

From Figure (5): the comparison between (ZN) and (PSO) shows that the PSO has more effectiveness for tuning PID controller as the coefficients obtained by using PSO improved the response of the PID controller .There is an overshoot of (19%) which less than the over shoot by using ZN method also, at PSO we have a settling time of (2.2 s) which shorter than the settling time of ZN. These results show that PSO optimization is better than ZN method for tuning the PID controller of AVR system (1).

Table (7): *Performance of the PSO tuned PID Controller for System (1):*

Parameter	**Value**
Max over shoot	19%
Settling time	2.2 sec
Rise time	0.56 sec

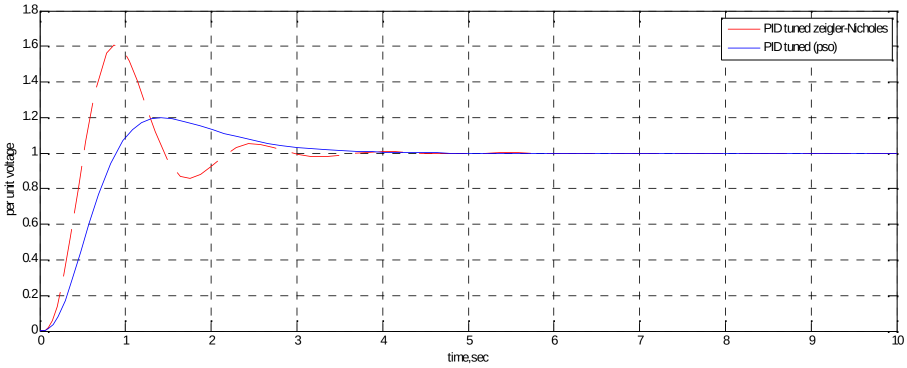

Figure (5): Comparison between Zeigler-Nichols and PSO for System (1)

Table (8): ***Comparison between Zeigler-Nichols and PSO Eigen Values for System (1)*****.**

Eigen Values	Zeigler-Nichols	PSO
1th 2 nd 3 rd 4 th 5 th	-24.671 -2.2165+7.1548*i* -2.2165 -7.1548*i* -2.1979 + 3.0433*i* -2.1979 - 3.0433*i*	-21.768 -2.9329 + 0.61729*i* -2.9329 - 0.61729*i* -2.9068 + 0.36128*i* -2.9068 - 0.36128*i*
Dominant Roots	-2.2165 ± 7.1548*i* -2.1979 ± 3.0433*i*	-2.9329 ± 0.61729*i* -2.9068 ± 0.36128*i*
Relative stability	Low	High

<u>For AVR System (2)</u>

-Response of system (2) when equipped by PID controller which is tuned by (PSO) technique.

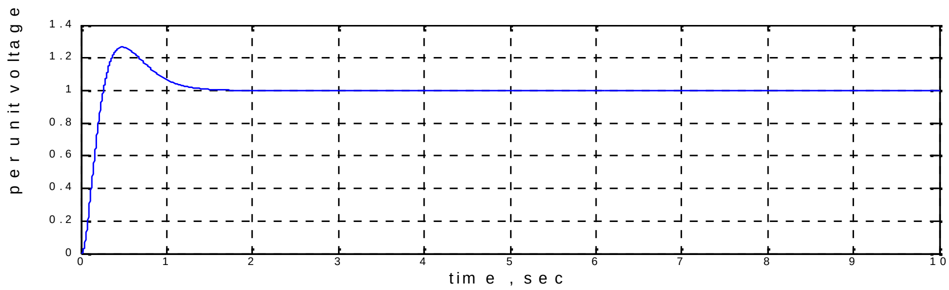

Figure (6): AVR Response Using (PSO) Tuned PID Controller System (2), the PID Controller Eliminate the Steady State Error

Table (9): *Controller Coefficients According to (PSO) for system (2):*

K_p	48.9305
K_i	99.8175
K_d	7.6213

Table (10): Performance of the PSO tuned PID Controller for system (2):

