Design of PI Controller in Pitch Control of Wind Turbine: A Comparison of PSO and PS Algorithm

Sasmita Behera *‡, Bidyadhar Subudhi **, Bibhuti Bhusan Pati ***

*Dept. of Electrical & Electronics Engineering, Veer Surendra Sai University of Technology, Burla, India- 768018

**Dept. of Electrical Engineering, National Institute of Technology, Rourkela, India- 769008

*** Dept. of Electrical Engineering, Veer Surendra Sai University of Technology, Burla, India- 768018
(sasmitabehera2000m@gmail.com, bidyadharnitrkl@gmail.com, pati_bibhuti@rediffmail.com)

‡ Corresponding Author; First Author, Dept. of Electrical & Electronics Engineering, Veer Surendra Sai University of Technology, Burla, India-768018, sasmitabehera2000m@gmail.com

Received: 04.12.2015 Accepted:27.01.2016

Abstract- In wind energy conversion system (WECS) at wind speed above rated, the pitch angle is controlled to keep the generated output fixed so also the speed and the frequency. The model is built as a discrete model of WECS connected to Grid including a Line to Ground (LG) fault in Grid. A Proportional-Integral (PI) controller with gain K_p and K_i is used in pitch angle control loop. The proportional gain K_p and integral gain K_i are tuned through Particle Swarm Optimization (PSO) and Pattern Search (PS) algorithms. A comparison of two different objective functions with its weight adjustment is presented. The performances of the algorithms in designing the optimal controller are compared. The analysis indicates the superiority of PSO over PS and few others. It also takes less time to achieve the minimum error criteria. The controller designed using PSO minimizing the proposed objective function has better settling time as regards wind turbine speed response, compared to the others. The control action is validated in real time using OPAL-RT taking different cases of random wind speed, gust, gust with random wind speed and Line to Ground fault.
Keywords Particle Swarm Optimization; Tuning; Power system fault; PI control; Wind power generation.

1. Introduction

Variable speed wind energy conversion system (WECS), needs initiation of pitch control of wind turbines after a specific wind speed at which the generator power or generator speed of wind turbine reaches its rated value such that mechanical stress during wind gust is reduced, electrical power delivered to grid is controlled[1,18]. In this conversion scheme, Doubly Fed Induction Generator (DFIG) accompanied by a lower rated power converter is used to capture more power enabling variable speed operation [2,3]. When it delivers power to a grid (infinite bus), which imposes the frequency to be constant, the speed must be set to a constant value mechanically by pitch control of blades of the wind turbine by suitable pitch drives [4]. Proportional-Integral (PI) controllers have been widely used since last six decades in industries for process control applications. It has been emphasized in the report of Jonkman *et al.* [5] that there is no substantial benefit of using a Proportional-Integral-Derivative (PID) controller for pitch control; rather a PI controller gives satisfactory performance. Thus, in order to control the pitch angle Proportional (P), PI controllers have been used in variable speed WECS with DFIG [1]. Research on adaptive and fuzzy controllers are also going on parallel [7,8]. But anti-wind up and bump-less transfer are some issues which are to be properly dealt for switching between controllers with different gains [9]. Controller design by optimization of gains for a desired objective have exhibited superior performance in many applications rather than Ziegler-Nichols tuning method [1-3],[6],[12-15].Their implementations are also costly and complicated. A proper design of controller beforehand with slightly under-damped response can overcome excessive control effort. Thus to get optimized gains of the controller, proportional gain K_p and integral gain K_i for a desired performance of the control loop, Particle Swarm Optimization (PSO) technique which is a simple algorithms with many adjustable parameters, has been used and shown satisfactory performance for global search in many areas [1-3],[6]. A comparative study of unified controller parameter design using PSO, Mean Variance Optimization (MVO) and Neural Network (NN) based approach is reported in DFIG with FACTS devices by Qiao *et al.* [3]. A wind turbine can be approximated to be a second order system [4]. One simple optimizing algorithm; Pattern Search (PS) is confined mostly to search in a local area [10-13]. Other optimization techniques like Genetic Algorithm

(GA), Simulated Annealing (SA), Ant Colony Optimization (ACO), Differential Evolution (DE) and Firefly Algorithm (FA) have been applied successfully in many Engineering problems [14-18]. However, these techniques have not been compared in pitch control of WECS [19], in a consistent performance evaluation.

The performance of the control system depends on the controller structure and the techniques employed to optimize the controller parameters. An optimization problem essentially needs a performance evaluation criteria or fitness function. In regulator control problem such as pitch control, the performance evaluation criteria i.e. the conventional objective functions, based on error are Integral of absolute error (IAE), Integral of squared error (ISE), Integral time absolute error (ITAE) and Integral time squared error (ITSE) etc. defined and used in [1],[13-14],[17], [19]. A system is considered to be optimal when these error criteria index reaches an extreme value, commonly a minimum value. But IAE and ISE fail to settle the speed to the reference minimizing overshoot as well as the settling time. In pitch control obtaining a minimum overshoot as well as the settling time in wind turbine speed response has importance as it will not only reduce the transients as well as reduce the effort of the actuator thus save power and wear and tear of the actuator.

The objective of the presented work is as follows:

(i) To develop a discrete model of the grid connected WECS including simulated fault for analysis under disturbances with designed controller and suitable for test in real time.
(ii) To design a new objective function to obtain minimum overshoot & settling time in wind turbine speed in the operating range.
(iii) To study the effect of objective functions on the performance of wind turbine speed response.
(iv) To minimize objective functions for a step change of input wind speed below rated to above rated, instead of a fixed operating point supported in the work by Taher et al. [20].
(v) Apply the PSO technique after optimizing its parameters for similar problem. The optimal gains obtained for the new objective function reduces control efforts.
(vi) To demonstrate the advantages of PSO over PS and other optimization algorithms (FA, GA, DE, SA and ACO) reported for the same problem here.
(vii) To test the performance of optimal controller with other operating range of wind speeds and disturbances such as randomly varying realistic wind speed, wind gust and single line to ground fault on grid connection.
(viii) Test the effectiveness of control in real time experiment using OPAL-RT.

The paper is organized as follows, Section 2 gives a brief description of the pitch control system and Section 3 describes the algorithms of the optimization techniques. Section 4 compares the performance of the optimal pitch controllers designed followed by real time experimental results in Section 5 and Section 6 presents the concluding remarks.

2. Description of the pitch controller

The DFIG average model in MATLAB SIMULINK connected to a grid is used for the study [21-22]. In that model only a proportional controller is used in pitch control loop. Merit of using the model lies in that, it is a ready to use model with wind turbine and DFIG connected to grid. But the model does not include grid side fault condition, which is additionally included in the model for analysis under disturbances in this work.

