International Journal of Scientific & Engineering Research, Volume 7, Issue 3, March-2016
ISSN 2229-5518

Automatic tuning of (PID) and Fuzzy controllers Using (PSO) Algorithm for Hybrid energy Systems

Rana Mohamed Fathy, Mohamed Ahmed Ebrahim, Fahmy Metwaly Bendary

Abstract— In order to meet the increasing energy demand, alternate energy resource has been developed. Because of renewable energies are unavailable in all times, hybrid system is preferred to make sure that electricity feeding loads all time and for reliability. In this paper hybrid system contains wind farm and hydro power plant. Simulation is carried out using Mat lab/Simulink. The simulation results show that the proposed controllers have remarkable effects on the system. Advanced controllers are used to track max power point under different conditions.

Index Terms— wind, hydropower, control systems, proportional-integral-derivative (PID) controller, Fuzzy, PSO

1 Introduction

IN the past few decades, renewable energy has been widely developed due to a tremendous increase in energy demand. Hybrid system of renewable energy is the most popular due to continuity of energy which can be offered to loads. Wind power is clean, low cost and plentiful on the earth. In early days, large farms used wind power generated by windmills to do farm works. Recently, advanced technology enables turbines to generated large electrical power to meet the energy demand [7].

Hydropower has now become one of the best sources of electricity on earth. It is produced due to the energy provided by moving or falling water. Most of the countries now have hydropower as the source of major electricity producer. The most important advantage of wind and hydropower is that they are green energy [9], which mean that no air or water pollutants are produced, also no greenhouse gases like carbon dioxide are produced which make these sources of energy environment-friendly. They prevent us from the danger of global warming [8].

Due to wind variation and any sudden conditions or disturbance on this hybrid system there is a need to track maximum power point and get maximum power under different conditions

The Proportional-Integral-Derivative (PID) controllers are the most popular controllers in industry because of their effectiveness and simplicity, and fuzzy –logic controllers providing a high value unified response to distribution network customer's needs

This paper presents an artificial intelligence (AI) methods of particle swarm optimization (PSO) algorithm for tuning the optimal (PID) and (Fuzzy) controller's parameters and compare between two controllers to determine best values of controllers parameters to reach better o/p power for the hybrid system. Simulation results are presented to show that PSO-based fuzzy is capable of providing performance over PID controller. Fuzzy logic controller has better

Stability small overshoot and fast response . PSO is more efficient in improving the step response characteristics such as reducing the

- Rana Mohamed Fathy,Shoubra Faculty of Engineering,Benha University, Cairo,Egypt,ranafathy91@yahoo.com
- Mohamed Ahmed Ebrahim ,Shoubra faculty Of Engineering ,Benha University, Cairo –Egypt,Mohamedahmed_en@yahoo.com
- Fahmy Metwaly Bendary, Shoubra Faculty of Engineering ,Benha
- University, Cairo-Egypt ,fahmybendary10@gmail.com

steady –state error, rise time, settling time and max over shoot.

We use controllers in the system to improve the system performance, and enhance its stability, they are also used to reach the optimum value of bitch angle β to track max o/p power in wind farm under any operating conditions or wind variation. On the other hand these controllers are adopted to control the voltage and frequency of the hydro system as well as to enhance system stability and the rotor speed of the hydro turbine and the value of excitation voltage to get the best o/p of power.

2 CONTROL STRATEGIES TO GET THE BEST VALUES FOR CONTROLLER'S PARAMETERS PID AND FUZZY

2.1 Overview of particle swarm optimization (PSO) algorithm

Theory of particle swarm optimization (PSO) has been growing rapidly. PSO has been used by many applications of several problems. The algorithm of PSO emulates from behavior of animals societies that don't have any leader in their group or swarm, such as bird flocking and fish schooling. Typically, a flock of animals that have no leaders will find food by random, follow one of the members of the group that has the closest position with a food source (potential solution). The flocks achieve their best condition simultaneously through communication among members who already have a better situation. Animal which has a better condition will inform it to its flocks and the others will move simultaneously to that place. This would happen repeatedly until the best conditions or a food source is discovered. The process of PSO algorithm in finding optimal values follows the work of this animal society. Particle swarm optimization consists of a swarm of particles, where particle represents a potential solution.

