Aalborg Universitet

AALBORG UNIVERSITY

Design of a pitch controller for a wind turbine using hybrid mean-variance mapping optimization

Behera, Sasmita; Sahoo, Subham

Published in:
ECTI Transactions on Electrical Engineering, Electronics, and Communications

DOI (link to publication from Publisher):
10.37936/ECTI-EEC.2021193.222600

Creative Commons License
CC BY-NC-ND 4.0

Publication date:
2021

Document Version
Publisher's PDF, also known as Version of record

Link to publication from Aalborg University

Citation for published version (APA):
Behera, S., & Sahoo, S. (2021). Design of a pitch controller for a wind turbine using hybrid mean-variance mapping optimization. *ECTI Transactions on Electrical Engineering, Electronics, and Communications*, *19*(3), 298-311. https://doi.org/10.37936/ECTI-EEC.2021193.222600

General rights
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal -

Take down policy
If you believe that this document breaches copyright please contact us at vbn@aub.aau.dk providing details, and we will remove access to the work immediately and investigate your claim.

Downloaded from vbn.aau.dk on: December 05, 2025

Design of a Pitch Controller for a Wind Turbine Using Hybrid Mean-Variance Mapping Optimization

Sasmita Behera[1†] and **Subham Sahoo**[2], Non-members

ABSTRACT

A variable-speed wind energy conversion system (WECS) has the advantage of extracting more power from the time-varying wind. To achieve this, the pitch angle is controlled to maintain the speed of the turbine and hence the generated power at a constant level, while reducing mechanical stress on the turbines. In this work, a proportional-integral (PI) controller is used for pitch angle control. The optimal PI control gains K_p and K_i are tuned by the hybrid mean-variance mapping optimization (MVMO-SH) technique, particle swarm optimization (PSO), and a genetic algorithm (GA). Different fitness evaluation criteria and optimization techniques are compared, and the performances of optimal controllers presented in the time domain. The results reveal that MVMO-SH achieves the minimum error criteria within a shorter time. The optimal controller design gives an error of less than 10^{-6} in the region for which it is tuned. The performance of the optimal PI controller designed for one operating condition is tested in different cases of wind gust, random variation of wind, and disturbance from the grid side to mitigate line to ground fault. The performance of the controller is shown to be satisfactory in all the cases studied.

Keywords: Pitch Control, Proportional-Integral, Mean-Variance Mapping Optimization, Particle Swarm Optimization, Genetic Algorithm, Doubly-Fed Induction Generator, Wind Gust, Fault

1. INTRODUCTION

A decline in fossil fuel stocks and the ability to meet power needs while considering the economic and environmental aspects, has forced governments to turn their attention to wind energy conversion as a renewable energy source. The doubly-fed induction generator (DFIG) with a low-rated power converter [1] is suitable for variable-speed operation and captures more power than fixed-speed systems. The frequency must match that of the grid when connected to it. The speed is affixed mechanically by pitching the blades on the turbine shaft. The pitch control is initiated when the generator power or angular speed of the wind turbine touches its rated value to control the power delivered to the grid and reduce mechanical stress during wind gusts [2]. For below-rated wind speed, the torque is controlled to maximize the power coefficient of the wind turbine and the power captured from the wind.

Proportional-integral (PI) controllers are simple in design and perform well with a low overshoot percentage and fewer maintenance costs when designed optimally [3]. Specifically, the WECS proportional-integral-derivative (PID) controller does not provide any substantial improvement [4] in comparison to the PI controller. Furthermore, the proportional controller (P) alone cannot compensate for the effect of constant disturbances [5]. Thus, to achieve the requisite behavior of the control loop to tune the pitch controller parameters, different optimization techniques have been used in the DFIG such as bacteria foraging optimization (BFO), particle swarm optimization (PSO), the genetic algorithm (GA), and differential evolution (DE) [1, 6–13]. Neural networks (NNs) have been used for pitch angle control as presented in [14], while three different controller structures for pitch control have been reported in [15]. The PSO, mean-variance optimization (MVO), and NN techniques have been compared to testify for DFIG with FACTS devices to provide a unified controller parameter design [16]. Jauch *et al.* represented a wind turbine as a second-order system [17]. These researchers obtained a different set of pitch controller gains for different operating regions of wind speed and studied the effect using a realistic model of the Northern European power system.

