ISSN 2522-6967

Tuning of Velocity PID-Fuzzy Power System Stabilizer by Particle Swarm Optimization

T. Hussein Elmenfy[a]*and Samah Abdelsalam[b]

[a]Electrical and Electronics Engineering Dept, University of Benghazi, Benghazi , Libya
[b] Higher Institute for Technical and Science, Regdalleen, Libya
*Auther e-mail: Tawfiq.elmenfy@uob.edu.ly

ABSTRACT

Power system stabilizer (PSS) is used to damping power system oscillation. It will act as supplementary feedback through the generator excitation system, which produce a component of electrical torque in phase with speed variation. Fuzzy controllers are nonlinear. It is more difficult to set the fuzzy controller gains compared to conventional PID controller. This research proposed velocity (PID-FPSS) its gains are tuned by particle swarm optimization technique (PSO). The objective is to damp local-area oscillation that occur following power system small or large disturbances. The effectiveness of the proposed technique is illustrated by applying the velocity PID-FPSS to a single-machine infinite bus power system that is typically used in the literature to test the performance of power system stabilizers. The simulation has been conducted in MATLAB, SIMULINK (R2013a) package . A comparison between the proposed PID-FPSS and a well-designed robust power system stabilizer (RPSS) confirms the superiority of the proposed technique.

Keywords: single machine infinite-bus system, velocity pid controller, fuzzy logic system, and swarm optimization technique.

1. Introduction

Power system stabilizers are used for many years as supplementary feedback control signal in automatic voltage regulator (AVR) to add damping of the electromechanical oscillations. It will act through the generator excitation system, which produces a component of electrical torque in phase with speed variations to the speed deviation [1, 2].

The parameters of conventional PSS (CPSS) [3] are derived from mathematical model of the plant and designed at one operating condition. This confirm that CPSS is not a suitable for a wide range of operating conditions. In CPSS the parameters are evaluated at particular loading conditions.

The design requirements are considered in [4] introduced both time domain and frequency domain specifications which are initially specified before designing the PSS controller, the optimization based linear control design technique is used to determine the optimal controller parameters. [5] proposed fuzzy expert system, the generator speed deviation and acceleration are chosen as input signals to fuzzy logic power system stabilizer and the desired output is integral square time square error and simulation results shows the superior of proposed technique over CPSS.

Fuzzy logic based PSS for stability enhancement of a two-area four-machine system are designed in [6]. In order to accomplish the stability enhancement, speed deviation ($\Delta\omega$) and active power deviation (ΔP) of the rotor synchronous generator were taken as the inputs to the fuzzy logic controller. These variables take significant effects on damping the generator shaft mechanical oscillations. The stability signals were computed using fuzzy membership function depending on the variables. The design of a proportional, derivative and integral (PID) based power system stabilizer (PSS) introduced in [7], that design carried out using a new Meta heuristic harmony search algorithm (HSA) to optimize the parameters. The design of PID controller is considered with an objective function based on eigenvalue shifting to guarantee the stability of nonlinear plant for a wide range of conditions using HSA. The use of the cost function to minimize the summation of the absolute value is used to design Conventional PSS in [8], and the absolute value is composed of the difference between the square of required compensation phase in excitation system and the square of phase provided by CPSS which should be minimum.

In this paper, a velocity PID-Fuzzy Power System Stabilizer and its gains are tuned by Particle Swarm Optimization is presented. This stabilizer uses the speed deviation, the derivative of the speed deviation and the output electrical -power as inputs to the fuzzy controller (obtained online and assumed to be measured from the output of the plant). This controller is used to provide the necessary damping to the plant.

2. Velocity PID Controller

On form of the controller widely used in the industrial process, control is called a three terms of Proportional-Integral-Derivative (PID) controller [9]. The PID controller are the most commonly used in industrial process (plant) control. The PID controller has the following formula.

$$u(t) = c_{0+}K_p\left[e(t) + \frac{1}{T_i}\int_0^t e(t)dt + T_d\frac{d}{dt}e(t)\right] \qquad (1)$$

Where:

$u(t)$:Control action

K_p: Proportional gain

T_i: Integration time

T_d: Derivative time

c_0: Controller bias

$e(t) = y_{sp} - y_s$= set point- process measurement.