The particles of the swarm fly through hyperspace and have two essential reasoning capabilities: their memory of their own best position – local best (lb) and knowledge of the global or their neighborhood's best – global best (gb). Position of the particle is influenced by velocity. let $X_i(t)$ denote the position of particle (i) in the search space at time step; unless otherwise stated, t denotes discrete time steps. The position of the particle is changed by adding a velocity $V_i(t)$, to the current position.

$$X_i(t+1) = V_i(t) + V_i(t+1) \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots.. (1)$$

Where

$$V_i(t) = V_i(t-1) + c_1 r_1\big(local\ best(1) - X_i(t-1)\big)$$

International Journal of Scientific & Engineering Research Volume 7, Issue 3, March-2016
ISSN 2229-5518

$$+c_2 r_2(global\ best(t) - X_i(t-1))$$
………………………………………………………………. (2)

Where $V(t)$ stands for velocity
Acceleration coefficient $c_1 and\ c_2$
i is the generation number, $c_1 = c_2 = 2$,
However, after a number of iterations the best positions are almost similar therefore, the step –size vanishes to zero once it gets to the optimum value, which means that there will be no oscillations around the peak point

2.3 OVERVIEW OF PROPORTIONAL-INTERGRAL-DERIVATIVE (PID)

The proportional Controller

The proportional term produces an output value that is proportional to the current error value. The proportional response can be adjusted by multiplying the error by a constant Kp, called the proportional gain constant. If the proportional gain is too high, the system can become unstable. If the proportional gain is too low, the control action may be too small when responding to system disturbances. Tuning theory and industrial practice indicate that the proportional term should contribute the bulk of the output change.

The proportional term is given by:

$P_{out} = K_p e(t)$ …………………………………………….. (3)

The Integral Controller

The integral in a PID controller is the sum of the instantaneous error over time and gives the accumulated offset that should have been corrected previously. The accumulated error is then multiplied by the integral gain and added to the controller output. The integral term accelerates the movement of the process towards set point and eliminates the residual steady-state error that occurs with a pure proportional controller. However, since the integral term responds to accumulated errors from the past, it can cause the present value to overshoot the set point value.

The integral term is given by:

$I_{out} = K_i \int_0^t e(t)dt$ …………………………………………… (4)

The Derivative Controller

Derivative control is used to reduce the magnitude of the overshoot produced by the integral component and improve the combined controller-process stability. However, the derivative term slows the transient response of the controller.

The derivative term is given by:
$D_{out} = K_d\ d/dt\ e(t)$ …………………………………………… (5)

Controller	Rise time	Over shoot	Settling time
Kp	Decrease	Increase	Small change
Ki	Decrease	Increase	Increase
Kd	Small change	Decrease	Decrease

Table (1) Characteristic of P, I and D controllers

$$u(t) = K_p e(t) + K_i \int_0^{\tau} e(\tau)dt + K_d\ de(t)/dt$$
………………………………… ………………….. (6)

Where

u(t) : controller output , K_P: Proportional gain, a tuning parameter , K_i: Integral gain, a tuning parameter , K_d: Derivative gain, a tuning parameter , e: Error , t: Time or instantaneous time (the present) , τ: Variable of integration; takes on values from time 0 to the present τ.

Equivalently, the transfer function in the Laplace domain of the PID controller is

$$G_c(s) = K_p + K_i/s + K_d s$$
………………………………………………….. (7)

2.4 OVERVIEW OF FUZZY CONTROLLER

The basic configuration of a pure fuzzy –logic controller is composed of four parts
The (fuzzification) is the process of mapping the i/ps crisp values into fuzzy variables using normalized membership functions and i/p gains
The fuzzy logic inference engine: deduces the proper control action based on the available rule base
The fuzzy control action is transferred to the proper crisp value through the defuzzification process using normalized membership functions and o/p gains.