In [18] a PI pitch controller for a large wind turbine generator was designed and analyzed. Fuzzy logic for pitch control has also been demonstrated in [19]. Switching between a fuzzy and PID controller, depending on the region of operation, has been reported in [20] but, the steady-state

Manuscript received on October 23, 2019 ; revised on November 7, 2020 ; accepted on July 4, 2021. This paper was recommended by Associate Editor Kriangkrai Sooksood.

[1]The author is with the Department of Electrical and Electronics Engineering, Veer Surendra Sai University of Technology, Burla, India.

[2]The author is with the Department of Energy, Aalborg University, Aalborg, Denmark.

[†]Corresponding author: sasmitabehera2000m@gmail.com

©2021 Author(s). This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivs 4.0 License. To view a copy of this license visit: https://creativecommons.org/licenses/by-nc-nd/4.0/.

Digital Object Identifier 10.37936/ecti-eec.2021193.222600

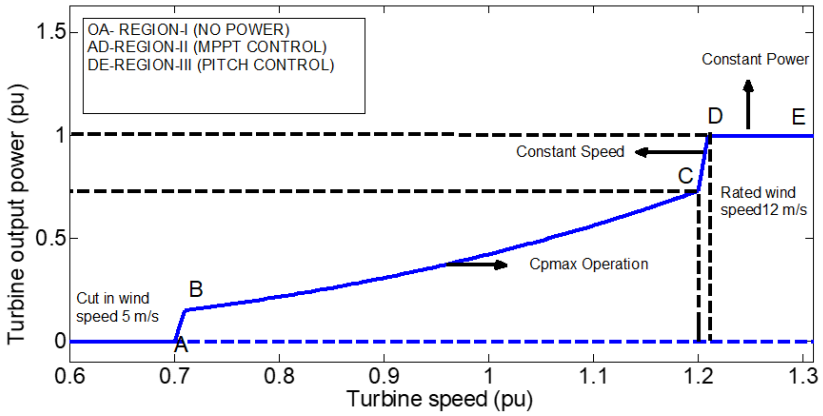

Fig. 1: *Wind turbine power vs. turbine angular speed and regions of operation.*

error is around 10%. However, swarm optimization has furnished an agreeable controller performance. Recently, a hybrid version of mean-variance mapping optimization (MVMO-SH) has been presented [21]. Apart from being a swarm intelligence-based procedure, it incorporates local search and multi-parent crossover strategies to increase diversity thus balancing exploration and exploitation. Mutation via the mapping function and crossover are also enhanced. Unlike existing swarm-based optimization algorithms, MVMO-SH does not need a large population. Hence, a single particle approach may be sufficient for less challenging optimization problems. The performance of this algorithm has been tested for many standard unimodal, multimodal, and mixed functions and compared numerically with other optimization techniques like the PSO, DE, and GA [23]. A small archive size is required and this can be adjusted, thereby adding to its advantages over other techniques. The MVMO technique has been successfully applied in model identification [22] including online optimization-based control and the optimization of transmission expansion planning [23] and other tuning problems in power systems [24, 25]. However, this new technique has not been applied to the pitch control of a DFIG, motivating further research to be carried out.

The objectives of the presented work are:

- To tune the PI controller in the pitch angle control loop using a hybrid (MVMO-SH) technique to obtain the optimal PI control gains K_p and K_i for a particular range of wind speed.
- To compare the achievement, i.e., the objective function, based on errors such as the integral of absolute error (IAE), integral of squared error (ISE), integral time absolute error (ITAE), and integral time squared error (ITSE), etc., defined in [27] for better time-domain performance of pitch control.
- It has been reported that a set of optimal PI controller gains obtained at a specific wind speed above the rated level is not optimal for another [7]. Therefore, this current study aims to optimize the controller gains to achieve a large step variation in wind speed during simulation from below the rated level to above it.
- To compare the performance of MVMO-SH quantitatively with PSO and GA in terms of settling time and peak overshoot to obtain a lower error level for the same number of generations and swarm size in a similar optimization framework.
- To test the performance of the optimal PI controller design for one wind gust operation, random wind variation in different operating regions, and line to ground faults on the grid. In all the cases studied, the controller design performs satisfactorily, with less steady-state error, overshoot, and settling time.

Section 2 of this paper embodies an explanation of the system and algorithm. In Section 3, the optimizations are compared while the performance of the optimal pitch controller design is discussed in Section 4. Section 5 provides the concluding remarks.

