CHAPTER 3

CONTROLLER DESIGN OF DFIG FOR WECS USING PSO TECHNIQUE

3.1 An overview of Wind Energy Conversion System

The renewable assets are valuable, and also obtainable throughout the earth. Shifting to renewable possessions in addition to financial benefits can obtain other benefits such as clean environment along with less weather change by means of declining greenhouse gas release [4]. However, the progressed wind turbines accessibility is too much 98% in a stormy area with capacity factors 35–40%. The electrical energy production by wind technology has tinted first time during the 1970s due to the oil crisis. The worldwide fashion toward clean energy is a motivation for supplementary integration of wind-based power in the system [5]. However, the wind velocity changes radically depending on the environmental circumstances along with the time of operation. Therefore, it has a huge margin of the speed difference. Such margins of speed alteration compose wound rotor induction machines appropriate for power generation through wind energy. The wind turbines can run either fixed speed (actually within a speed range about 1%) or varying speed [122], and wind turbine aerodynamic model has been characterized by the well-known C_P (λ, β) curves [123]. For the given power coefficient C_P, the mechanical power that the wind turbine extracts from the wind is calculated as follows;

$$P_m = \frac{1}{2}\rho A v_w^3 C_P(\lambda,\ \beta) \tag{1}$$

3.2 DFIG principle of operation along with modelling

A part of the wound rotor induction generator (recognized as the DFIG), which is the most frequently employed generator in WECS [35]. Currently, such kind of generator is widely established as appropriate wind power alteration scheme as is described in Figure (3.1) [35].

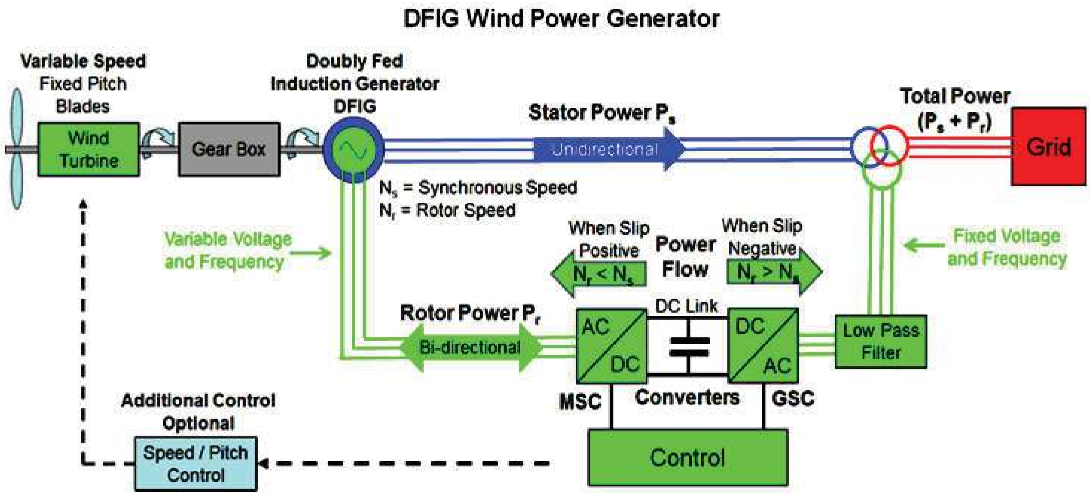

Figure 3.1: DFIG based wind turbine system typical mechanism illustration

DFIG is the wound rotor induction generator in nature. Its rotor circuit is usually insured with electric control strategy to allow changeable speed process [35]. However, the DFIG stator winding is associated with the grid by a power transformer and DFIG's power often, from a few kilowatts to several megawatts. The size of rotor side converter is nearly 30% from the full capacity converter. During low wind speed, more electrical energy might be obtained from the variable speed DFIG system, in distinguishing using a fixed speed wind generator [37]. Finally, the enhancement of voltage profile in a remote location to the DFIG in WECS has been described in [124]. Whereas in [36], DFIGs have two operating modes; in (i) mode Nr>Ns, s is –ve, then generator in super-synchronous mode and both stators, as well as rotor windings, convey power to the grid. While in (ii) mode Nr<Ns, s is +ve the generator in subsynchronous mode and stator winding provides power to both the grid as well as rotor winding. The mathematical modeling of DFIG is explained clearly in chapter 2.

3.3 Voltage Source Converter Controller

The VSC is used to the DFIG to control rotor as well as grid side converter and DC link voltage with Modified System Model to Design of Supervisory PID Controller for the considered system[46].The explanation about voltage source controller has been presented in previous chapter 2. Whereas, supervisory based PID controller parameter has given in chapter 2 Table

(2.1), [44].

However, the Transfer function of the 6[th] order reduced model to the DFIG system is presented in [46].

$$G(S)=\frac{0.000324s6-1.75s5-2366s4\ +\ 7.9e6s3\ +\ 7.5e9s2\ +\ 5e12s\ +\ 2.18e14}{s6\ +\ 2340s5\ +\ 8.67e6s4\ +\ 4.79e9s3\ +\ 2.7e12s2\ +\ 1.27e14s\ +\ 9.6e14} \quad (2)$$

3.4 Particle Swarm Optimization

An overview and brief description for particle swarm optimization along with algorithm features described in [125-131]. In 1995 James Kennedy and American Social Psychologist together with Russell C. Eberhart invented an innovative progression computational method called as Particle Swarm Optimization (PSO) [125-126]. This approach is appropriate to solve the nonlinear dilemma. This process is established on the flock activities such as birds detect food using flocking. The first alternative of the PSO algorithm brings using having the inhabitants (named the swarm) for the applicant result (known as particles). These particles are turned in the search space using a few straightforward formulas. The motions of the particles are guided to their best-known position in the search space in addition to the entire swarm's most prestigious situation. Whenever an enhanced position is being detected, then it will move towards to guide the movement of the flock. The procedure is repetitive, furthermore using performing, so it is trusted, but not assured, that an adequate result will finally be detected. Here in this procedure, a set of particles is introduced in the d-dimensional search space using at random selecting velocity as well as position. The first situation of the particle is considered as the best place for the initiate, as well as the velocity of the particle is updated based on experience to the other particles from the swarming population.