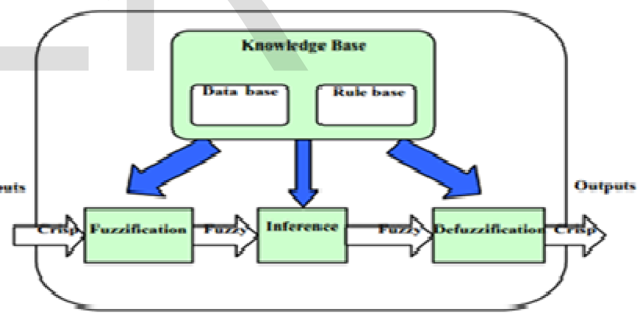

Fig (1) The basic structure of the fuzzy controller

The speed deviation (Δω) & the deviation in accelerating power (Δp) are chosen as inputs, the output control signal from FLPSS is Upss as shown in table (2)

International Journal of Scientific & Engineering Research Volume 7, Issue 3, March-2016
ISSN 2229-5518

Fuzzy-logic PSS rules

		ΔP						
		NB	NM	NS	Z	PS	PM	PB
Δω	NB	NB	NB	NB	NB	NM	NS	Z
	NM	NB	NB	NM	NM	NS	Z	PS
	NS	NB	NM	NM	NS	Z	PS	PM
	Z	NM	NM	NS	Z	PS	PM	PM
	PS	NM	NS	Z	PS	PM	PM	PB
	PM	NS	Z	PS	PM	PM	PB	PB
	PB	Z	PS	PM	PB	PB	PB	PB

Table (2)

Where N, Z AND P stand for negative, zero, positive
B, M AND S stand for big, medium, and small
For example: NB means negative big
Let Xmax = - Xmin, the range of each fuzzy variable is normalized between (-1:+1) by introducing a scaling factor to represent the actual signal
The membership functions NB,NM,NS,Z,PS,PM,PB will have their centroids at (-1,-2/3,-1/3,0,1/3,2/3,1)

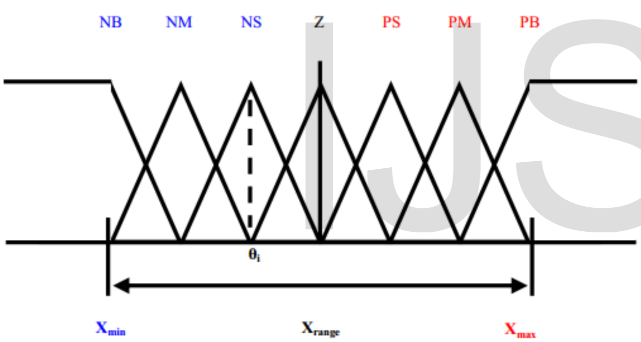

Fig (2) Fuzzy membership functions of 7*7 (49 rules)

3 SYSTEM UNDER STUDY IS AHYBRID SYSTEM CONTAINS WIND FARM AND HYDRO POWER PLANT

3.1 Wind farm model and simulation:

A wind farm consisting of six 1.5-MW wind turbines is connected to a 25-kV distribution system exports power to a 120-kV grid through a 25-km 25-kV feeder. The 9-MW wind farm is simulated by three pairs of 1.5 MW wind-turbines. Wind turbines use squirrel-cage induction generators (IG).