For test in real time using OPAL RT which supports discrete/ continuous models in MATLAB/Simulink, a fixed step discrete model is simulated. The overall grid connected system is shown in Fig.1.

The pitch angle control structure inside one wind turbine used in this work is shown in Fig.2. The turbine rotor angular speed ω_r is sensed through sensors. The desired turbine speed, ω_ref is 1.2 pu in wind speed region above the rated, where maximum power extracted is 9 MW contributed by six units of wind turbine-DFIG of individual capacity of 1.5 MW each and the rated speed of wind is 12 m/s. For simplified calculation, all the electrical quantities and angular speed are normalized to per unit system (pu) with a base power of electrical generator and base speed of synchronous speed respectively. The tower dynamics and aerodynamic interaction between turbines are neglected in this design.

The operations of the wind turbine can be divided into three regions. The work of this paper is confined mainly to design of an optimal PI pitch controller for full load region marked as region III in Fig.3 where, wind speed is higher than rated but less than the cut out wind speed. The main control purpose in this region is to keep the generator power P_g around the rated generator power $P_{g,rated}$ at 1 pu. To achieve this goal, turbine rotor speed ω_r is to be kept around 1.2 pu (ω_ref), then P_g remains around $P_{g,rated}$. In full load region, to facilitate ω_r around 1.2 pu, the desired speed, pitch control is used. The PI controller sets the pitch control command which is input to the pitch drive. The pitch drive adjusts the pitch angle i.e. the angular position of the blade with reference to the plane perpendicular to the horizontal axis about its base on the rotor of the turbine-generator. The pitch control taken up here is collective pitch control (CPC) [1], same set for all the blades at a time through one drive.

The wind speed is increased in step from 8 m/s to 14 m/s and the controller gains K_p (Proportional Gain) and K_i (Integral Gain) running optimization program linking the model. Some of the equations related to pitch control are given below for understanding the system [1], [23-24].

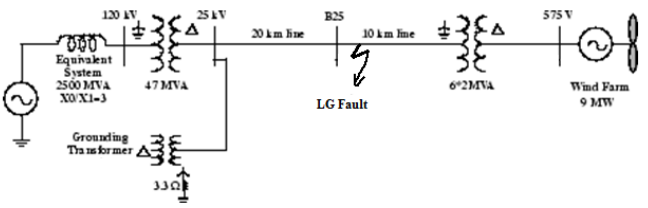

Fig. 1. Single line diagram of grid connected WECS

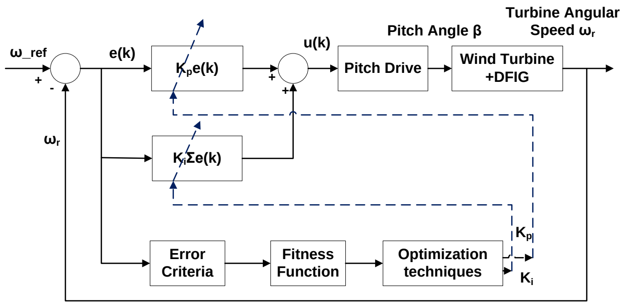

Fig. 2. Schematic diagram of Proportional-Integral (PI) Pitch Angle Controller in DFIG

The power available to the wind turbine shaft is given by:

$$P_w = C_p * P_{air} = \frac{1}{2} C_p \rho A V_a^3 = \frac{1}{2} \rho C_p \pi R^2 \tag{1}$$

Where, P_w = Power contained in wind (watt), C_p =Power coefficient which is, $f(\lambda, \beta)$, ρ= Air density (1.225 kg/m^3 at 15^0 C and normal pressure), A=Swept area/area of blades (m^2)=πR^2, V_a= Velocity of wind (m/s), β= Pitch angle (deg).

Where in, the tip speed ratio

$$\lambda = \omega_r R / V_a \tag{2}$$

Where, ω_r = Rotational speed of the wind turbine rotor (in pu), R=Radius of the swept area in m.

The shaft mechanical power developed by the turbine (P) at any wind speed can be expressed as,

$$P = \frac{1}{2} \rho C_p \pi \left(\frac{R^5}{\lambda^3}\right) \omega_r^3 \tag{3}$$

and

$$C_p(\lambda, \beta) = 0.5176\left(\frac{116}{\lambda_i} - 0.4\beta - 0.002\beta^{2.14} - 5\right) \exp^{-21/\lambda_i} + 0.0068\lambda \tag{4}$$

Where,

$$\frac{1}{\lambda_i} = \frac{1}{\lambda + 0.08\beta} - \frac{0.035}{\beta^3}$$

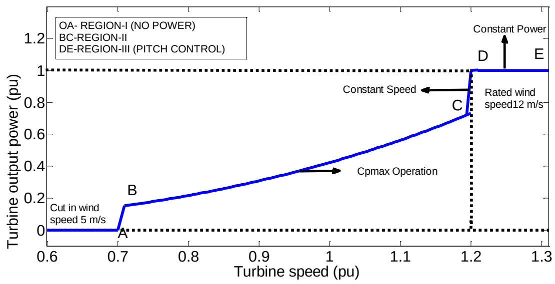

Fig. 3. Variation of wind turbine power with rotor speed, both in pu, showing regions of control

The drive train dynamics for one mass model considered here is expressed as

$$\frac{d\omega_r}{dt} = \frac{1}{2H}(T_e - T_m) - \frac{F}{2H} \tag{5}$$

Where, H=Wind turbine inertia constant (s) and F=Friction factor (pu).

So, (3) shows that, P is proportional to the cube of the rotor speed in region-II. For each wind speed increase λ changes and C_p depends on λ and β. As wind speed increases, λ decreases which affect C_p, hence power output P increases. In order to deliver constant power to grid, in region-III the pitch angle control signal is increased which decreases C_p and there by controls P. The pitch control signal is generated through a PI controller from the error.

The error is given as:

$$e(k) = \omega_{\_ref} - \omega_r(k) \tag{6}$$

and the control signal

$$u(k) = K_p e(k) + K_i \sum_{k=1}^{ksim} e(k) \tag{7}$$

The purpose of a feedback control system is to reduce the error, e(k), between any variable and its set value to zero as quickly as possible, where k is the no. of sample out of the total no. of samples taken for simulation k_{sim} for the discrete model. The best value of controller gains K_p & K_i are obtained by minimizing a function based on error. Therefore, any criterion used should take into account the variation of response over the whole range of time. Four basic criteria IAE, ISE, ITAE and ITSE are in common use [1], [14-15]. It is reported in literatures that, ITAE optimizes better than the others [13] as it minimizes the integral of absolute error with time i.e. the area under the error response is minimized for small as well as large errors, thereby minimizing the steady state error and overshoot as well. Hence, this ITAE criterion is evaluated in optimizing the gains of the controller subsequently as (J1) given by:

$$J1 = \sum_{k=1}^{ksim} k.\Delta t.|e(k)| \tag{8}$$

Where, Δt is the sampling time and represents the total simulation time.