Parameter	Value
Max over shoot	26%
Settling time	1.2sec
Rise time	0.18 sec

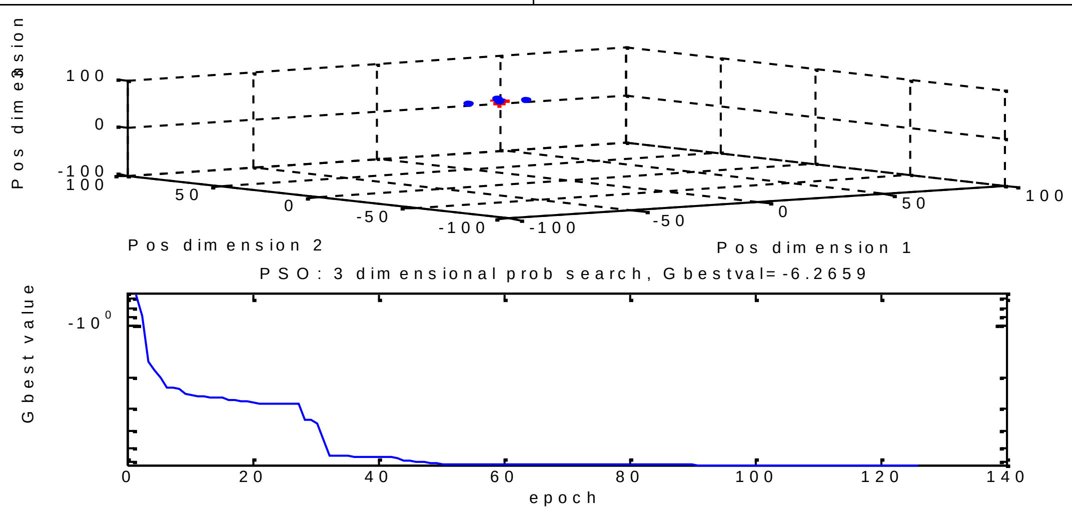

Figure (7): Search Process of Optimal Parameter Values of an AVR System (2) by Using PSO

The eigen values for system (2) with (PSO): W = -133.17,-45.31, -12.921, -4.9222 ±1.1731i

***Table (11)**: Controller Coefficients According to (Ziegler- Nichols) for System (2):*

$\mathbf{K_p}$	34.8
$\mathbf{K_i}$	118.69
$\mathbf{K_d}$	2.55084

***Table (12):** Performance of the Zeigler- Nichols tuned PID Controller System (2):*

Parameter	**Value**
Max over shoot	81%
Settling time	4.6 sec
Rise time	0.13 sec

The eigen values for system (2) with Zeigler-Nichols:

W = -119.58,-76.253, -3.9863, -0.71568±8.495i

From Figure (8), it can be seen, that the G_{best} value is = -6.2659 which mean that the system is more stable as from the eigen values for the AVR system when we used Zeigler-Nichols for tuning PID controller the max real part at the negative direction is (-3.9863), which shows that when PSO is used in tuning PID controller the AVR system is more stable and has a satisfactory response. From Figure (9), the comparison between Zeigler-Nichols and (PSO) shows us that the (PSO) is more efficient for tuning PID controller as the coefficients obtained using PSO make the PID to controller give better voltage response. These results show that PSO optimization gave better response in this case than Zeigler–Nichols method for tuning the PID controller of AVR system (2).

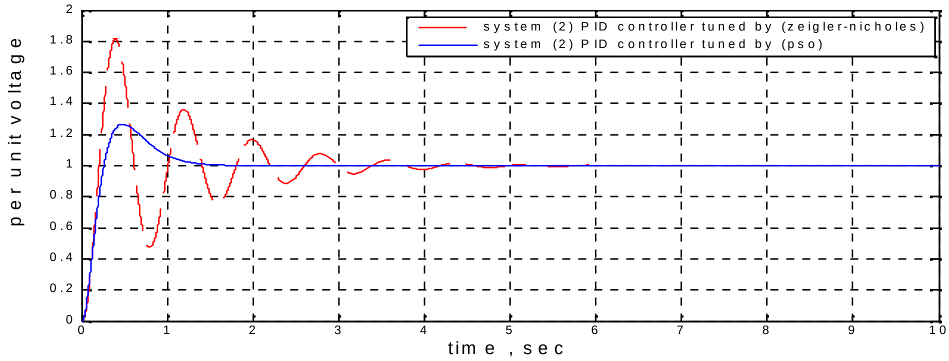

Figure (8): Comparison between Zeigler-Nichols and PSO for System (2)

Table (13): ***Comparison between Zeigler-Nichols and PSO Eigen Values for System (2)***

Eigen Values	Zeigler - Nichols	PSO
1th	-119.58	-133.17
2nd	-76.253	-45.31
3rd	-3.9863	-12.921
4th	$-0.71568 - 8.495i$	$-4.9222 - 1.1731i$
5th	$-0.71568 + 8.495i$	$-4.9222 + 1.1731i$
Dominant Roots	-3.9863 $-0.71568 \pm 8.495i$	-12.921 $-4.9222 \pm 1.1731i$
Relative Stability	Low	High

6. The Robustness Check:

According to the external coefficients like temperature and the life span of the electronic components forming the (AVR) system which can affect on the behavior of (AVR) when dealing with the different disturbances that can be occurred. The robustness test will be occurred at the case of ±20% of normal values for amplifier parameters τ_A , K_A .