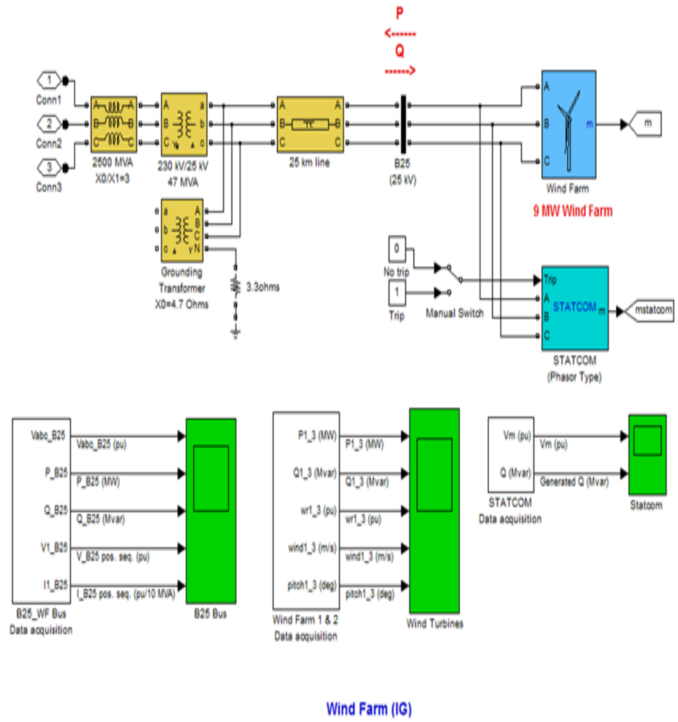

Fig (3) wind farm under study

The rotor is driven by a variable-pitch wind turbine. The pitch angle is controlled in order to limit the generator output power at its nominal value for winds exceeding the nominal speed (9 m/s). In order to generate power the IG speed must be slightly above the synchronous speed. Each wind turbine has a protection system monitoring voltage, current and machine speed

The simulation results by using PI controller KP=5; Ki=25; wind speed range from 8m/s to11 m/s (input to system as step function it's initial value 8 and the final value 11)

Where output power by Megawatts for 2 turbines against time by seconds

International Journal of Scientific & Engineering Research Volume 7, Issue 3, March-2016
ISSN 2229-5518

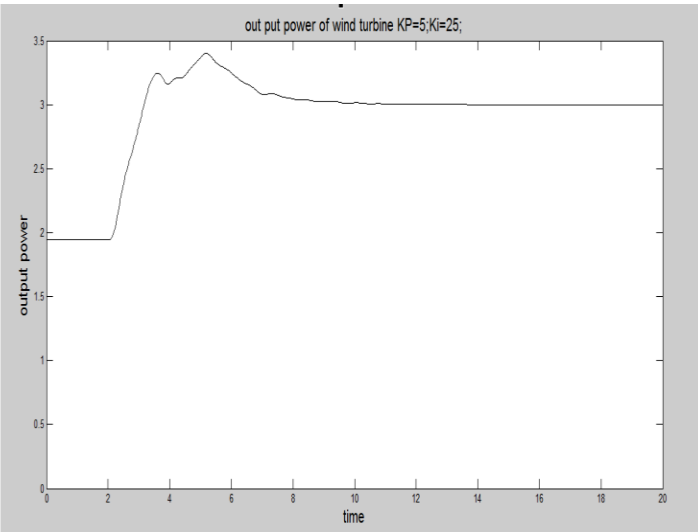

Fig (4) Output power for wind turbine with (PI) controller

Adjust the initial values for the system or to make any changes by these steps:

1. In the Simulation/Configuration Parameters/Data Import/Export Parameters menu, uncheck the "Initial state" parameter.
2. Open the "Wind Farm" subsystem and in the Timer blocks labeled "Wind1" and "Wind2", Wind3" temporarily disable the changes of wind speed by multiplying the "Time(s)" vector by 100.
3. In the "Wind Farm" subsystem, double click on the "Three-Phase Fault" block and disable the AB to ground fault (deselect "Phase A Fault" and "Phase B Fault").
4. Start simulation. When Simulation is completed, verify that steady state has been reached by looking at waveforms displayed on the scopes. The final states which have been saved in the "xFinal" array can be used as initial states for future simulations. Executing the next two commands copies these final conditions in "xInitial" and saves this variable in a new file (myModel_init.mat).
5. >> xInitial=xFinal;
6. >> save myModel_init xInitial
7. In the File/Model Properties/Callbacks/InitFcn window, change the name of the initialization file from "power_wind_ig_xinit" to "myModel_init". Next time you open this model, the variable xInitial saved in the myModel_init.mat file will be loaded in your workspace.
8. In the Simulation/Configuration Parameters menu, check "Initial state".
9. Start simulation and verify that your model starts in steady-state.
10. Open the "Wind Farm" subsystem and in the Timer blocks labeled "Wind1", "Wind2" and Wind3" re-enable the changes of wind speed respectively a t=2 s , t=4 s and t=6 s (remove the 100 multiplication factors).
11. In the "Wind Farm" subsystem, re-enable the AB to ground fault in the "Three-Phase Fault" block (check "Phase A Fault" and "Phase B Fault")
12. Save your Model.