The pitch angle command activates a servomechanism, which consumes power. Again the power consumed for movement of the servo system back & forth for pitch control system, is to be reduced which increases the life of the pitching mechanism. The faster the settling, the lesser the mechanical stress on the turbine and structure. Therefore, a new objective function has been introduced in order to reduce the settling time for faster settling and also giving weightage to error as follows:

Objective function

$$J2 = \sum_{k=1}^{ksim} k.\Delta t.|e(k)| + \mu.t_s \tag{9}$$

Where, μ is the weight given to settling time t_s. μ is chosen after verifying the effect on the performance, to achieve a compromising result between steady state error & settling time so as that damping is introduced and least settling time and overshoot in turbine rotor angular speed response ω_r is achieved.

3. Optimization Techniques

3.1.Particle Swarm Optimization (PSO)

PSO is one extremely simple population-based optimization method; which requires no gradient information. The general idea behind the optimization technique is birds (particles) flocking for food in a search space, where food is kept in best habitat, which corresponds to the optimized value of the problem. The particle position and velocity keep changing and best particle position p_best and best position among all the particles g_best is updated with each iteration. Finally the position vector reaches the best habitat and correspondingly, the best solution is achieved. It contains various control parameters such as confidence coefficients c1, c2 and inertia weights w1, w2 which changes the result obtained, the details of which are given by Poultangar et al. in [1]. However, the flow of the algorithm can be depicted from Table 1. These types of algorithm are dependent on the control parameters.

Table 1. Steps of Particle Swarm Optimization Algorithm

Step 1	Initialization
	Initialize algorithm parameters, No. of particles NP, Dimensions,Inertia_weight, Swarms_best_weight, Particles_best_weight, Swarm_size, Iterations and search bounds, initialize each particle randomly
Step 2	Evaluate the population
	Run the simulation model & evaluate the objective function
Step 3	Update p_best, g_best
Step 4	Update position & velocity of particles
Step 5	Repeat for particle K< NP
Step 6	Check termination criteria else continue to step2

3.2.Pattern Search (PS)

The Pattern Search (PS) optimization technique is a derivative free evolutionary algorithm. It is simple in concept, easy to implement and computationally less expensive. It has a flexible and (exploratory move step s_k, expansion factor λ_k, contraction factor θ_k) well-balanced operator to enhance and adapt the global search and fine tune local search [11]. The PS algorithm computes a sequence of solutions that may or may not approach to the optimal point. It begins with a set of points called mesh, around the initial points X0[13]. The proper initial point affects the optimal search. The mesh is created by adding the current point to a scalar multiple of a set of vectors in four coordinate directions in a two dimensional plane called a pattern. The scalar multiple is the exploratory move step size/mesh size. The points in the pattern are evaluated by the objective function in a particular fashion, starting from x axis in anticlockwise direction; and compared which is called polling. For minimization problem, if a point in the mesh is having lesser objective function value, it is taken as the current point at the next iteration and the poll is a success. After a success, the algorithm steps to next iteration. The current mesh size is increased by a factor 2 called the expansion factor. Now if in a particular iteration, no corner point in the mesh has a lesser objective function value than the value at initial/current point at that iteration, the poll fails and same current point is used in the next iteration. Besides, the current mesh size is also multiplied by 0.5, a contraction factor, so that the mesh reduces size at the next iteration and the process is repeated until stopping criteria is reached. The algorithm is dealt in detail showing the steps and variants by Torczon *et al.*[11]. The simple steps are presented in Table 2.

4. Results & Discussion

The implementation of all programs was performed in MATLAB® Version 2009b, in Dell PC with i-5 processor and 4GB RAM. The model parameters adopted by the authors are given in Appendix A[1]. While simulating, the wind speed change started at 5 s from 8 m/s to 14 m/s in a single step. This range of wind speed is taken instead of

Table 2. Steps of pattern search algorıthm

Step 1	Initialization
	Initialize algorithm parameters mesh size, Iterations, contraction factor, expansion factor, maximum function evaluations, search bounds and initial point X0
Step 2	Fitness Evaluation
	Run the simulation model & evaluate the objective function
Step 3	Poll the points in the pattern checking f(X1)>f(X0)
Step 4	If yes expand
Step 5	Else contract
Step 6	Repeat iteration for N< Nmax
Step 7	Check termination criteria else continue to step2

larger value because average wind speed varies in this range and higher value occurs occasionally [24,26]. Furthermore, for higher wind speed if the controller is tuned, there is chance of reduced control effort for lower wind speed as sensitivity of pitch angle to aerodynamic torque is small in this range.

The parameters of PSO are chosen after tuning parameters of algorithm for this problem as in Table 3. The model is a discrete model with sample time $5X10^{-5}$ to study accurately the dynamics resulting from control system. Hence, the simulation is performed for 100 s. The best values are highlighted in bold. The various parameters for the PS, SA, DE, FA, GA and ACO algorithms that are used to obtain the proposed optimized controller gains are given in Appendix B.

The controller gains K_p and K_i have been found out for optimal response using two different objective functions based on error for a step change of wind speed from 8 to 14 m/s in the same search space for population size N=10, number of iterations I=20. The optimization was run 20 times as randomness is present in each of the algorithms and the best values are given. The time domain performance is also presented in comparison to five other algorithms such as FA, GA, DE, SA and ACO with same population/ genome size and consistent number of iterations for the proposed objective function J2. It is observed from Table 4 that increasing order of best fitness value is: PS, PSO, SA, DE, FA, GA and ACO. But, when settling time is seen from

Table 3. Tuning of PSO parameters

Varied parameter	Min	Max	Avg	Std Dev	Other parameters
C1=1.0	8.7343	10.099	9.2983	0.7179	P=10, N=20, C2=0.5
C1=1.5	10.9362	11.218	11.2853	0.3519	
C1=2.0	**6.4144**	7.2286	6.7547	0.4243	
C2=0.5	**6.4144**	7.2286	6.7547	0.4243	P=10, N=20, C1=2.0
C2=1.5	6.4144	6.61392	6.4986	0.1033	
C2=2.0	11.0041	11.6414	11.2933	0.3299	
P=5	8.7343	9.054	8.9323	0.1715	N=20, C1=2.0, C2=0.5
P=10	**6.4144**	7.2286	6.7547	0.4243	
N=5	8.7343	11.220	10.0172	1.2430	P=10, C1=2.0, C2=0.5
N=10	6.7816	7.7315	7.1243	0.6152	
N=20	**6.4144**	7.2286	6.7547	0.4243	