3.4.1 Algorithmic rule for PSO

The i^{th} particle to swarm is symbolized as: $X_i = (x_{i1}, x_{i2}, x_{i3},.....x_{id})$ in the d-dimensional space.

The best former positions of the i^{th} particle are described at the same time as follows:

$P_{best} = (Pbest_{i,1}\ Pbest_{i,2}\ Pbest_{i,3}..........\ Pbest_{i,d})$.

The index of the best particle amongst the group is G_{bestd}.

The velocity of the i^{th} particle is presented as: $V_i = (V_{i,1}\ V_{i,2}\ V_{i,3} \ldots V_{i,d})$.

The updated velocity along with the distance from $Pbest_{i,d}$ to $Gbest_{i,d}$ is given as:

$$V_{i,m}^{t+1} = W*V_{i,m}^{t} + C_1*rand()*(Pbest_{i,m} - X_{i,m}^{t}) + C_2*rand()*(Gbest_{i,m} - X_{i,m}^{t})$$

Whereas $X_{i,m}^{(t+1)} = X_{i,m}^{(t)} + V_{i,m}^{(t+1)}$ For i=1,2,3.......n., and m = 1, 2, 3.....d.

Here, n= number of particles in group. d= dimension index.

t=pointer of iteration. $V_{i,m}^{(t)}$= Velocity of particle at iteration i.

W=inertia weight factor. C_1, C_2= acceleration constant.

rand ()= random number between 0 and 1.

$X_{i,d}^{(t)}$ =current position of the particle at 'i' iteration.

$Pbest_i$= best former position of the i^{th} particle.

G_{best}=best particle amongst the total particle in swarming population.

3.4.2 Algorithmic Approach for the PID controller design

Here, the PID controller designs into the PSO structure with three-dimensional search space K_P, K_I, and K_D; and to find fitness function based on time domain features for alteration. Furthermore set the numeral adjustment iterations, which is based on anticipated parameters along with the time of computation.

3.4.3 A minute Representation for algorithm

We establish the values of PSO algorithm constants as inertia weight factor W = 0.3, along with acceleration constants C_1, C_2 = 1.5. As we have to optimize three parameters that are; K_P, K_D, K_I to the controller, and search optimal value in the three-dimensional search space. So we arbitrarily formatted a swarm of "100" population in three-dimensional search space using $[X_{i,1}\ X_{i,2}\ X_{i,3}]$ and $[V_{i1}\ V_{i2}\ V_{i3}]$ as preliminary position as well as velocity. Considered primary fitness function of each point along with the position with minimum fitness function is exhibited as g_{best}

(initial value of global best Optima) as well as the optimal fitness function as f_{best1} (Initial best fitness function). Run the program by means of PSO algorithm through thousands (or yet additional numbers) of iterations; in addition to program brought back the final optimal value of fitness function as "f_{best}" along with last global optimum point as "G_{best}." However, PSO algorithm procedure as flow chart is shown in Figure (3.2) [131].

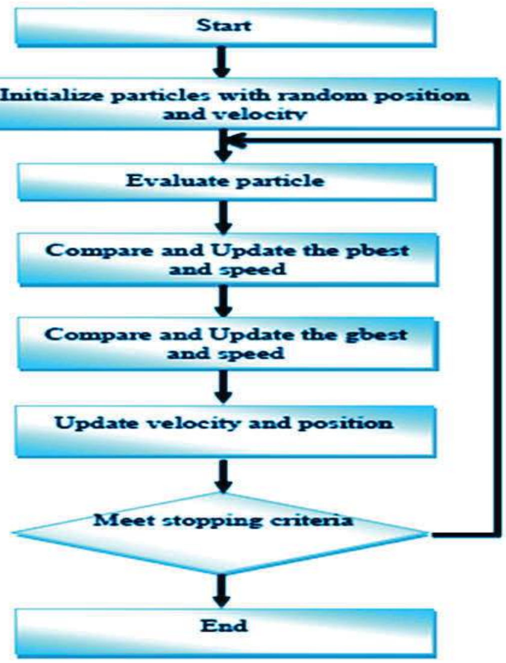

Figure 3.2: PSO algorithm process as flowchart

3.4.4 The conception of fitness function to the controller design

For best results to optimize all parameters of the PID and describe a three-dimensional search space in which all three dimensions present three different parameters of the PID. Every precise point in search space accounts for a particular arrangement of [K_P, K_I, K_D]. For which an appropriate answer has found, and a fitness function determines the performance of the position to the PID parameters. This fitness function comprises numerous essential features to the performance index for design. Along with the best position in search space, the fitness function reaches an optimal value with four necessary functions which defined fitness function. By this feature; steady state errors, peak overshoot, rise time as well as settling time has been come out for the described system. However, the contribution of these essential functions towards the innovative fitness function which is determined by a scale factor is described. At best position fitness function has minimal value.