The hydropower plant:

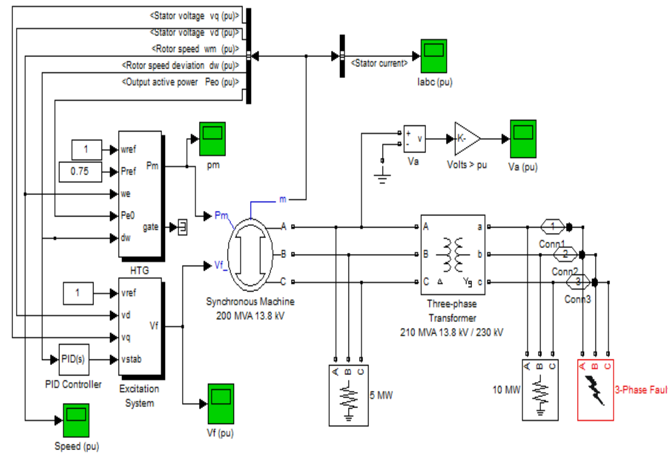

Fig (5) Hydro power plant under study

3.2 Hydro plant model and simulation:

A three-phase generator rated 200 MVA .At t = 0.1 s, a three-phase to ground fault occurs on the 230 kV bus. The fault is cleared after 6 cycles (t = 0.2 s). During this demo, you will initialize the system in order to start in steady-state with the generator supplying 150 MW of active power and observe the dynamic response of the machine and of its voltage and speed regulators.

Hydro power plant mainly consists of three sections, ‘governor (controller), hydro servo system and hydro turbine’. Integrally this section is known as hydro turbine governor which is coupled to a synchronous generator to drive the shaft so that the mechanical energy of turbine is converted to the electrical energy. This system is supplying power to a common electrical three phase parallel RLC load. Models are simulated on MATLAB/Simulink. The speed governing control system ensures the constant speed operation during variation of load

Adjust the initial value for the hydro plant for the system or to make any changes by these steps:

1. Start Simulation and observe the three machine currents on the Iabc scope. If the 9 parameters defining initial conditions for the Synchronous Machine are set at zero or not set correctly, the simulation will not start in steady state.

2. In order to start the simulation in steady-state you must initialize the synchronous machine for the desired load flow. Open the Powergui and select 'Load Flow and Machine Initialization'. A new window appears. The machine 'Bus type' should be already initialized as 'PV generator', indicating that the load flow will be

International Journal of Scientific & Engineering Research Volume 7, Issue 3, March-2016
ISSN 2229-5518

performed with the machine controlling the active power and its terminal voltage. Specify the desired values by entering the following parameters:

Load flow: Terminal voltage (Vrms) = 13800; Active Power = 150e6. Then press the 'Update Load Flow' button.

Once the load flow has been solved the phasors of AB and BC machine voltages as well as the currents flowing out of phases A and B are updated. The machine reactive power, mechanical power and field voltage requested to supply the electrical power should also be displayed: Q = 3.4 Mvar; Pmec = 150.32 MW (0.7516 pu); field voltage Ef = 1.291 pu.

3. In order to start the simulation in steady state with the HTG and excitation system connected, these two Simulink® blocks must also be initialized according to the values calculated by the load flow. This initialization is automatically performed when you execute the Load Flow, as long as you connect at the Pm and Vf inputs of the machine either Constant blocks or regulation blocks from the machine library (HTG, STG, or Excitation System). Open the HTG block menu and notice that the initial mechanical power has been automatically set to 0.7516 pu (150.32 MW) by the Load Flow. Then, open the Excitation System block menu and note that the initial terminal voltage and field voltage have been set respectively to 1.0 and 1.1291 pu.