The disadvantages of the controller in eq.(1) are the derivative of the error is subject to the derivative kick and the integral part not compatible to the digital computer. Also set point (y_{sp}) can change instantly in a step change- this cause $e(t)$ to have a step change whenever set point is changed.

The sudden step change in $e(t)$ cause the derivative of $e(t)$ to be infinite- this cause the derivative part to be huge whenever the set point is changed (derivative kick).

To eliminate the derivative kick, the derivative part should be calculated based on the measurement (y_s) rather than the error e(t) as in eq.(2).

$$u(t) = c_0 + K_p\left[e(t) + \frac{1}{T_i}\int_0^t e(t)dt + T_d\frac{d}{dt}y_s(t)\right] \quad (2)$$

As mentioned before, the integral part not compatible with digital computers.

Since, $e(t) = y_{sp} - y_s$ (3)

Then,

$$\frac{d}{dt}e(t) = \frac{d}{dt}y_{sp} - \frac{d}{dt}y_s \quad (4)$$

Under normal operating conditions, when set point is not changing:

$$\frac{d}{dt}e(t) = -\frac{d}{dt}y_s \quad (5)$$

The integral part can be replaced with summation and the derivative part replaced by finite difference.

Then eq.(2) can be rewrite at time (t) as:

$$U(t) = c_0 + K_p\left[e(t) + \frac{\Delta t}{T_i}\sum_{i=1}^{n} e(\Delta t) - T_d\left(\frac{y_s(t) - y_s(t-\Delta t)}{\Delta t}\right)\right] \quad (6)$$

In eq.(6) summation can cause overflow errors on digital computers and (Δt) is sampling time and must be small compared to process response time.

Velocity PID controller form development as:

- Evaluate position form at (t) to get u(t).
- Evaluate position form at $(t - \Delta t)$to get $u(t - \Delta t)$.
- Subtract to get $\Delta u(t) = u(t) - u(t - \Delta t)$

$$u(t-\Delta t) = c_0 + K_p\left[e(t-\Delta t) + \frac{\Delta t}{T_i}\sum_{i=1}^{n} e(\Delta t) - T_d\left(\frac{y_s(t-\Delta t) - y_s(t-2\Delta t)}{\Delta t}\right)\right] \quad (7)$$

Subtract Eq.(6) from Eq. (7) to get velocity form:

$$\Delta u(t) = K_p\left[e(t) - e(t-\Delta t) + \frac{\Delta t}{T_i}e(t) - T_d\left(\frac{y_s(t) - 2y_s(t-\Delta t) + y_s(t-2\Delta t)}{\Delta t}\right)\right] \quad (8)$$

The gains to be tuned in velocity PID controller by PSO are $K_p,\ T_i\ and\ T_d$.

3. Overview of Particle Swarm Optimization

Particle swarm optimization (PSO) is a population based algorithm and it is collection of particles move in steps of regions and its developed by Dr. Eberhart and Dr. Kennedy in 1995, inspired by social behavior of bird flocking or fish schooling [10].

The features of the PSO are as follows

- The method is based on searching on groups or swarm such as fish schooling and bird flocking.
- PSO is easy to implement as algorithm which is written in a very few lines of code and there are few parameters to adjust.
- PSO learned from the scenario and its used to solve the optimization problems

PSO is a method used for optimization of continuous nonlinear functions. According to the research results for the bird flocking, birds find the foods by flocking not by each individual. Each particle keeps track of its coordinates in the space, which are associated with the best solution. This value is called pbest. Another best value that is tracked by the global version of the particle swarm optimizer is the overall best value, and its location obtained so far by any particle in the group, is called gbest.

The PSO concept is, at each time step, changing the velocity of each particle toward its pbest and gbest location.

The modified velocity of each agent can be calculated using the current velocity and the distance from pbest and gbest as shown below

$$v_i^{k+1} = w_i v_i^k + c_1 rand \times (pbest - s_i^k) + c_2 rand \times (gbest - s_i^k) \quad (9)$$

Where,

v_i^k : Current velocity of particle i at iteration k,

v_i^{k+1} : Modified velocity of particle i,

$rand$: random number between 0 and 1,

s_i^k : current position of particle i at iteration k,

pbest : pbest of particle i,

gbest : gbest of particle i,

w_i : Weight function for velocity of agent i,

c_i : Weight coefficient.