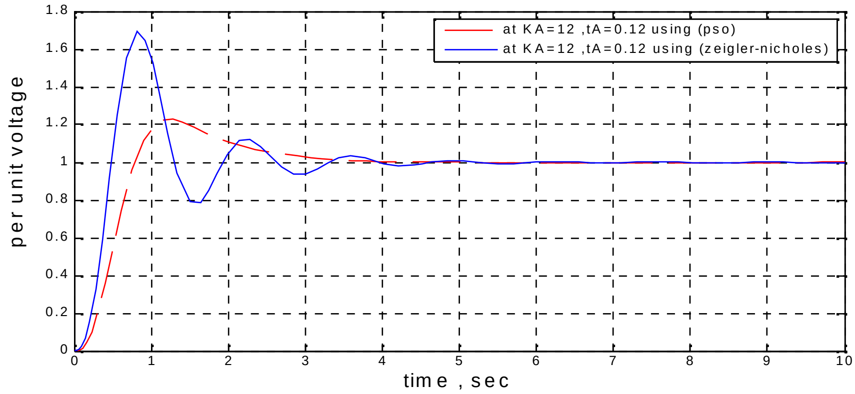

Figure (9): Comparison between ZN and PSO Behavior for System (1) to deal with changes of (+20%) for Amplifier Parameters

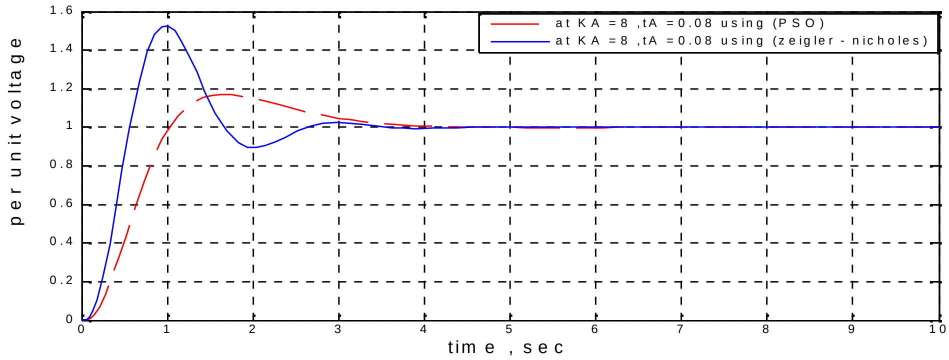

Figure (10): Comparison Between Zeigler – Nichols and PSO Behavior for System (1) to deal with changes of (-20%) for Amplifier Parameters

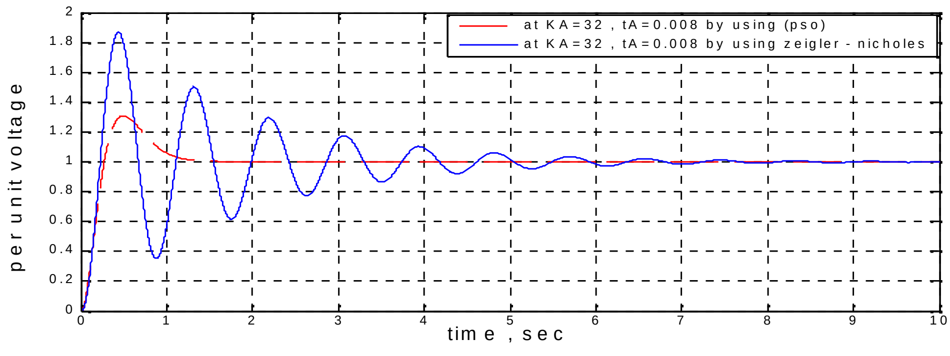

Figure (11): Comparison Between ZN and PSO Behavior for System (2) to Deal with Changes of (-20%) for Amplifier Parameters