2. SYSTEM MODEL AND PITCH CONTROLLER

2.1 Regions of Wind Turbine Operation

The wind turbine operation is divided into three regions, as indicated in Fig. 1.

In region I, the wind speed is below the cut-in value and hence, no power is generated. In region II, where the partial load is located, the wind speed is more than the cut-in value but less than the rated level. The prime control objective in this region is to maximize the power generated by the wind turbine. To achieve this, the power coefficient C_p should be set to maximum $C_{p\,\max}$, thus maximizing the torque. BC is the locus of maximum power at different wind speeds. At D with a power output of 1 pu, region III begins, where the full load is located. From D to E, or sometimes C to E, wind speed exceeds the rated level but is lower than the cut-out value [2]. In this region the generated power P_g is limited to its

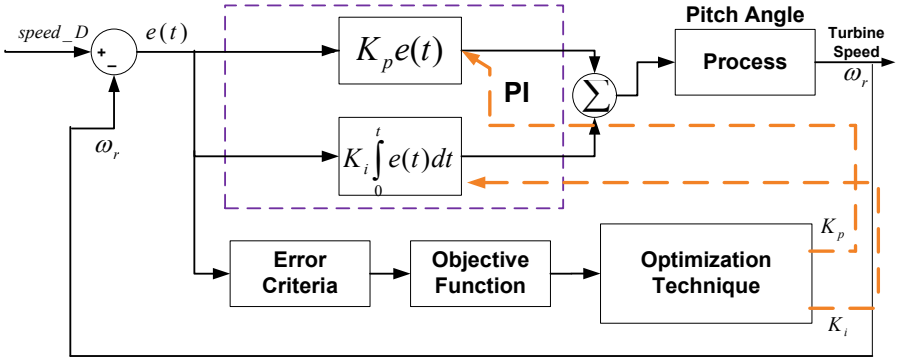

Fig. 2: *Proportional-integral (PI) pitch controller.*

rated value $P_{g,\text{rated}}$ while the turbine rotor speed ω_r is set at 1.21 pu. This is attained by pitch control. In this paper, the research is restricted primarily to proposing an optimal PI pitch controller for region III.

The phasor model of the DFIG based WECS connected to the grid proposed in [26] is used in this paper to design the pitch controller. The structure of the pitch angle controller is shown in Fig. 2.

The turbine rotor speed ω_r is provided by sensors. The reference turbine *speed_D* at 1.21 pu is based on that of the generator when the wind speed is in region III. The maximum power extraction is 9 MW from six identical WECS at a wind speed of 12 m/s with an individual contribution of 1.5 MW. The wind speed is taken as a step signal from 8 m/s to 14 m/s while the PI gains K_p and K_i are obtained by running the optimization program linking the model. The pitch control command is given from the controller to the pitch drive which then sets the pitch angle. Thus, the speed of the rotor in the wind turbine system is controlled. The turbine rotor speed response is observed in accordance with different error criteria and optimization techniques. Some of the mathematical relations of the pitch control dynamics are presented under different subsections portraying the model [2, 7].

2.1.1 Aerodynamic model

The power transfer to the wind turbine shaft is expressed as:

$$P_{\text{windturbine}} = C_p \cdot P_{\text{air}} = \frac{1}{2} C_p \rho A V_a^3 = \frac{1}{2} \rho C_p \pi R^2 V_a^3 \tag{1}$$

where P_{air} is the power available in wind (W), C_p is the power coefficient which is $f(\lambda, \beta)$, ρ is air density (1.225 kg/m^3 at 15°C and normal pressure), A is the area swept by blades (m^2) $= \pi R^2$, and V_a is the wind velocity (m/s).

The mechanical power at the shaft for any wind speed can be expressed as,

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

where the tip speed ratio,

$$\lambda = \frac{\omega_r R}{V_a} \tag{3}$$

where ω_r is the angular speed of the turbine shaft and R is radius of blades in m.

C_p can be given by,

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

and

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

According to Eq. (2), the mechanical power of the shaft varies with the cube of the rotor speed $P \propto \omega^3$. When the wind speed rises λ decreases and hence, P increases as C_p depends on λ and the pitch angle β. In order to limit the power delivered to the grid P_g to a constant value $P_{g,\text{rated}}$, the pitch angle is increased in region III (the full load region) which decreases C_p and thus controls P.