The chosen fitness function is defined as follows:

$F = (1- EXP(-\beta)) (M_P + E_{SS}) + (EXP(-\beta)) (T_S - T_r)$

Here, F: Fitness function, M_P: Peak Overshoot, T_S: Settling Time, T_r: Rise Time, β: Scaling Factor (Depends on upon the alternative of the designer), scaling factor $\beta = 1$. In the Matlab library, we have to define a fitness function which has PID parameters as input values, as well as it returns fitness value of the PID based controller model as output and formatted as follows.

The Function [F] = fitness (K_D K_P K_I)

The step-wise description to the DFIG system for PSO based PID controller is as follows:

Step-1: Description of the PSO.

Step-2: The PSO algorithmic rule.

Step-3: The PSO algorithmic approach for the design of PID controller.

Step-4: The PSO algorithm with PID controller gain K_P, K_I, K_D.

Step-5: The fitness function concept for PID controller design with gain K_P, K_I, K_D.

The function [F] = fitness (K_P, K_I, $K_{D)}$

Step-6: The implementation of the fitness function in 6^{th} order transfer function of the DFIG system.

Step-7: For the best fitness function (f_{best}) of the DFIG system, the PSO-based PID controller gain (K_P, K_I, K_D) for 6^{th} order transfer function is described.

The PSO based controller gains for the 6^{th} order Transfer function reduced the model to the DFIG system [46], at best fitness function (fbest=0.1967) by several iterations which are given in Table (3.1).

Table 3.1: Gains of the PSO-based controller

Gbest Parameters	K_P	K_I	K_D
Gains	39.9781	7.6902	0.0271

The pictorial representation of the concept of fitness function to the controller design for DFIG system is shown in the following Figure (3.3).

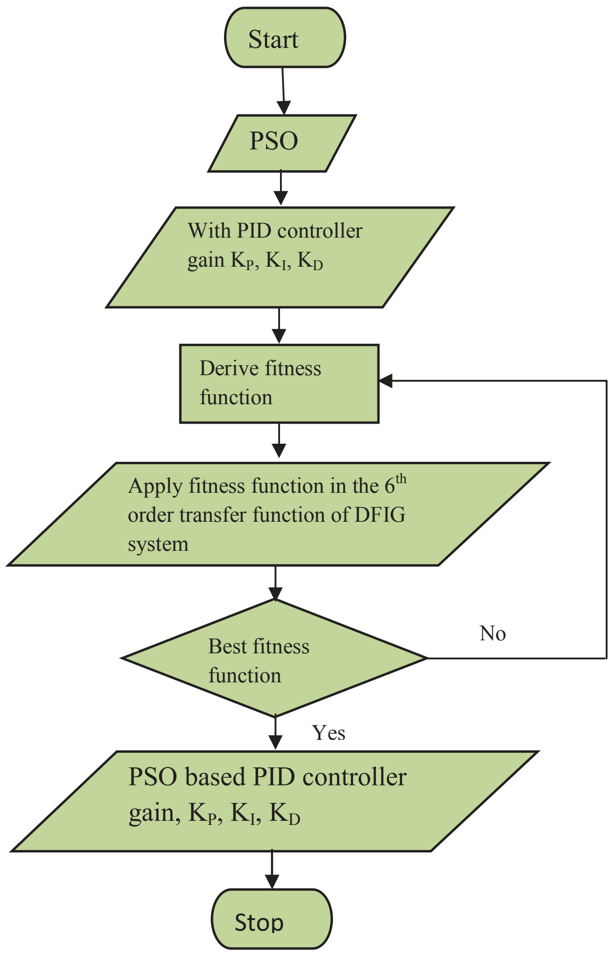

Figure 3.3: Flow chart for PSO based controller gain

3.5 Simulation along with results

3.5.1 Simulink response of DFIG system

Here Wind Farm - DFIG Average Model (MATLAB 13b) shows a 9 MW wind farm with the DFIG driven by a variable speed wind turbine. It consisted of six 1.5 MW wind turbines and linked toward a 25 kV distribution system exports power to a 120 kV grid throughout a 30 km, 25 kV feeder. In this illustration, the wind velocity is preserved invariable at 15 m/s. The control system uses a torque controller to retain the rate at 1.2 pu. The reactive power produced by the wind turbine is regulated at 0 Mvar. Intended for a wind velocity of 15 m/s, the turbine production power is equal to 1 pu of rated power, the pitch angle is 8.7 deg, and the generator speed is 1.2 pu. In the beginning, the DFIG wind farm produces 9 MW. The corresponding turbine speed is 1.2 pu of synchronous generator speed. The DC voltage is harmonized at 1150 V, and also reactive power is reserved at 0 Mvar. At the t=0.03 s the positive-sequence voltage rapidly drops to 0.5 pu. Causing an oscillation on the DC bus voltage and DFIG output power during a voltage sag, the control system attempts to standardize DC voltage along with reactive power at their set points (1150 V, 0 Mvar). The system recuperates in nearly four cycles. The response of the DFIG based wind turbine with PSO based controller gain and supervisory PID controller regarding voltage along with current at the terminals, active power generated; reactive power requirements and DC capacitor voltage, as well as generator speed, are shown in Figure 3.4 to 3.16 respectively.