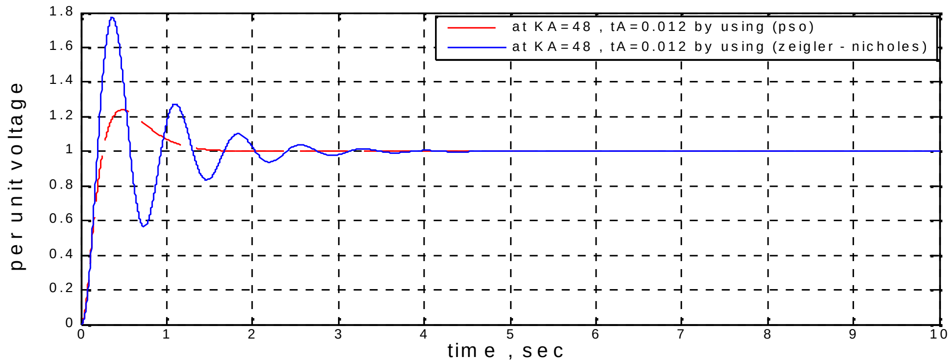

Figure (12) Comparison between ZN and PSO Behavior for System (2) to deal with Changes of (+20%) for Amplifier parameters

When the PID controller is tuned by (PSO) it has an effectiveness to deal with the disturbance and changes by a very good behavior as we can see there is no changes at the settling with a constant rise time although, we have an increasing at the max over shoot when the parameter changes by (-20%) and decreasing when changes by (+20%). Finally we can say that the tuning for PID controller by (PSO) achieve reliability for the AVR system and improve its ability to face any expected disturbance according to changes of parameters as we saw and the tuned controller show a very effected behavior and we can say that the system do not feel any changes according to the disturbance.

7. Excitation System Stabilizer – Rate Feed Back

The stabilizer – rate feed back model of time constant is $\tau_F = 0.04$ second and the derivative gain is adjusted to $K_F = 2$ will be considered [5]. And we will discuss its effective when we put it with the PID controller which tuned with (PSO) for system (1), (2). In Figures (13), (14): it can be seen that addition of rate feed back stabilizer improve the behavior of the controller effectively as the results show a very satisfactory transient and a long rise time with a negligibly over shoot but when we add rate feed back to PID controller it gives a long settling time of (6 sec for system (2) and 7 sec for system (1)) which that is a disadvantage of stabilizer and we avoid the steady state error as the controller reduces the steady-state error to zero.

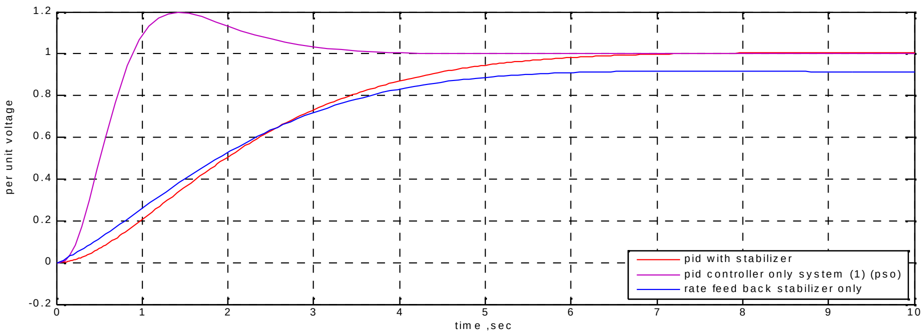

Figure (13): Terminal Voltage Step Response for System (1)

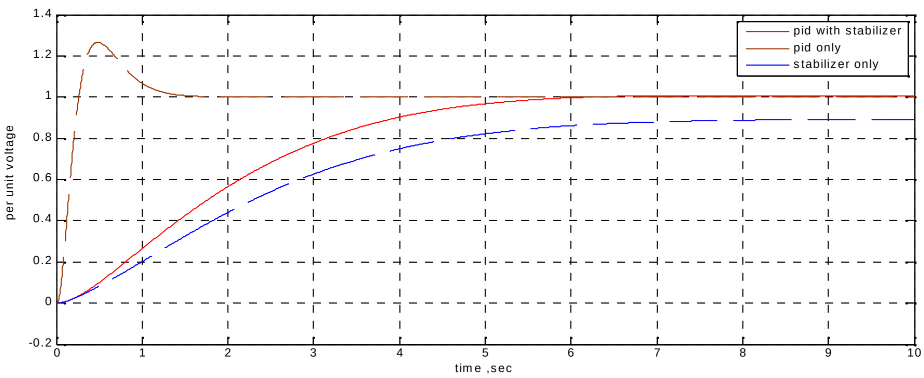

Figure (14): Terminal Voltage Step Response for System (2)