4. Open the 4 scopes and restart the simulation. The simulation now starts in steady state. Observe that the terminal voltage Va is 1.0 p.u. at the beginning of the simulation. It falls to about 0.4 pu during the fault and returns to nominal quickly after the fault is cleared. This quick response in terminal voltage is due to the fact that the Excitation System output Vf can go as high as 11.5 pu which it does during the fault. The speed of the machine increases to 1.01 pu during the fault then it oscillates around 1 p.u. as the governor system regulates it. The speed takes much longer than the terminal voltage to stabilize mainly because the rate of valve opening/closing in the governor system is limited to 0.1 pu/s.

The simulation results of PID controller (the speed governing control system):- "rotor speed (p.u)" at 112.5 rpm (in case of 3 phase to ground fault)

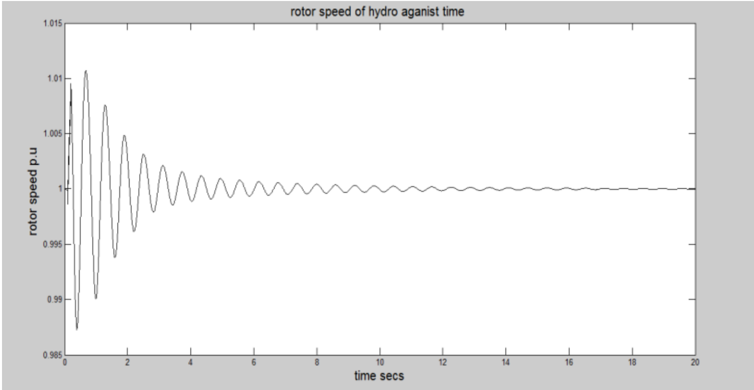

Fig (6) Rotor speed for hydro turbine with (PID) "governor" controller

3.3 The hybrid system between wind and hydro connected to grid:

The hybrid system connected to a 230 kV, 10,000 MVA network through a Delta-Wye transformers and frequency adjust for system at 60 hertz

Under operating conditions: At t = 0.1 s, a three-phase to ground fault occurs on the 230 kV bus. The fault is cleared after 6 cycles (t = 0.2 s). And the wind change from 8 m/s to 11 m/s.

Case study:

3 phase to ground fault on the system from 0.1 to 0.2 seconds and the wind speed variation from 8 m/s to 11 m/s. Track to keep speed constant and get best values for controller's parameters. The simulation results and comparing between PID and fuzzy 7*7 controllers as following

4 Simulation results for hybrid system by using particle swarm optimization

4.1 By using particle swarm optimization (PSO) to make automatic tuning for getting parameters of (PID) controllers best values

Run no	7
K_{p_1}	21.096
K_{p_2}	4.7701
K_p	1.8253
K_{i_1}	8.6539
K_{i_2}	2.5381
K_i	15.457
K_{d_1}	12.662
K_{d_2}	2.1535
K_d	10.802

International Journal of Scientific & Engineering Research Volume 7, Issue 3, March-2016
ISSN 2229-5518

Table (2) best values for PID parameters

As shown in table (2)

Where: $K_{p_1}, K_{i_1}, K_{d_1}$ $controller's\ parameters\ of\ hydropower\ plant\ controller\ for\ exciter$

$K_{p_2}, K_{i_2}, K_{d_2}$: $controller's\ parameter of\ hydropower\ plant\ controller\ for\ hydroturbine\ (governer\ controlling\ system\ )$

K_p, K_i, K_d

: $controller's\ parameter of\ wind\ farm\ controller\ for\ turbines$

4.2 By using particle swarm optimization (PSO) to make automatic tuning for getting parameters of (Fuzzy) controllers best values

Run no	3
K_{p_1}	13.239
K_{p_2}	17.989
K_p	19.658
K_{i_1}	22.842
K_{i_2}	10.299
K_i	15.431
K_{d_1}	6.8563
K_{d_2}	14.809
K_d	15.239

Table (3) best values for Fuzzy parameters

Where: $K_{p_1}, K_{i_1}, K_{d_1}$ $controller's\ parameters\ of\ hydropower\ plant\ controller\ for\ exciter$

$K_{p_2}, K_{i_2}, K_{d_2}$: $controller's\ parameter of\ hydropower\ plant\ controller\ for\ hydroturbine\ (governer\ controlling\ system\ )$