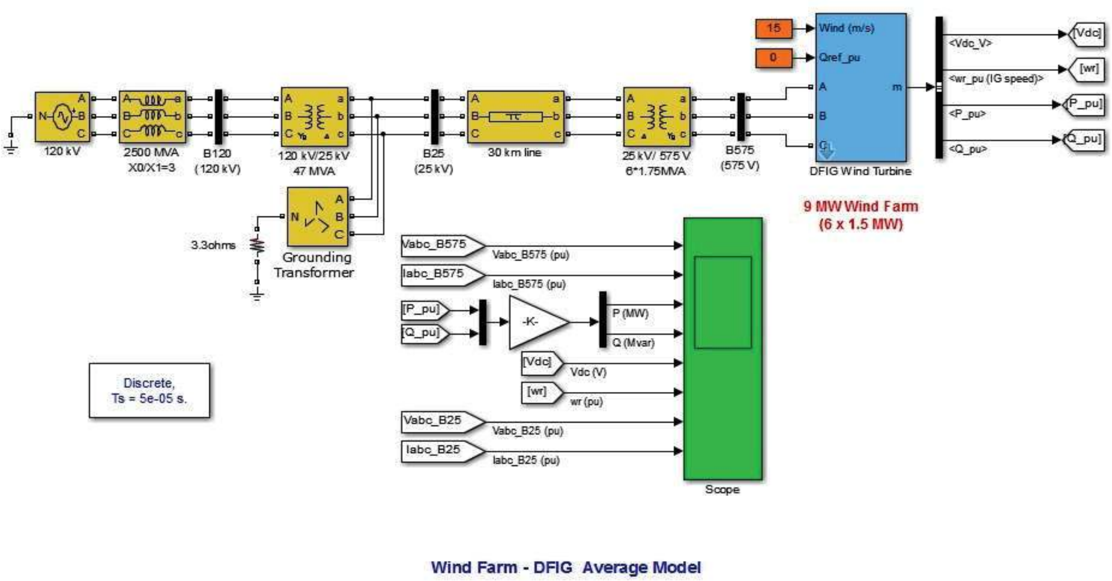

Figure 3.4: DFIG based wind turbine Average Matlab Simulink Model diagram

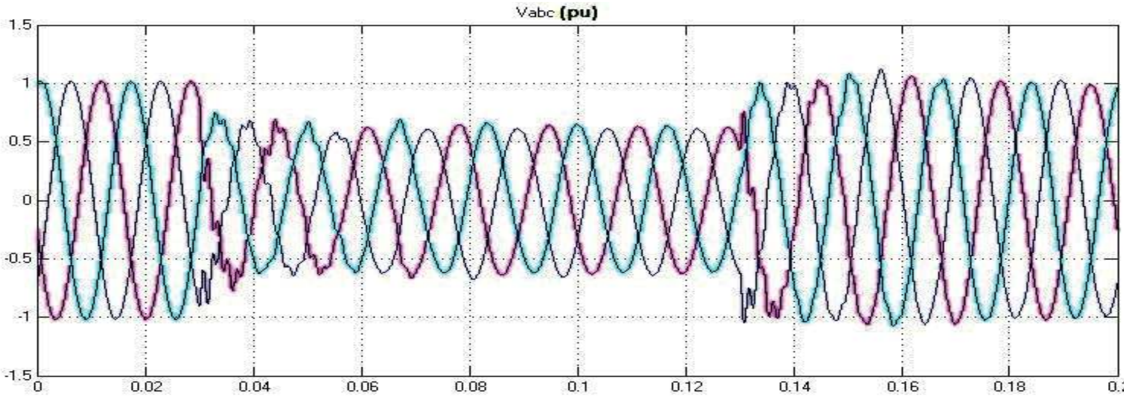

Figure 3.5: Voltages at the DFIG terminals in Pu (PSO based controller)

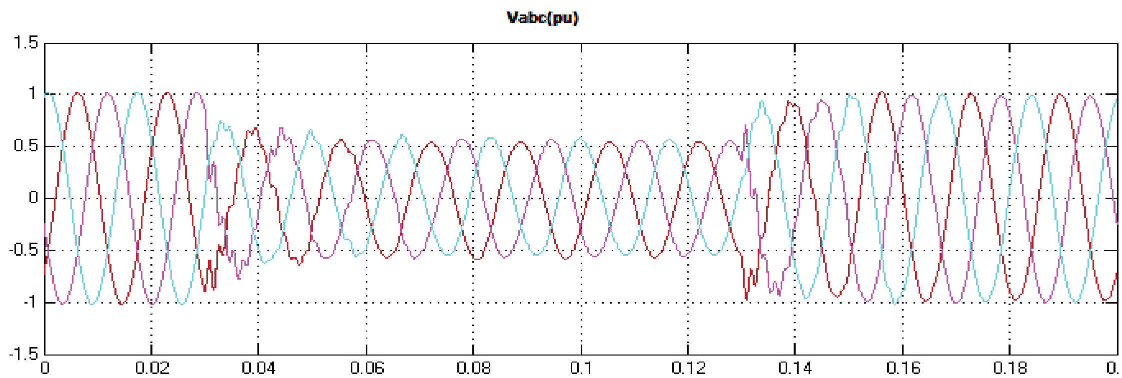

Figure 3.6: Voltages at the DFIG terminals in Pu (supervisory-based controller)