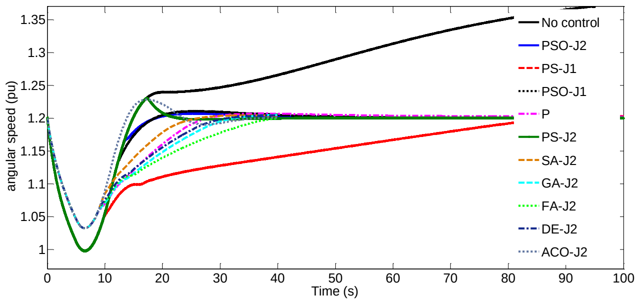

Fig. 4. Variation of rotor angular speed with time(s) for step change of wind speed

Table 4, the order is: PSO, PS, SA, DE, GA, ACO. The steady state error is the least for PSO. In Fig. 4 all the responses with gains determined using various algorithms with different objective functions and control are given for a comparison for the cases in Table 4. The turbine speed response without any controller ω_r increases gradually without settling to the set point of 1.2 pu in 100 s. This is due to that, the order of the system as a whole is higher than a second order system.

It can be seen that there is considerable overshoot and settling time is also high with only ITAE (J1). As the ITAE criteria gives the lowest steady state error (SSE) as compared to other standard criteria [13], a new objective function has been proposed here taking ITAE and giving weightage to settling time t_s. It can be seen from Fig. 4, that for the new two part objective function given in (9), the settling time is drastically reduced and the overshoot is also minimum. The error is in the range of 10^{-4} though higher than other four cases in Table 4, is well within requirement. The weightage given to t_s is varied to achieve best result as tabulated in Table 5. With ITAE (J1) only the steady state error is less but with larger overshoot. With objective to minimize t_s, the settling time is reduced, but overshoot has increased. The best result is obtained with μ= 0.5.

The simulation model was also run for 40 s and 100 s under two cases of optimization and the results after 20 generation are summarized in Table 6. From the tabulated results it is clear that for both cases of simulation run time, the algorithm run time for PS is nearly 1.7 times as compared to PSO. With increase of simulation time, the fitness value increases as the objective function is a function of time in both cases of optimization. But, with increase of simulation run time, standard deviation of fitness is decreased, with corresponding decrease in time domain performances i.e. the average settling time and average % overshoot. This indicates an improvement of search. For simulation time of 40 s PSO takes the least time to complete optimization. The increasing order is: PSO, SA, FA, PS, GA, DE and ACO. Thus, PSO is computationally less expensive. Though, DE and FA have given better performance than PSO in other applications, with reduced population size and iterations their

Table 4. Tıme domaın performance ındıces evaluatıon for optımızatıon technıques

Controller	Optimization algorithm	k_{pp_pitch}	k_{ii_pitch}	Best Fitness Value	Objective function	Settling time t_s(s) (2% band)	%Over-shoot	ITAE	SSE
PI	PSO	170.9139	12.7403	20.2403	J1	67.3014	0.9103	20.2403	$1.5507X10^{-5}$
	PS	2496.0	0.001	21.7794	J1	74.9685	-0.5838	21.7794	0.0029
PI	PSO	**480.4684**	**49.8**	**24.2972**	**J2**	**14.6862**	**0.6346**	**16.9541**	**$-1.1228X10^{-4}$**
	PS	172	50	23.6971	J2	17.4593	2.2455	14.9675	$-9.4731X10^{-5}$
P		500	-	-	-	25.4682	-0.1085	31.3964	0.0026
PI	FA	2375.3731	30.772	26.4241	J2	29.4799	-0.0015	11.6841	0.0047
PI	GA	917.1396	4.1099	27.0501	J2	25.2548	0.4194	14.4227	0.0068
PI	SA	1043.4691	49.8728	24.8882	J2	19.6759	0.3515	15.0502	0.0043
PI	DE	1250	25.001	26.2108	J2	23.8019	0.3682	14.3098	0.0058
PI	ACO	7	3	30.1789	J2	19.8095	2.3533	20.2742	0.0087

Table 5. Effect of change of weıght ın objectıve functıon

Weight (100s run)	k_{pp_pitch}	k_{ii_pitch}	Best Fitness Value	Settling time t_s (s) (2% band)	% Overshoot	ITAE	SSE
ITAE	170.9139	12.7403	20.2403	15.2360	0.9103	20.2403	1.55e-05
0.1	140.2020	15.4247	19.4936	14.9861	0.7653	17.9950	-2.66e-05
0.3	879.7971	41.5416	27.1622	17.5346	0.7778	21.9018	-1.7260e-4
0.5	**480.4684**	**49.8**	**24.2972**	**14.6862**	**0.6346**	**16.9541**	**-1.1228e-4**
0.7	278.8957	39.0128	25.9797	18.1200	-0.3011	13.2957	-1.8425e-4
0.9	140.1367	37.7449	27.6482	17.7421	1.5353	11.6803	-8.2875e-5
1.0	82.4695	28.0604	29.1586	19.0429	2.8431	10.1157	-1.0523e-4
Only t_s	72.6567	24.4956	19.1271	19.1271	2.8451	15.1313	-1.7667e-4

perfomance suffer here. This is also in accordance with no free lunch theorem. The PSO has better searched globally with higher standard deviation where as the standard deviation for 20 runs has reduced gradually in the order : PSO, PS, SA, FA, GA, DE and ACO.

The best fitness in each iteration is plotted in Fig. 5 for the best results in minimizing the objective function J2 by PSO and GA. It can be seen that the variation from first iteration (starting from 670) to 20th iteration (23.6971) is wide in case of PS. whereas, the variation is with in 30 from initial to final in PSO. It also indicates that, if less no. of iterations is chosen, PS will be far behind PSO in achieving minimum. Performance of PS may be at par with PSO only when it is initialized in a smaller search bound near the global optimum. But, it will take more runs with few converging to the global minimum than PSO. On the other hand, PSO with random initialization converges to global optimum in most of the cases.

For K_p=**480.4684** and K_i=**49.8** optimized by PSO the turbine speed and the control action are shown in Fig. 6 for a change of wind speed from 8 m/s to 14 m/s starting at 5 s and finally maintaining 14 m/s. It is seen that, the pitch angle has increased up to 3 deg at 27 s and gradually reduced to 2 deg after 35 s. It is not reduced to zero because wind speed is finally settled to 14 m/s(2 m/s higher than rated).