2.1.2 Drive train model

Power transmission from the wind turbine shaft to the generator is performed by the drive train. The drive train considered here is a one mass model for simplicity. The mechanical equation of the rotor motion is given as:

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

where H is the equivalent inertia constant of the wind turbine and generator rotor in s and T_e is the electrical torque. The mechanical torque T_m expression developed by the turbine is,

$$T_m = \frac{P_{\text{windturbine}}}{\omega_r} \tag{6}$$

2.1.3 Generator model

The DFIG is coupled to the wind turbine. The rotor is interfaced with the grid through a back-to-back voltage source converter (VSC) whereas the stator is directly connected. All variables and equations of the models in the three-phase ABC reference frame are converted to the synchronously rotating DQ0 or *d-q* reference frame. The control is easy with the two orthogonal d (direct), q (quadrature) components which behave as dc quantities. The dynamic modeling equations of the DFIG with some assumptions [5] are as follows:

$$V_{ds} = -r_s i_{ds} + \frac{d\lambda_{ds}}{dt} - j\omega_s \lambda_{qs} \tag{7}$$

$$V_{qs} = -r_s i_{qs} + \frac{d\lambda_{qs}}{dt} + j\omega_s \lambda_{ds} \tag{8}$$

$$V_{dr} = -r_r i_{dr} + \frac{d\lambda_{dr}}{dt} - js\omega_s \lambda_{qr} \tag{9}$$

$$V_{qr} = -r_r i_{qr} + \frac{d\lambda_{qr}}{dt} + js\omega_s \lambda_{dr} \tag{10}$$

and slip of the rotor (s),

$$s = \frac{\omega_s - \omega_r}{\omega_s} \tag{11}$$

where V_{ds} and V_{qs} are the stator voltages in *d-q* reference frame, V_{dr} and V_{qr} are the rotor voltages in *d-q* reference frame, i_{ds} and i_{qs} are the stator currents in *d-q* reference frame, i_{dr} and i_{qr} are the rotor currents in *d-q* reference frame, r_s is the stator resistance, r_r is the rotor resistance, ω_s is the rotational angular speed of the synchronously rotating reference frame, λ_{ds} and λ_{qs} are the flux linkage of the stator in *d-q* axes, and λ_{dr} and λ_{qr} are the flux linkage of the rotor in *d-q* axes.

The electromagnetic torque generated by the DFIG is given as

$$T_e = \lambda_{dr} i_{qr} - \lambda_{qr} i_{dr} \tag{12}$$

2.1.4 Formulation of the control problem

The pitch drive is an electric or pneumatic device, taking the control command and setting the blades around their longitudinal axes at the desired pitch angle. The collective pitch control (CPC) is set at the same angle for all blades using a single drive. The pitch control command is generated by a PI controller from an error in the angular speed of the wind turbine rotor. Since the blades are long, the pitch drive moves them at a slow rate of 2° per second in order to reduce vibration and actuator fatigue.

The error is given as:

$$e(t) = (speed\_D - \omega_r) \tag{13}$$

where *speed_D* is the desired turbine speed in pu = 1.21 pu.

The pitch control command $u(t)$ is obtained from the PI controller as:

$$u(t) = K_p \cdot e(t) + K_i \int e(t)dt \tag{14}$$

The feedback control system must swiftly nullify the errors $e(t)$, among variables and the desired values. Thus, the controller design problem is caused by minimization. Therefore, any criterion used to measure the quality of system response must take into account the variation over the whole range of time. Four basic criteria are in common use [27]:

- Integral of absolute error

$$\text{IAE} = \int |e(t)|\, dt \tag{15}$$

- Integral of squared error

$$\text{ISE} = \int e^2(t)dt \tag{16}$$

- Integral of time multiplied by absolute error

$$\text{ITAE} = \int t\,|e(t)|dt \tag{17}$$

- Integral of time multiplied by squared error

$$\text{ITSE} = \int te^2(t)dt \tag{18}$$

In Eqs. (14) to (18), time t is the time taken for the simulation. The integration is carried out for the time the simulation model is run, namely from 0 to t. These criteria are used here as fitness functions and minimized to optimize the subsequent gains of the controller with the search region $K_{p\min} \leq K_p \leq K_{p\max}$ and $K_{i\min} \leq K_i \leq K_{i\max}$, where $K_{p\min}$, $K_{i\min}$ are lower and $K_{p\max}$, $K_{i\max}$ are upper boundaries of the gains.