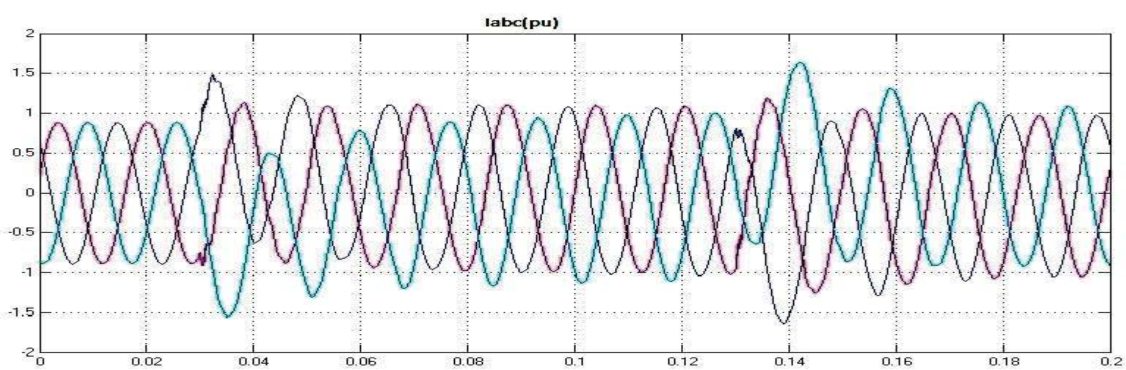

Figure 3.7: Currents at the DFIG terminals in Pu (PSO based controller)

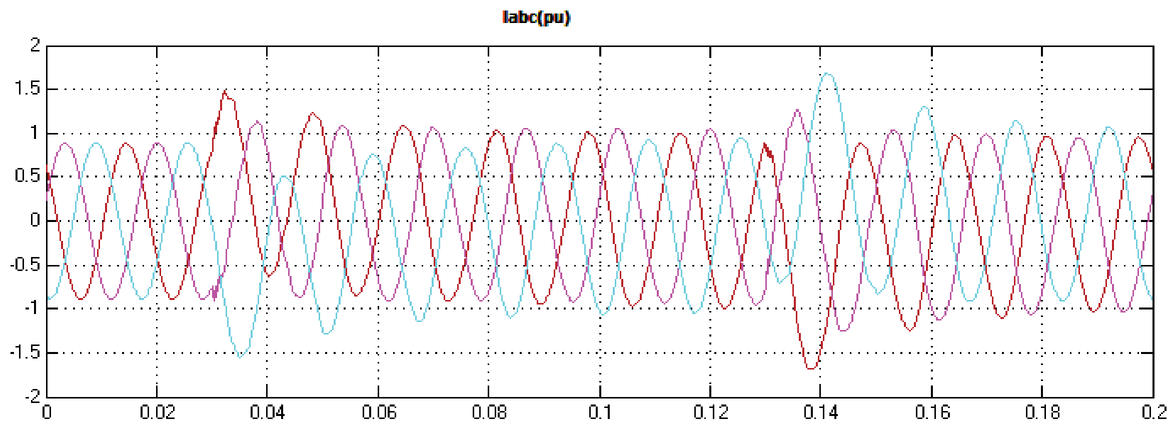

Figure 3.8: Currents at the DFIG terminals in Pu (supervisory-based controller)

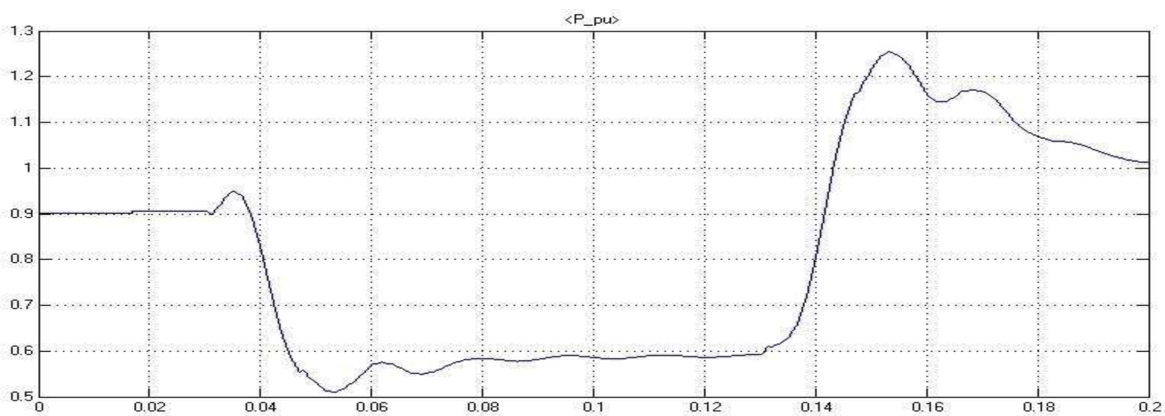

Figure 3.9: Active power delivered of the DFIG in Pu (PSO based controller)

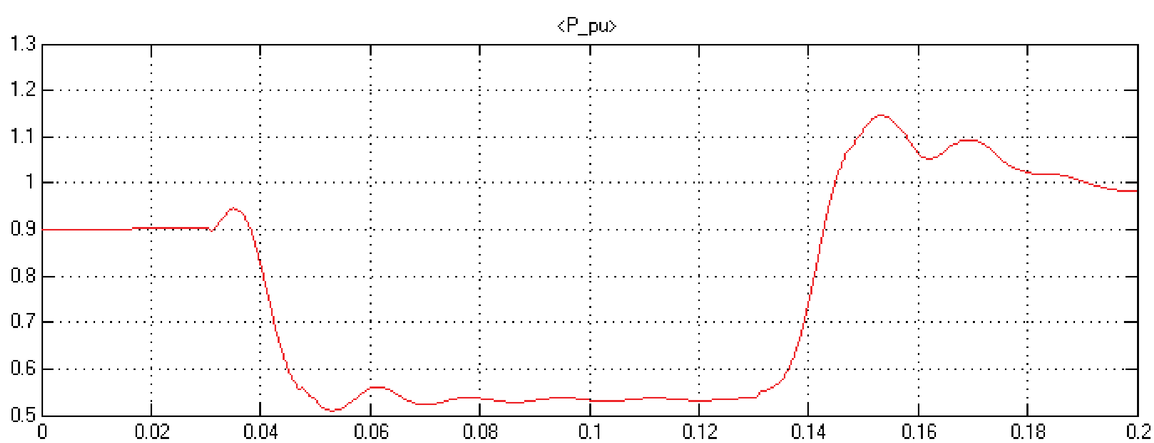

Figure 3.10: Active power given of the DFIG in Pu (supervisory-based controller)