Table 6. Performance of algorithms with simulation run time

Simulation time (s)	Optimization algorithm	Average fitness	Standard deviation	Average settling time (s)	Avg % overshoot	Algorithm run time (s)
40	PSO	24.9384	5.7819	15.7145	1.1100	1943
40	PS	23.8011	2.3922	26.7911	1.6449	3241
100	PSO	25.6223	4.7601	14.9604	0.8731	4747
100	PS	24.3924	1.9570	15.0504	1.2806	8102
40	FA	26.9374	1.4018	30.1408	1.2178	2214
40	GA	27.4598	0.8614	26.7142	1.8215	18383
40	SA	25.2117	1.8716	20.1046	1.7012	2058
40	DE	26.5819	0.7713	24.8317	1.7514	19212
40	ACO	30.4318	0.7013	20.0105	2.3832	37801

As wind is stochastic, hence it necessitates testing the controller under such a condition. So a random wind speed signal shown in Fig.7 varying between 9 to 15 m/s is applied to test the system with different controllers. From Table 7, the standard deviation in angular speed of rotor is seen to be low with the proposed PI controller (0.0338) as compared to the standard P control (0.0384) and without control (0.0482). Similarly, the pitch angle has increased less (5.0911) as compared to P control (7.0107). Thus better control is achieved with the proposed controller as compared to P control designed by Zeigler-Nichols tuning rule. In a similar work on active power control using fuzzy-PI control[8] the standard deviation is shown to be 0.0751 in active power(the controlled variable) where as in our work it is reduced to 0.0338 in angular speed of rotor. Thus the speed fluctuation is reduced as a result of better control action for smoothing it. The effect of different control can be seen from Fig.7 in responses of wind turbine angular speed ω_r and pitch angle β. It shows the effectiveness of the proposed method when random wind speed profile is applied.

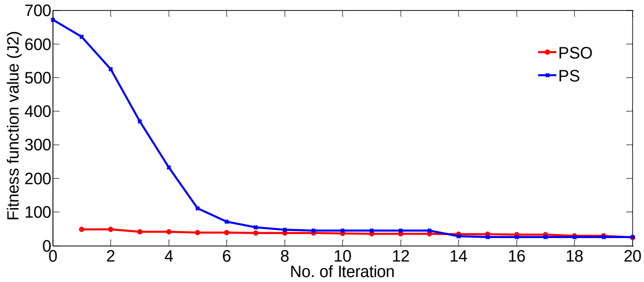

Fig. 5. Convergence plot of PSO and PS algorithms

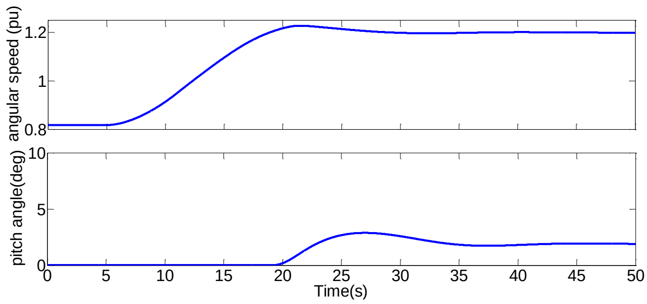

Fig. 6. Variation of angular speed and pitch angle with time for step change in wind speed

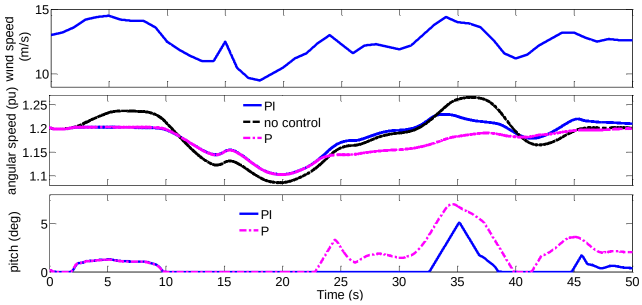

Fig. 7. Comparison of controllers under random wind speed

5. Real Time validation of results in Opal-RT

The Opal-RT is a fully digital power system simulator [26] developers to prove their ideas, prototypes and final products in a realistic environment with a typical sampling time step of 50 µs with a combination of custom software and hardware. It is an ideal tool for the design, development and testing of power system protection and control schemes.

The model was compiled in master and slave processors. Two outputs from the system, turbine speed or angular speed of rotor ω_r and corresponding control signal pitch angle are observed for the same cases as studied in SIMULINK. The results are observed in a digital storage oscilloscope running it from host PC, which is connected to RT-Lab simulator as in Fig.8.

Table 7. Performance of controller for random wind speed (9 to 15 m/s)

CONTROLLER	Angular speed (pu)	Pitch Angle (deg)	
	Std Dev	Max	Std Dev
NO CONTROL	0.0482	0	
P	0.0384	7.0107	1.7994
PI (PSO)(J2)	0.0338	5.0911	0.9476

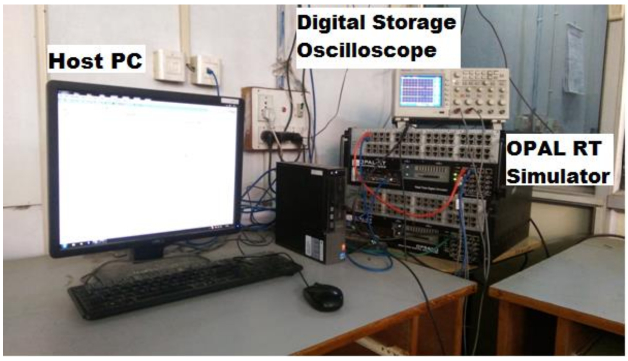

Fig. 8. Experimental set up of OPAL-RT lab

rated wind speed of 12m/s was checked in both MATLAB SIMULINK and OPAL RT. The pitch angle signal was also 0 deg, which is expected at rated as in Fig.9 (a). When run in OPAL RT, the pitch angle shows a zero(blue) and rotor angular speed (yellow) one big and one small division 1.2 pu in Fig.9 (b) . This is the reference angular speed of rotor that is to be tracked in pitch control when wind speed is above 12 m/s. The small yellow arrow in right points to the reference level.

3.1. Random wind speed

A wind speed signal taken from real measurement of wind speed from NREL with standard deviation 1.2560 is applied to the model [26]. It can be seen from Fig.10 (a) that for random wind speed change within 9 to 15 m/s the pitch angle is increased to 5 deg for randomness in wind speed increasing to 14.5 m/s, the pitch angle varies within 1 deg when model is run in MATLAB SIMULINK. It indicates that, the ω_r settles around 1.2 pu after 10 s though the wind speed falls below the rated wind speed of 12 m/s after 12 s. Similar results are shown in Fig.10 (b) when model is run in real time in OPAL RT. As the variation of angular speed of turbine rotor ω_r is in the range of 0.8 to 1.4 pu and the pitch angle from 0 to 27 deg, and both are taken at a time in one scope, the variation of pitch angle with in 1deg (one small div) is seen as jitter, similar is case of ω_r.