K_p, K_i, K_d

: $controller's\ parameter of\ wind\ farm\ controller\ for\ turbines$

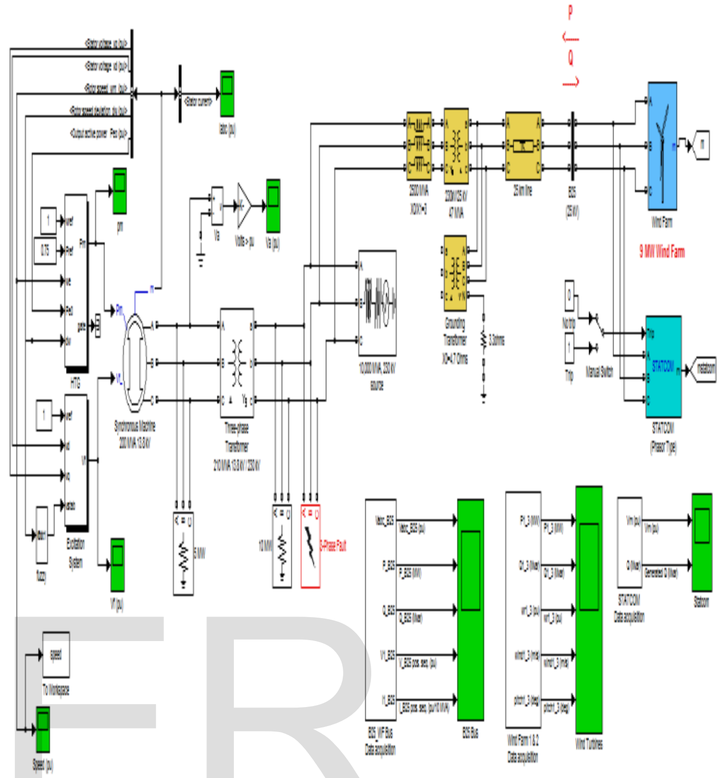

Fig (7) the hybrid system to grid

International Journal of Scientific & Engineering Research Volume 7, Issue 3, March-2016
ISSN 2229-5518

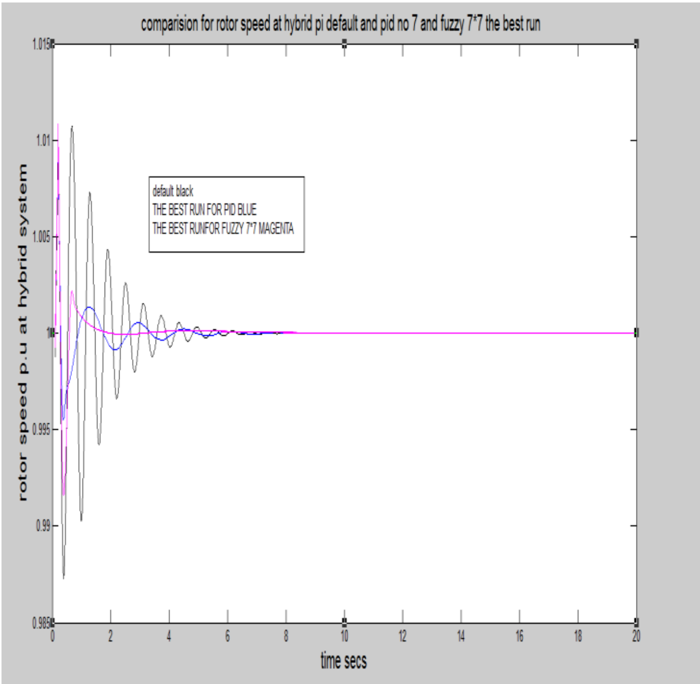

Fig (8) rotor speed responses using different controllers by PID controller's parameters and (PID, Fuzzy) with PSO