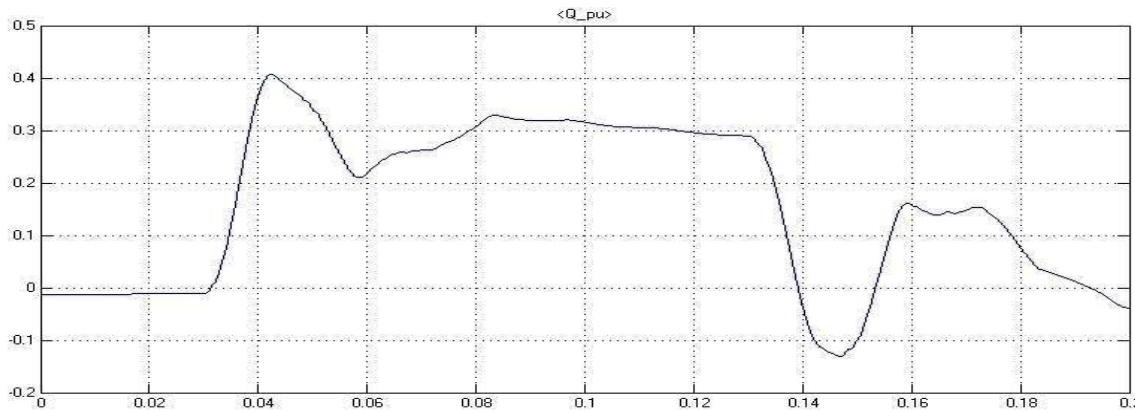

Figure 3.11: Reactive power requirement of the DFIG in Pu (PSO based controller)

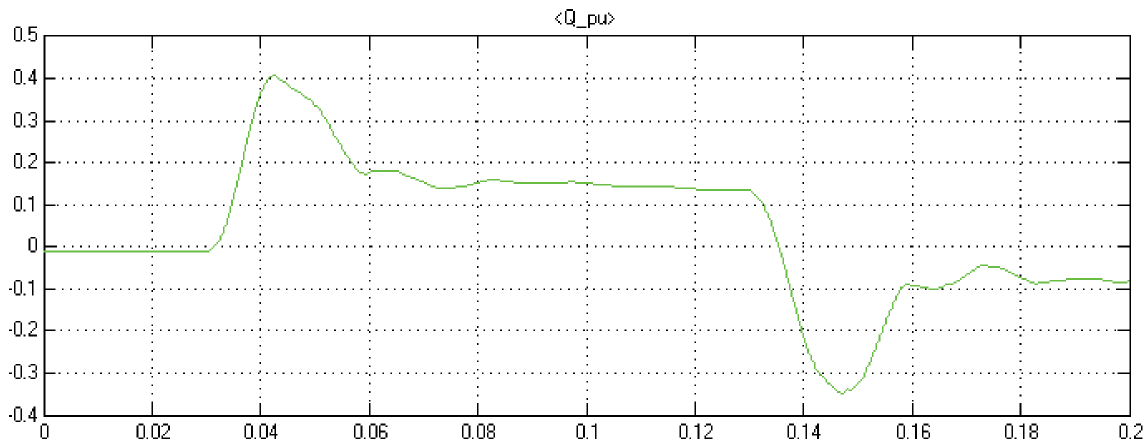

Figure 3.12: Reactive power requirement of the DFIG in Pu (supervisory-based controller)

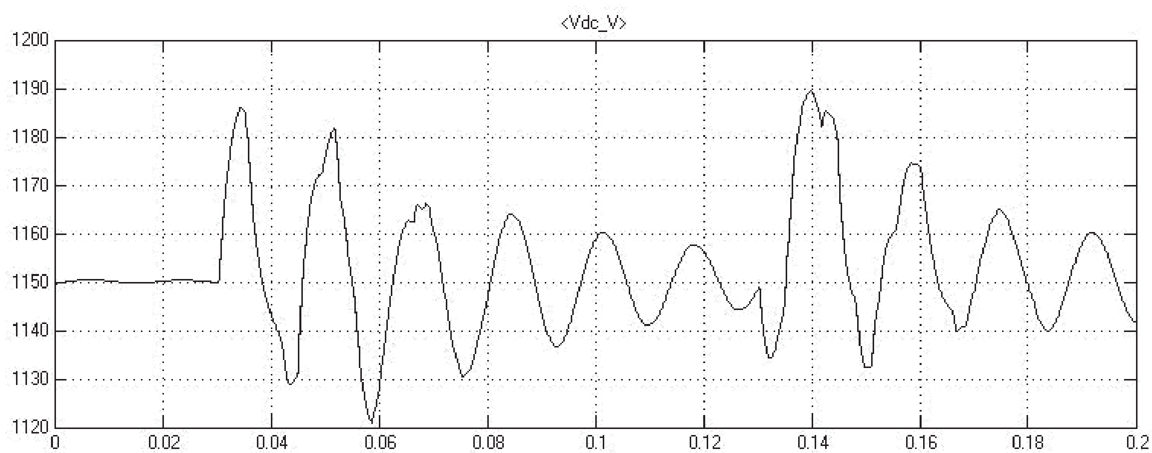

Figure 3.13: DC link voltage at the ordinary linkage capacitor (PSO based controller)