3.2. Wind Gust (14 to 20 m/s)

This wind gust at 5 s taken here is 14 to 20 m/s beyond the maximum value of wind speed (14 m/s) for which the controller is tuned shown in Fig.11. With the designed PI controller, the response tracks as in Fig. 12(a). The overshoot with such steep rise of wind speed also is 1.244 pu which is within 5% band of the steady state value. The pitch angle rises up to 6 deg at 8 s then reduces to 1.4 deg. But, as with reduced control signal rotor angular speed again raises to 1.206 pu, pitch angle rises to 2.6 deg finally settles to 2 deg at 33 s when ω_r settles to reference. The real time results in Fig.12 (b) shows a similar control action (2 peaks). The results in scope are shifted by 1big division in x-axis. Hence 5 s is 1&1/2 big division.

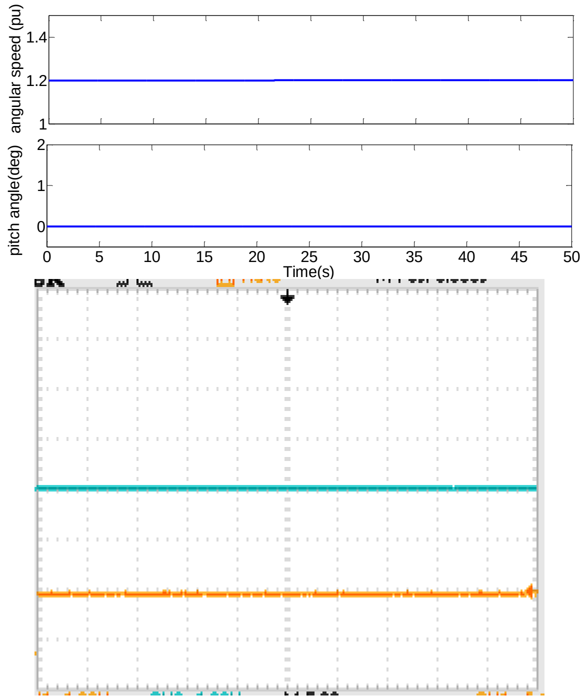

Fig. 9 (a) Angular speed of rotor and pitch angle for rated wind speed 12 m/s

(b)Real time OPAL-RT results (angular speed of rotor ω_r (yellow) y scale-1 pu/div, pitch angle (blue) y scale-10 deg, x-scale 10 s/div)

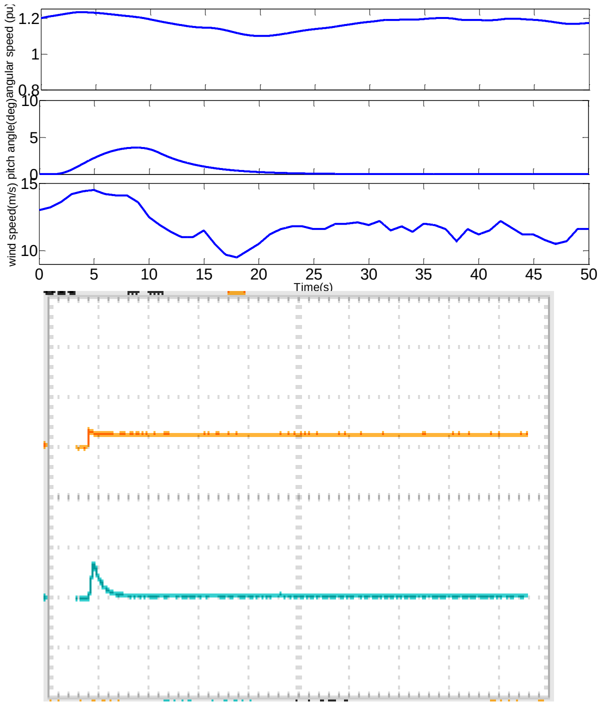

Fig. 10 (a) Variation of angular speed of rotor and pitch angle with time for random wind speed

(b)Real time OPAL-RT results (angular speed of rotor ω_r (yellow) y scale-1 pu/div, pitch angle (blue) y scale-10 deg, x-scale 10 s/div)

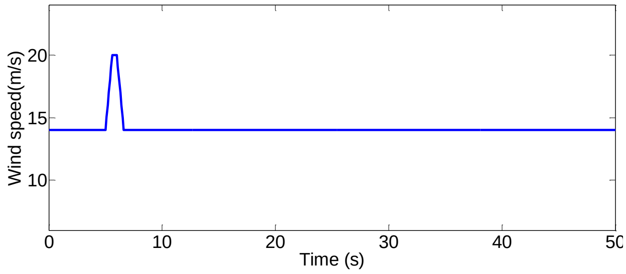

Fig. 11. Wind gust from 14 to 20 m/s at 5 s.

3.3. Wind Gust (8 to 14 m/s) with randomness & Line to Ground (LG) fault

A case of random change of wind speed with gust occurring at 5 s with a line to ground fault on B phase of grid occurring at 10 s is simulated. The wind speed is shown in Fig.13. The mean wind speed after gust is 17 m/s. This situation is common under stormy weather condition. The controller designed is able to perform as seen from the Fig.14 (a) and settles the response of the wind turbine speed at 25 s after gust and short circuit. The pitch angle after rising to 9.8 deg to control speed due to gust, for subsequent random variation, falls to 8.2 deg. On comparison of this result with that of case 2 the final settled pitch angle is around 8.2 deg in this case, higher for higher wind speed, where as in case 2 it