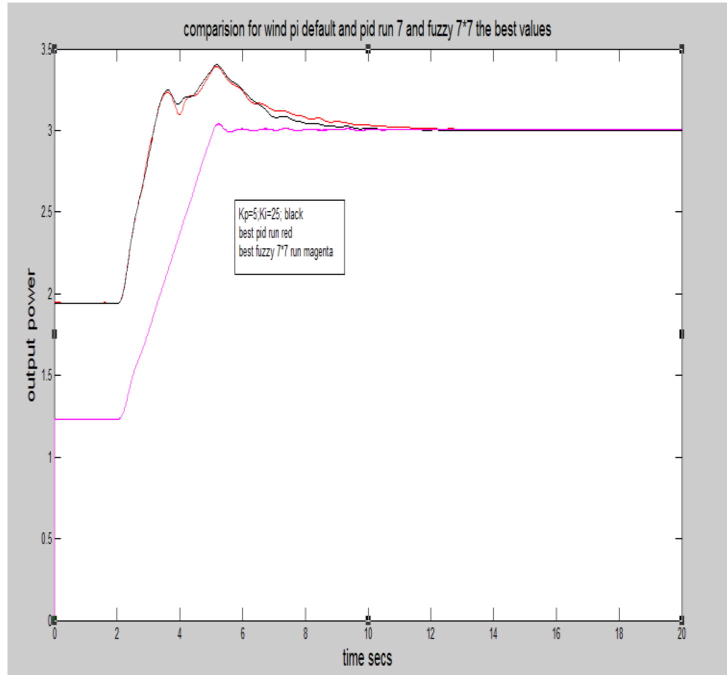

Fig (9) output power of wind turbine (PID, fuzzy) controller's by PSO automatic tuning and PID without PSO

As clear in fig (9) PID controller (governor) is used to control the servo motor which controls the amount of water falling on the hydro turbine (the energy is provided by moving or falling water) which effects on the speed of hydro turbine and the output power also. Fuzzy is used to make the same operation, the main target of controlling the servo motor and the amount of falling water to keep system stable by keeping the rotor speed of hydro turbine stable By comparing the results and the speed curve against time it is clear that using (fuzzy controller is better than PID controller) more stable and less oscillation. In fig (8) PID controller used to eliminate the error (difference between power measured from turbine and the reference value of power) to track the max power point

5 CONCLUSION

The renewable energy sources, hydro and wind energy have the capability to complement each other. The grid connected system of wind turbines and hydro turbine ensure the stable operation of the system. A new wind-hydro generating station, a case study for automatic tuning for controller parameters by using particle swarm optimization for hybrid system to track maximum power point and reduce oscillations . system connected to grid has been modeled and simulated in MATLAB/ Simulink. The proposed system ensures the stable operation of hybrid system connected to the grid and also generates maximum power when the wind speed increases or sudden loading sudden load rejection.

6 Acknowledgments

I would like to thank to the doctors supervising this research.

7 References

.

REFERENCES

[1] Elgerd OI. , "Electric energy systems theory: An introduction", Mc Graw- Hill; 1971

[2] Prabha Kundur," Power System Stability and Control " ,Mc Graw – Hill, Inc, USA, 1994.

[3] Hadi saadat , " Power System Analysis " , Tata Mc Graw – Hill, edition ,2002.

[4] K. J. Astrom, "Control System Design", 2002.

[5] J. Perng, G. Chen, S. Hsieh, "Optimal PID Controller Design Based on PSO- RBFNN for Wind Turbine Systems", Energies 2014

[6] Yue Hou , wind turbine control systems analysis , introduction to PID controller design ,2014

[7] http://www.juwi.com/wind_energy/references.html

[8] http://greenliving.lovetoknow.com/Advantages_and_Disadvantages_of_Non_Renewable_Energy

[9] Dursun, G. Cihan, The role of hydroelectric power and

International Journal of Scientific & Engineering Research Volume 7, Issue 3, March-2016
ISSN 2229-5518

contribution of small hydropower plants for sustainable development in Turkey. Renew. Energ. 36(4), 1227–1235 (2011)

[10] http://www.mathworks.com/?refresh=true

[11] http://www.mathworks.com/help/physmod/sps/examples/wind-farm-ig.html?refresh=true

[12] http://www.mathworks.com/help/physmod/sps/examples/synchronous-machine.html?searchHighlight=synchronous%20machine

[13] L. Wang, "Adaptive Fuzzy System and Control", Prentice-Hall, New Jersey, 1994.