Using the above equation, a certain velocity that gradually gets close to pbest and gbest can be calculated. The current position (searching point in the solution space) can be modified by the following equation.

$$s_i^{k+1} = s_i^k + v_i^{k+1} \quad (10)$$

Figure (1) shows the concepts of modification of a searching point by PSO.

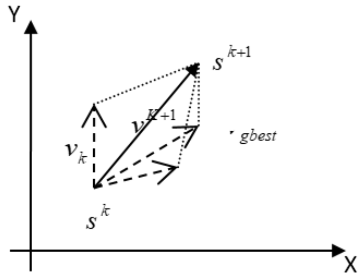

Fig. (1): Concept of modification of a searching point.

ISSN 2522-6967

s^k : current searching point,

s^{k+1} : modified searching point,

v^k : current velocity,

v^{k+1} : modified velocity,

v_{pbest} : velocity based on pbest,

v_{gbest} : velocity based on gbest.

Figure (2) shows a searching concept with agent in a solution space.

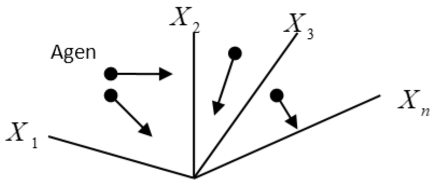

Fig. (2): Searching concept with agent in a solution space by PSO.

The algorithm of PSO

The proposed algorithm of PSO for searching the optimal PID-FPSS gains as follow

1. Initialize the swarm with random positions and velocities.
2. Calculate the fitness function of each particle by:
   $\frac{1}{2} sum(y_{sp} - y_s)^2 t$ (11)
   In this paper (y_{sp}) is actual speed set point and (y_s) is actual speed measurement.
3. Determine the $pbest$ and the $gbest$ positions.
4. Update the velocity of particle using Eq. (9).
5. Update the position of particle using Eq. (10).
6. If the evaluation value of each particle is better than the previous$pbest$, the value is set to$gbest$. If the best $pbest$ is better than$gbest$, the value is set to$gbest$.
7. If the iteration number reaches the maximum iteration number, then go to step 8. Otherwise, go to step 2.

Plotest , $gbest$ and compute the Eq. (11) for the control candidates with optimal values of PID-FPSS gains.

4. Design of Velocity PID-FPSS

The control problem is to design velocity PID controller in eq. (8) as fuzzy logic controller. Velocity PID controller can be converted into a fuzzy controller in a nonlinear manner to enhance robustness.

Consider fuzzy PID controller with three inputs as ($\Delta\omega, \Delta\dot{\omega}$ &ΔP_e) and out put (Δu).

Where:

$\Delta\omega$: Speed deviation (actual speed deviation – actual speed measurement).

$\Delta\dot{\omega}$: Derivative of speed deviation.

ΔP_e: Electrical power deviation.

Δu: Control action

The proposed PID-FPSS is designed as follow [11]:

1. Identification of inputs and output variables as ($\Delta\omega, \Delta\dot{\omega}$, ΔP_e& Δu).
2. Select the membership functions for the inputs and outputs as in Fig. (3).
3. Construction the fuzzy control rules as in table (1).
4. Selection of the compositional rule of inference.
5. Defuzzification method, transformation of fuzzy control set into fuzzy control action.

The center of area method is used for defuzzification. Gaussian membership function to represent the inputs/ output linguistic variables. For each inputs/ output variables, three labels are defined as shown in Fig. (3).

Table 1. Fuzzy rule base of PID-FPSS.

$\Delta\omega$ / $\Delta\dot{\omega}$	*N*	*Z*	*P*
N	*N*	*N*	*Z*
Z	*N*	*Z*	*P*
P	*Z*	*P*	*P*

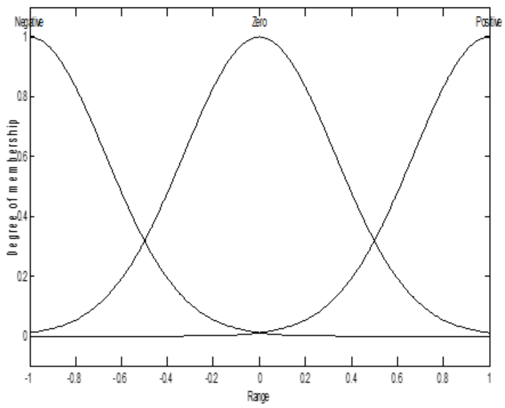

Fig. (3): Fuzzy variable, X_i, three-membership function.