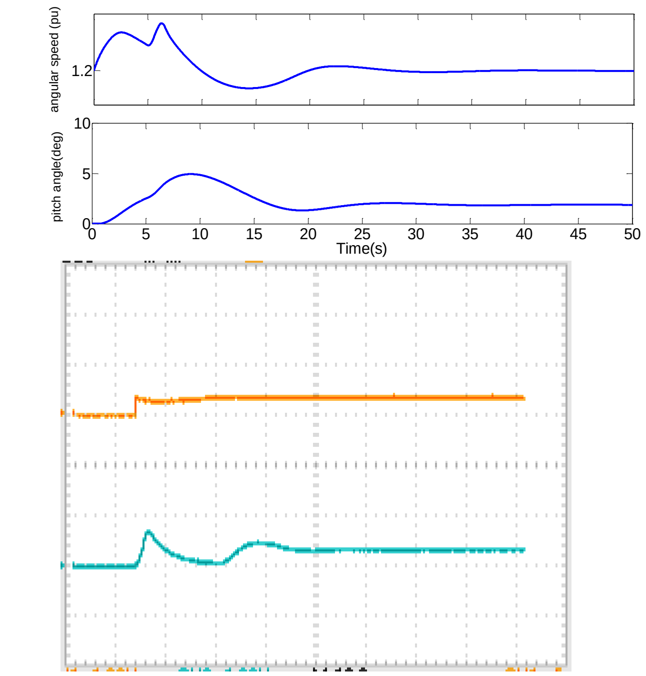

Fig. 12. (a) Variation of angular speed of rotor and pitch angle with time for wind gust

(b)Real time OPAL-RT results (angular speed of rotor ω_r (yellow) y scale-1 pu/div, pitch angle (blue) y scale-10 deg , x-scale 10 s/div)

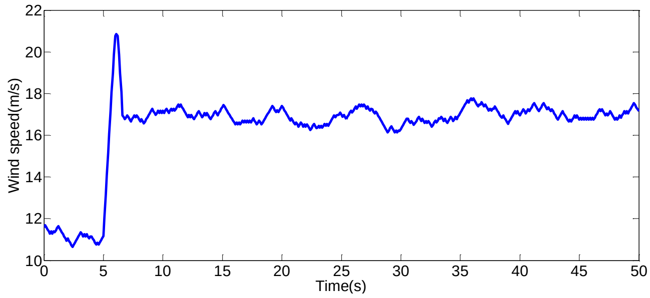

Fig. 13. Wind gust at 5 s with random wind speed

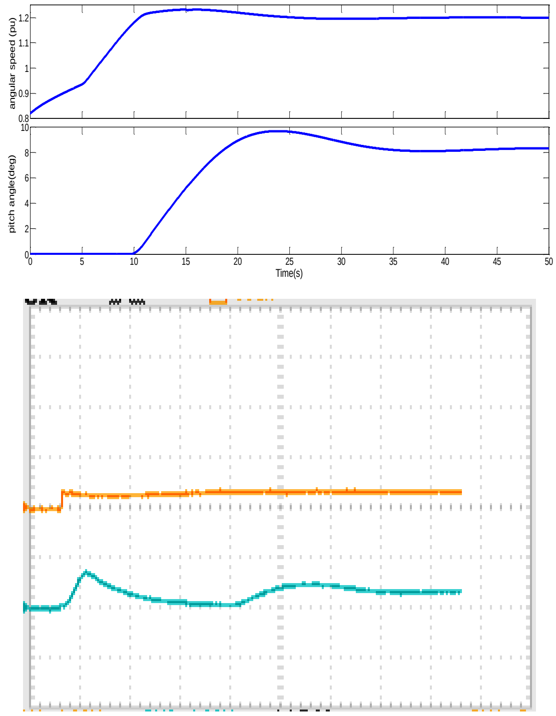

Fig. 14 (a) Variation of angular speed of rotor and pitch angle with time for wind gust, randomness & Line to Ground (LG) fault

(b)Real time OPAL-RT results (angular speed of rotor ω_r (yellow) y scale-1 pu/div, pitch angle (blue) y scale-10 deg, x-scale 10 s/div)

is around 2 deg for lower wind speed. The real time result under this condition in Fig.14 (b) shows similar rise of pitch angle starting earlier at 7 s than 10 s shown in MATLAB/Simulink result. However, as random signal is used to generate wind model, the response differs in different run. As randomness is present, as ω_r further increases pitch control action again increases to 4 deg after a decrease to 2 deg. The flickers noticed in ω_r and pitch angle are due to random wind speed. For above rated random variation in wind speed the pitch control varies for change in wind speed which is noticed.

6. Conclusions

A well designed control system for wind energy conversion system (WECS) enables better power quality and efficient generation. In this paper optimal fixed gain PI controller was designed using seven optimization techniques, PSO, PS, FA, GA, SA, DE and ACO taking two objective functions. Ample insight into the algorithm parameter setting and rigorous analysis on proper objective function design are also given. In addition, their performances were compared. PSO takes less time for same number of iterations and independent of initial point, hence achieves global optimum. Whereas, PS and SA attain similar results with proper setting of initial condition, with increased iteration and increased time. Further, here the controller performance has been analyzed for realistic wind speed and different simulated disturbances. Though the steady state error is little higher for the new objective function given, the control effort is less as the response has minimum overshoot in the optimization range. The settling time also varies as the time of occurrence of disturbance changes, but it is minimized. Hence it can be concluded that after design and fixing gains of PI controller by this novel objective function, a simple controller is expected to control the system under varying disturbances.

Appendix A

Parameters of wind turbine:

Power capacity of individual turbine=1.5 MW, Cut-in wind speed=5 m/s, Rated wind speed=12 m/s, Cut out wind speed=25 m/s, Rated turbine speed =1.2pu, Maximum pitch angle=27deg, Rate of change of pitch angle=2deg/s.

Appendix B

Parameters of Pattern Search algorithm:

Initial size=1, Expansion factor=2.0, Contraction factor=0.5, Maximum iteration=20.

Parameters of Genetic algorithm[14]:

Selection: Stochastic Uniform, Crossover probability: 0.8, Type: Scattered, Mutation probability: 0.1, Elite count: 2, Scaling function: Rank.

Parameters of Simulated Annealing algorithm[15]:

Method: Fast annealing, Temperature update function: Exponential temperature update with 0.95 ^iteration.

Parameters of Ant Colony Optimization algorithm[16]:

Initial pheromone τ_0=10, pheromone weight α=0.3, pheromone evaporation rate ρ=0.1.

Parameters of Differential Evolution algorithm[17]:

Strategy: DE/best/1/exp step size scaling factor F=0.8, crossover probability CR =0.7.

Parameters of Firefly algorithm[18]:

Light absorption coefficient γ=1, Randomization parameter or step size factor α=0.2, Attractiveness β= 1.0, randomness reduction scaling factor δ=0.97.