The reduction in steady-state error using the above criteria has been proven to satisfy other specifications in the time domain such as peak overshoot, rise time, settling time, and steady-state error in the output of many optimization problems considered in the literature [1, 6–13, 16, 23, 24, 28]. When addressing a problem, one criterion may outperform another. The ISE and ITAE criteria exhibit better performance compared to IAE and ITSE [7, 28]. The ISE criterion integrates the square of the error over time. It picks large errors and penalizes them more than smaller ones since the square of a large error will be much greater. The control system designed by the ISE acts fast, but small persistent errors remain, imparting low amplitude oscillation. Whereas the ITAE persistent weight errors are heavier than those at the start of the response. The ITAE tuning produces systems with faster settling than the ISE

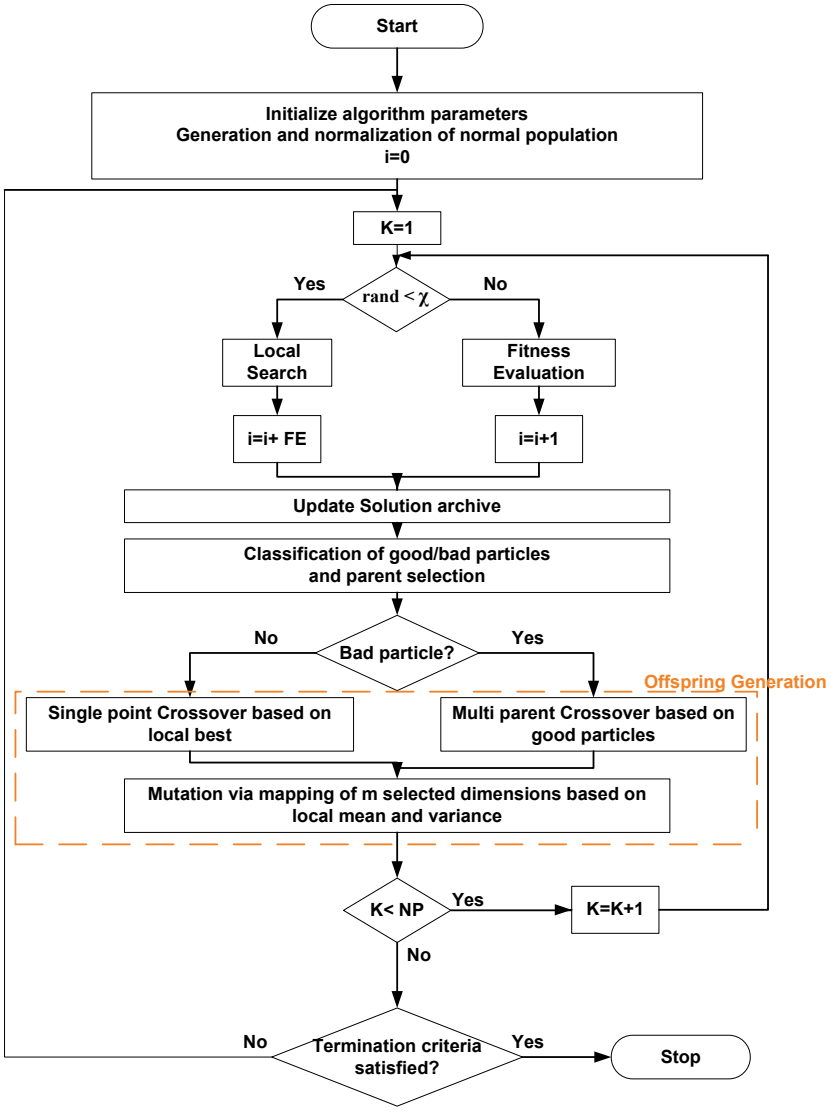

Fig. 3: *Overall procedure for the hybrid MVMO [25], where K is particle counter, i is fitness evaluation counter, FE is count of total fitness evaluations, and rand is random number in (0, 1).*

tuning method. Nevertheless, the IAE and ITSE are used by the researchers in [28]. Therefore, all the four objective functions defined in Eqs. (15) to (18) have been tested using this problem to verify performance.