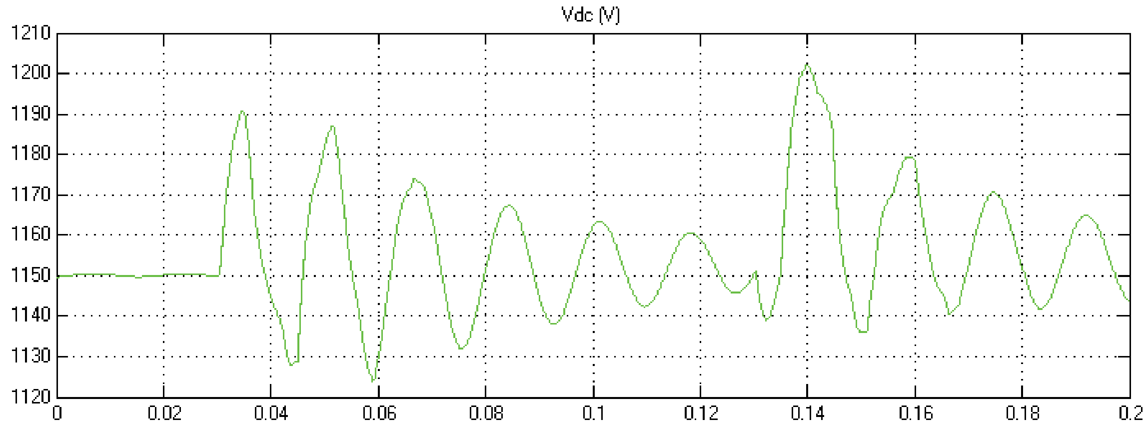

Figure 3.14: DC link voltage at the general coupling capacitor (supervisory-based controller)

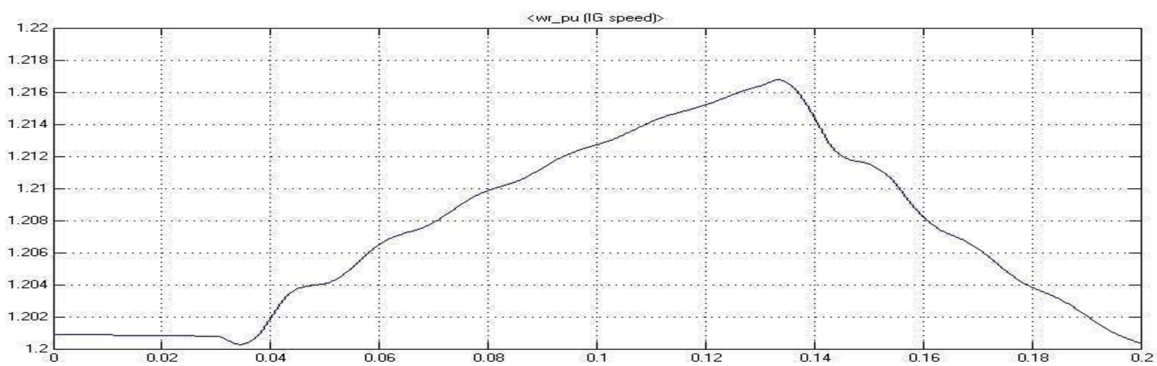

Figure 3.15: DFIG based wind turbine speed in Pu (PSO based controller)

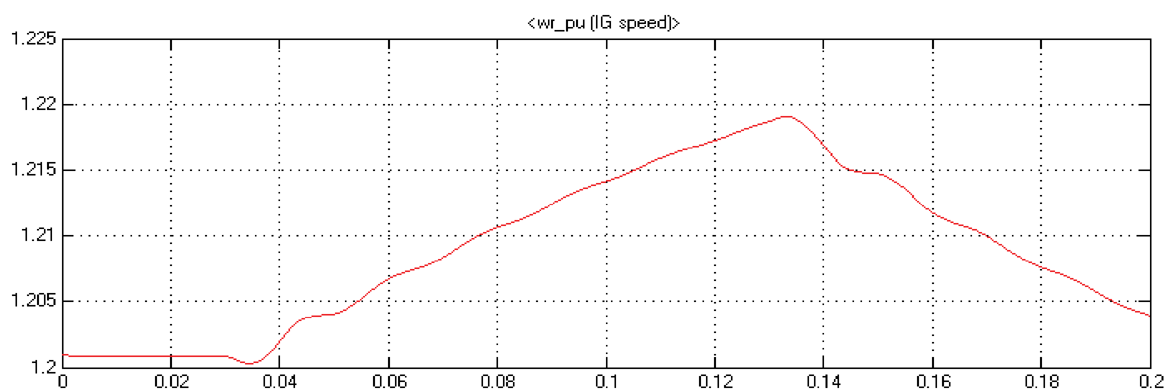

Figure 3.16: DFIG based wind turbine speed in Pu (supervisory-based controller)

Here Figure (3.4-3.16) concluded that the response of the DFIG system regarding terminal voltage, current, active-reactive power and DC-Link voltage along with generator speed had been improved with PSO based controller instead of a supervisory controller.