8. Conclusions:

In this paper, a PSO for the optimal design of an AVR system based on PID controller has been introduced. Tuning of optimum (PID) controller parameter yield high quality solution, new criteria for time domain performance evaluation was defined. Simulation results compared between the proposed method and ZN method. The proposed method was indeed more efficient also. The terminal voltage step response for AVR model was discussed in different cases as well as the effect of adding rate feed back stabilizer to the model on the terminal voltage response. Then the rate feed back will be compared with PID controller which (PSO) is used to find its coefficients.

9. References

[1] Ziegler JG, Nichols NB. Optimum settings for automatic controllers TranASME 1942;64:759–68. Jlhlj

[2] Q.H. Wu, B.W. Hogg, G.W. Irwin, A neural network regulator for turbogenerators, IEEE Trans. Neural Networks 3 (1992) 95–100.

[3] A. Visioli, “Tuning of PID controllers with fuzzy logic,” Proc. Inst. Elect. Eng. Contr. Theory Applicant., vol. 148, no. 1, pp. 1–8, Jan. 2001.

[4] R.A. Krohling, J.P. Rey, Design of optimal disturbance rejection PID controllers using genetic algorithm, IEEE Trans. Evol. Comput. 5 (February) (2001) 78–82

[5] H. Saadat, Power System Analysis. New York: McGrawHill, 1999.

[6] J. Kennedy and R. Eberhart, “Particle Swarm Optimization”, Proceedings of IEEE International Conference on Neural Networks (ICNN’95), Vol. IV, pp.1942 1948, Perth, Australia, 1995.

[7] E. Polak, Optimization: Algorithms and Consistent Approximations, Springer New York, 1997.

[8] Y. Shi and R. Eberhart, “Parameter Selection in Particle Swarm Optimization", Proc. Of the 1998 Annual Conference on Evolutionary Programming , San Diego ,1998.

[9] Y. Shi and R. Eberhart, “A Modified Particle Swarm Optimizer”, Proc. Of IEEE International Conference on Evolutionary Computation (ICEC'98), Anchorage, Alaska,May 4-9, 1998.

[10] M. A. Abido , " Particle Swarm Optimization for Multimachine Power System Stabilizer Design", Proc. of IEEE Power Engineering Society Summer Meeting , July 2001.

[11] S. Naka, T. Genji, T. Yura, and Y. Fukuyama, “Particle Distribution State Estimation Using Hybrid Particle Swarm Optimization ", Proc. of IEEE Power Engineering Society Winter Meeting, Columbus, Ohio, January 2001.

[12] M. Clerc, The Swarm and the queen : towards a deterministic and adaptive particle swarm optimization. Proc. ICEC, Washington, DC, pp 1951-1957, 1999.

[13] R. Eberhart and Y. Shi, " Comparing Inertia Weights and Constriction Factors in Particle Swarm Optimization ", Proc . Of the Congress on Evolutionary Computation (CEC2000), pp.84-88, 2000.

[14] E. Ozcan and C. K.Mohan, Analysis of a Simple Particle swarm Optimization System, Intelligence Engineering Systems Through Artificial Neural Networks, pp: 253-258, 1998.

[15] E. Ozcan and C. K.Mohan, Particle Swarm Optimization: Surfing the waves, Proc. of the Congress on Evolutionary Computation, pp: 1939-1944, 1999.

[16] M. Clerc and j. Kennedy, the Particle Swarm Explosion, Stability and Convergence in a Multi-Dimensional Complex Space, IEEE Trans. On Evolutionary Computation, Vol.6, pp: 19-30, 2002.

[17] F. Soils, R. Wets, Minimization by Random Search Techniques, Mathematics of Operations Research, Vol.6, pp: 19-30, 1981.

[18] M. A. Ebrahim Mohamed M.SC. in Electrical Engineering, “AI Based Decentralized Load Frequency Controllers for Interconnected Power Systems 2009.

[19] F. Glover, Future Paths for Integer Programming and Links to Artificial Intelligence, Computers and opertations Research, Vol.5, pp: 533-549, 1986.

[20] K. H. Gharib. O.S. Ebrahim, “Application of the Genetic Algorithm to Design an Optimal PID controller for the AVR system”, MTC Conference, Cairo, Egypt, 2009.