N, Z and P stand for negative, zero and positive. The value X_{max} and X_{min} represent maximum and minimum variation of the input and output signals. The values are selected based on simulation information. A decision table is constructed consisting of 9 rules. An example of the i^{th} rule is:

If Δ ω is N and Δ *P* is P then U is Z

A symmetrical fuzzy rule set is used to describe the PID-FPSS behavior as shown in Table 1.

The procedure to design a FLC can be found in [1].

The PID-FPSS parameters are getting by particle swarm optimization (PSO) are:

$$K_p = 7 \quad ; T_i = 0.32 \quad ; T_d = 0.12$$

5. Implementation to Single Machine Infinite-Bus Model

Figure (4) shows the system under study. It comprises a single machine connected to an infinite bus through a tie line. The machine is equipped with a static exciter. The system is represented by the block diagram proposed by deMello and Concordia [12]. The generator is represented by third-order nonlinear model and the static exciter is represented by a first-order model. The nonlinear equations of the machine are given in [13].

6. Simulation Results

The PSS that is used for comparison is a simple robust power system stabilizer (RPSS) is designed in [13] that can

ISSN 2522-6967

properly function over a wide range of operating conditions and extend the machine load ability. The lead compensator design is achieved by drawing the root loci for a finite number of extreme characteristics polynomials. Such polynomials are obtained, using the kharitonov theorem [13], to reflect wide loading conditions on characteristics equation coefficient.

For a comparison purpose, the system is configured to switch between two control techniques. In order to show the response of PID-FPSS and RPSS as shown in Fig. (4).

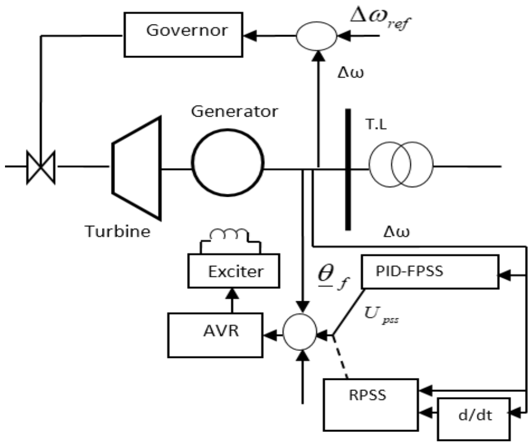

Fig. (3): Fuzzy variable, X_i, three-membership function

To investigate the power system performance. The system responses to a three-phase fault at t = 0 sec. and cleared after 100ms, will show the response of the system for two operating conditions.

The performance of each control law was measured by the performance index as in eq. (11)

The selected loading conditions are as follows:

a) ***Operating Condition 1***: Electrical Power ($P_e = 0.8\,pu$).
b) ***Operating Condition 2***: Electrical Power ($P_e = 0.3\,pu$)

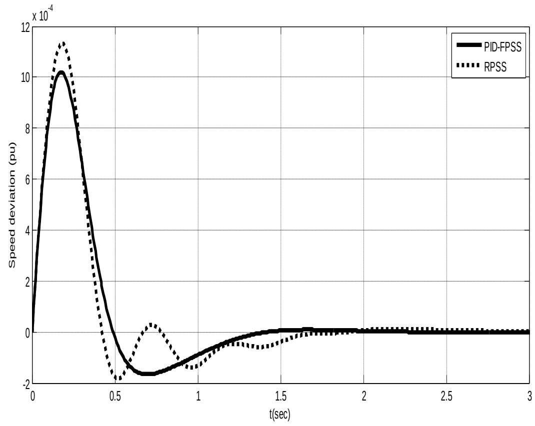

Fig. (5): Speed deviation response at ($P_e = 0.8\,pu$).