3.5.2 Supervisory Control Response by PID Controller

The response of the supervisory control based PID controller intended in [46]. However,from privious chapter 2 the step response of supervisory control based PID-controller is shown in Figure (2.18) along with the step response of Figure (2.18) has the following observations as shown in Table (2.3).

3.5.3 Response of PID Controller Using PSO Method

By using PID controller parameters which are designed in the previous Table (3.1). The step response of the system in revised for supervisory control based PID controller has been investigated, and step response of the system is exposed as in Figure (3.17). It is observable by a comparison result of supervisory PID control [46] along with one describe in this work which shows that; PSO based PID controller has extended to sufficiently damped supervisory control without compromising the speed of response to the control loop. Now the step response of open as well as close loop reduced order system [46] by PID controller with PSO technique is shown in Figure (3.17).

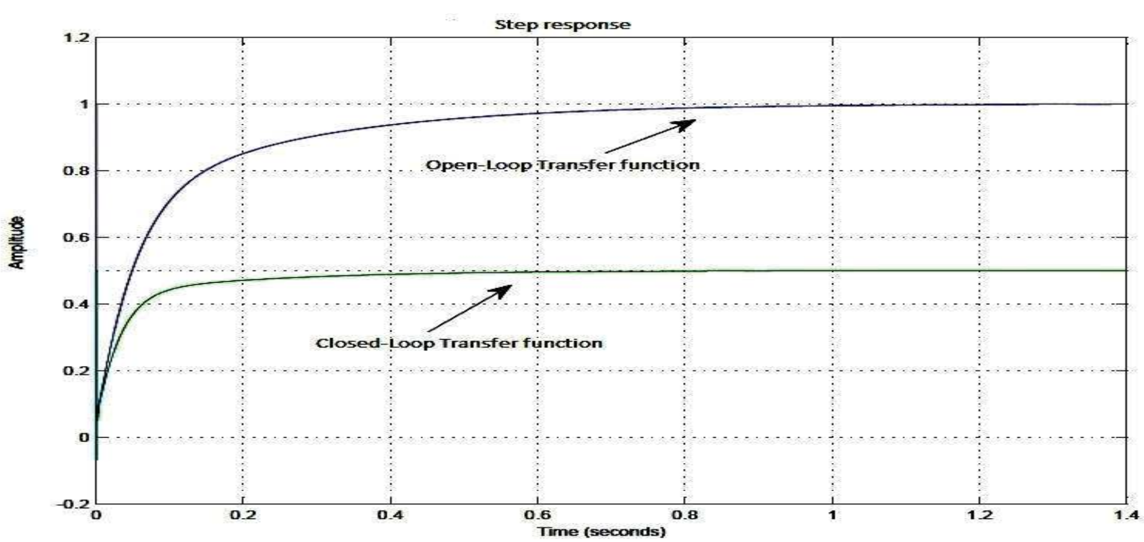

Figure 3.17: Step response of the PSO-based PID controller

The step response of Figure (3.17) has the following observations as shown in Table (3.2).

Table 3.2: step response observations of Figure (3.17)

Rise Time	Settling Time	Settling Min	Settling Max	Over Shoot	Under Shoot	Peak	Peak Time
0 sec	0.4061 sec	-0.0694 sec	0.4996 sec	0 %	13.8754%	0.500	0 sec

3.5.4 Comparison between supervisory and PSO based controller method

Comparison of Rise time, Settling time, Peak time, Overshoot and Peak are given in Table (3.3). The performance parameters comparison to the controllers are presented in Table (3.3). which is concluded as:

(i) The designed PSO based PID controller can reduce the steady state error to zero as the existing one.

(ii) The rise time, peak time and overshoot have been reduced to zero, and the response is no more underdamped.

(iii) The speed response is much better than that of the current controller. Hence intended controller in this thesis assists to eliminate reactive power variations in DFIG system.

Table 3.3: Comparison of Rise time, Settling time, Peak time and Overshoot as well as Peak between Supervisory and PSO based controller

Controllers	Rise Time	Settling time	Peak Time	Overshoot	Peak
supervisory PID	0.0152	0.5291	0.1419	15.1985	1.1512
PSO based PID	0	0.4061	0	0	0.5000

The publication from this part of the thesis work is as follows;

- **Om Prakash Bharti, R. K. Saket, S.K. Nagar, "**Controller Design for Doubly Fed Induction Generator Using Particle Swarm Optimization Technique," **Renewable Energy, Elsevier 114 (Part B), 2017, 1394-1406, indexed in SCIE (Thomson Reuters/Web of Science/Scopus).**

3.6 Conclusion

Though the supervisory control based PID controller improves system responses as compared with the open loop system, yet the numbers of vibrations are not detected completely. The PSO technique based designed controller not only enhances the system response but also reduces the percentage overshoot equal to zero. The obtained results show that the system using PSO based controller settles down in less time than the supervisory PID controller based system. The comparisons between PSO based controller and the supervisory PID controller precisely has been described. It is concluded that the settling time is reduced next to 12 percent approximately and the percentage overshoot, rise time, peak time, etc. are reduced to zero using proposed method. Finally, it is concluded that the PSO control technique provides another option to design a reliable and adequate controller for implementation in the DFIG based wind energy conversion systems.

Chapter 4 describes the controller design for doubly fed induction generator-based variable speed wind turbine by using Bioinspired techniques. It is based on exploiting the two efficient swarm intelligence based evolutionary soft computational method, i.e., Particle Swarm Optimization (PSO) and Bacterial Foraging Optimization (BFO) to design the controller for low damping plant of the DFIG. Finally, the obtained output is equated with a standard technique for performance enhancement of the DFIG based wind energy conversion system.