References

[1] I. Poultangar, R. Shahnazi and M. Sheikhan, “RBF neural network based PI pitch controller for a class of 5-MW wind turbines using particle swarm optimization algorithm,” ISA Transactions Vol.51, pp. 641–648, 2012. [Online]. Available: http://dx.doi.org/ doi:10.1016/j.isatra.2012.06.001

[2] Y. Tang, P. Ju, H. He, C. Qin and F. Wu, "Optimized Control of DFIG Based Wind Generation Using Sensitivity Analysis and Particle Swarm Optimization," IEEE Transactions on Smart Grid, Vol. 4, Issue 1, pp.509-520, 2013. [Online]. Available: http://dx.doi.org/ 10.1109/TSG.2013.2237795

[3] W. Qiao, J. Liang, G. K. Venayagamoorthy and R. Harley, “Computational Intelligence for Control of Wind Turbine Generators,” Proc. of IEEE Power and Energy Society General Meeting, pp.1-6, 2011. [Online]. Available: http://dx.doi.org/ DOI: 10.1109/PES.2011.6039778

[4] C. Jauch, S. M. Islam, P. Sorensen and B. B. Jensen, “Design of a wind turbine pitch angle controller for power system stabilization,” Renewable Energy, Vol. 32, pp. 2334–2349, 2007. [Online]. Available: http://dx.doi.org/ DOI: 10.1016/j.renene.2006.12.009

[5] J. Jonkman, S. Butterfield, W. Musial and G. Scott, “Definition of a 5-MW Reference Wind Turbine for Offshore System Development,” Technical Report NREL/TP-500-38060, Feb. 2009.

[6] X. Sui, Y. Tang, H. He and J. Wen, “Energy-Storage-Based Low-Frequency Oscillation Damping Control Using Particle Swarm Optimization and Heuristic Dynamic Programming,” IEEE Transactions on Power Systems, Vol. 29, No. 5, pp. 2539-2548, Sep. 2014. [Online]. Available: http://dx.doi.org/ DOI: 10.1109/TPWRS.2014.2305977

[7] Y. Qi and Q. Meng, “The Application of Fuzzy PID Control in Pitch Wind Turbine,” Proc. of International Conference on Future Energy, Environment, and Materials, Energy Procedia, Vol.16, pp. 1635-1641, 2012. [Online]. Available: http://dx.doi.org/ doi:10.1016/j.egypro.2012.01.254

[8] P. Garasi, M. Watanabe and Y. Mitani, "Power smoothing of wind turbine generator using Fuzzy-PI pitch angle controller," Proc. of IEEE Power Engineering Conference (AUPEC), pp. 1-5, 2014. [Online]. Available: http://dx.doi.org/ DOI: 10.1109/AUPEC.2014.6966531

[9] S. Simani and P. Castaldi, “Data driven and adaptive control applications to wind turbine benchmark model,” Control Engineering Practice, Vol. 21, pp.1678-1693, 2013. [Online]. Available: http://dx.doi.org/ doi:10.1016/j.conengprac.2013.08.009

[10] V. Torczon, "On the convergence of pattern search algorithms," SIAM Journal on optimization, Vol. 7,

No. 1, pp.1-25, 1997. [Online]. Available: http://dx.doi.org/ DOI:10.1137/S1052623493250780

[11] E. D. Dolan, R. M. Lewis and V. Torczon, "On the local convergence of pattern search," SIAM J Optimiz, Vol.14, No. 2, pp. 567–83, 2003. http://dx.doi.org/ DOI:10.1137/S1052623400374495

[12] Y. Bao, H. Zhongyi and X. Tao, "A PSO and pattern search based memetic algorithm for SVMs parameters optimization," Neurocomputing, Vol. 117, pp. 98-106, 2013. [Online]. Available: http://dx.doi.org/10.1016/j.neucom.2013.01.027

[13] R. K. Sahu, S. Panda and G. T. Chandra Sekhar, "A novel hybrid PSO-PS optimized fuzzy PI controller for AGC in multi area interconnected power systems," International Journal of Electrical Power & Energy Systems, Vol. 64, pp.880-893, 2015. [Online]. Available: http://dx.doi.org/10.1016/j.ijepes.2014.08.021

[14] S. Panda. and N.P. Padhy, "Comparison of particle swarm optimization and genetic algorithm for FACTS-based controller design," Applied Soft Computing, Vol. 8, No.4, pp.1418-1427, 2008.

[15] R. Romero, R.A. Gallego and A. Monticelli, "Transmission system expansion planning by simulated annealing," IEEE Transactions on Power Systems, Vol. 11, No.1, pp.364-369, 1996.

[16] S. Okdem and D. Karaboga, "Routing in wireless sensor networks using ant colony optimization," In IEEE Conference on Adaptive Hardware and Systems, 2006. *AHS 2006.* pp. 401-404, June 2006.

[17] U.K. Rout, R.K. Sahu and S. Panda, " Design and analysis of differential evolution algorithm based automatic generation control for interconnected power system," Ain Shams Engineering Journal, Vol.4, No.3, pp. 409-421, 2013.

[18] S. Arora and S. Singh, "The firefly optimization algorithm: convergence analysis and parameter selection," International Journal of Computer Applications, Vol. 69, No. 3, pp. 48-52, 2013.

[19] S. Behera, S. Sahoo and B. B. Pati, "A review on Optimisation Algorithms and application to Wind Energy Integration to Grid," Journal of Renewable & Sustainable Energy Reviews, Vol. 48, pp. 214-227, 2015. http://dx.doi.org/10.1016/j.rser.2015.03.066

[20] S. A. Taher and S. Mansouri, "Optimal PI controller design for active power in grid-connected SOFC DG system," International Journal of Electrical Power and Energy Systems, Vol. 60, pp. 268–274, 2014. [Online]. Available: http://dx.doi.org/10.1016/j.ijepes.2014.02.010

[21] R. Gagnon, B. Saulnier and A. Forcione (Hydro-Quebec), "Wind Farm – DFIG Average Model in MATLAB SIMULINK," The Mathworks. Inc. MATLAB R2010a.

[22] E. Kamal and A. Aitouche, "Robust fault tolerant control of DFIG wind energy systems with unknown inputs," Renewable Energy, Vol. 56, pp. 2-15, 2013. http://dx.doi.org/10.1016/j.renene.2012.10.024

[23] J. G. Slootweg, H. Polinder and W. L. Kling, "Representing wind turbine electrical generating systems in fundamental frequency simulations," IEEE Transactions on Energy Conversion, Vol. 18, No.4, pp. 516-524,2003. http://dx.doi.org/10.1109/TEC.2003.816593

[24] T. Ackerman, Wind power in Power System, 1st Ed., John Wiley & Sons. Ltd, England: 2005.

[25] RT-Lab Professional. [Online] Available at http://www.opal-rt.com/product/rtlab-professional.

[26] S. Behera and B. B. Pati, "Simplified Simulation of Wind Speed Model using MATLAB Simulink Blocksets for Controller Performance Studies", Proc. of IEEE International Conference on Electrical, Electronics, Signals, Communication and Optimization (EESCO), issue-4, pp. 1839-1843, 2015.