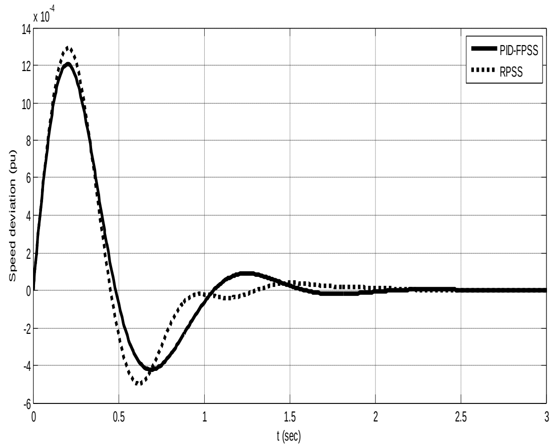

Fig. (6): speed deviation response at ($P_e = 0.3\,pu$).

TABLE 2. The performance index of ($\Delta\omega$), at operating condition 1**.**

Controller	Speed Deviation
PID-Fuzzy PSS	225
Robust PSS	256

TABLE 3. The performance index of ($\Delta\omega$), at operating condition 2.

Controller	Speed Deviation
PID-Fuzzy PSS	450
Robust PSS	501

7. Conclusion

PID-FPSS technique was designed based velocity PID controller and its gains are tuned by particle swarm optimization technique.

Fuzzy Power system stabilizer was developed based on linguistic rules of fuzzy basis function. The performance of the proposed stabilizer was investigated and compared with the robust PSS was designed in [13]. Through the dynamic simulations of a single machine infinite bus power system, the simulations form Fig. (5), Fig. (6), Table (2) and Table (3) have shown that a significant improvement can be achieved with PID-FPSS for a two different operating points.

References

[1] W. Watson and Manchur, "Experience with supplementary damping signals for generator excitation systems" IEEE Trans. on power apparatus and systems, vol PAS–1973, PP. 193-203.
[2] F. P. deMello, L. N. Hannett and M. Undrill "Practical approaches to supplementary stabilizing from accelerating power" IEEE Trans. On power apparatus and systems, vol PAS-1978, PP. 1515-1522.
[3] H. Joe Chow, E. G. Boukarim and A. Murdoch "Power system stabilizer as undergraduate control design project" IEEE Trans. On power apparatus and systems, vol PAS-2004, PP. 144-151.
[4] G. Y. Rajaa Vikhram, S. Latha " Design of power system stabilizer for power system damping improvement with multiple design requirements" international journal of soft computing and Engineering (IJSCE) ISSN: 2231-23077, vol. 2, 2012.
[5] M. Megala, Rajan, C. Cristober Asir "Design of fuzzy logic power system stabilizer in multi-machine power system"

ISSN 2522-6967

proceedings of the international conference on frontiers of intelligent computing: theory and application 2013.

[6] P. Kripakaran, M. Vigneshan, S. Asvnth, B. Sivachandiran " performance of power system stabilizer using fuzzy logic controller" journal of electrical and electronic engineering (JOSR-JEEE) , vol. 9. 2014, pp. 42-49.

[7] D. K. Sambariya, R. Prasad " Design of robust PID power system stabilizer for multimachine power system using HS algorithm" American Journal of Electrical and Electronic Engineeringvol. 3, No. 3, 2015, pp 75-82.

[11] John H. “Fuzzy control and identification" published by John Wiley & Sons, 2010.

[12] F. P. Demello, C. Concordia " Concepts of Synchronous Machine Stability as Affected Excitation Control" IEEE

[8] X. Li, Z. Wang, Junhua Xu, and B. Chen "Power system stabilizer parameters designing based on genetic simulated annealing algorithm" JOCET 2016 vol.4, No.3, 2016, pp 178-182.

[9] Haider A. and Sulochana W. "Intelligent PID Controller Tuning for Higher Order Process System" International Journal of u- and e- Service, Science and Technology Vol.8, No. 6, 2015, pp.323-330.

[10] T. Hussein and A. Shamekh “Performance Assessment of Fuzzy Logic Power System Stabilizer on North Benghazi Power Plant" Hindawi Publishing Corporationn Conference Papers in Engineering , 2013.

Transactions on Power Apparatus and Systems, vol. pas-88, no. , pp 316-328. 1969.

[13] H. M. Soliman, A. L. Elshafei, A. A. Shaltout and M. F. Morsi " Robust Power System Stabilizer" IEE Proceeding on Electrical Power Aplication, vol 14, pp 285-291.