2.2 Hybrid Mean-Variance Mapping Optimization

The MVMO-SH technique [21] has been applied to solve various real-world optimization problems [22–25]. The algorithm starts with the initialization of the total number of particles (NP). Next, the variables are normalized to $(0, 1)$. The mutation operation via mapping then ensures non-violation of the offspring beyond the search boundaries and concentrates on the mean value. The mapping operation is performed by adjusting the mean and shape variables. The shape variables depend on the variance. Search diversity is maintained by focusing on the best local measurement obtained until the required iteration. One significant feature of this algorithm is that the memory space acquired is small and the memory or archive size adjustable, without losing any substantial search diversity. Prior to fitness evaluation, the variables are de-normalized. The remaining steps are presented in the flowchart in Fig. 3.

After de-normalization, fitness evaluation is tested. In this work, the objective fitness functions are unconstrained as defined by Eqs. (15) to (18).

On successive iterations, each particle has continually updated the solution archive. The best offspring are stored in a downward arrangement of fitness in the archive. The archive is updated if the new solution is found to be better.

The particles are segregated into good particles (GP) and bad particles, NP-GP from the total particles. Each particle is assessed twice and the local

best is selected to be the parent of the next offspring. For each bad particle, the parent is synthesized using a multi-parent strategy.

The parent-child vector set is created using a combination of the inherited particles from the parent vector and a similar dimensional set (by selection) to undergo mutation through mapping based on the mean and variances designed from the solution archive.

Each selected dimension value for the child vector is calculated by

$$x_r = h_x + (1 - h_1 + h_0) \cdot x_r^* - h_o \tag{19}$$

where x_r^* is a randomly generated number with uniform distribution between (0, 1) for the r^{th} variable searched for, where the values of h signify the mapping transformation calculated as

$$h\left(x_{\text{mean}}, s_1, s_2, x\right) = x_{\text{mean}}\left(1 - e^{-x \cdot s_1}\right) + (1 - x_{\text{mean}}) e^{-(1-x)\cdot s_2} \tag{20}$$

where h_x, h_1, and h_0 are the mapping function outputs accountable for $h_x = h\left(x = x_r^*\right)$, $h_0 = h\left(x = 0\right)$, $h_1 = h\left(x = 1\right)$, hence, x_r is in [0, 1].

s_r is the shape factor with v_r as variance and f_s is the shaping factor

$$s_r = -\ln\left(v_r\right) \cdot f_s \tag{21}$$

At first, the mean x_{mean} corresponds to the initial value of x, and variance is set to 1 which indicates $s_r = 0$. However, over the iterations, the solution archive updates repeatedly, where the inputs and outputs both lie [0, 1]. The search exploration can be altered by changing the shape factors and the mean since they affect the shape of the mapping function. Furthermore, the shape parameters s_{r1} and s_{r2} are updated by the steps given:

$s_{r1} = s_{r2} = s_r$
if $s_r > 0$ **then**
  $\Delta d = 1 + \Delta d_0 + 2 \cdot \Delta d_0 \cdot (\text{rand} - 0.5)$
  if $s_r > d_r$ **then**
    $d_r = d_r \cdot \Delta d$
  else
    $d_r = \dfrac{d_r}{\Delta d}$
  end if
  if rand ≥ 0.5 **then**
    $s_{r1} = s_r;\ s_{r2} = d_r$
  else
    $s_{r1} = d_r;\ s_{r2} = s_r$
  end if
end if

The above steps guarantee exploitation without stagnation for an asymmetric characteristic of the mapping function when diverse s_{r1} and s_{r2} are taken, thereby enhancing the searching capability and managing the zero variance. Initial values of the factor d_r are about 1–3 of the imparted improved results [24].

The number of particles (NP) is empirically initialized using NP $= 20 \cdot D$ particles, where D is the dimension. But a fewer number of particles can perform well to solve simple problems [21]. The archive size can be adjusted as required.

The algorithm is performed with the probability that the local search can be improved, given as $\gamma = (1.5 \cdot D)/100$.

The proportion of GP is updated by:

$$\text{GP} = \text{round}\left(\text{NP} \cdot g_p^*\right) \tag{22}$$

$$g_p^* = g_{p_{ini}}^* + \alpha\left(g_{p_{final}}^* - g_{p_{ini}}^*\right) \tag{23}$$

$$\alpha = \frac{i}{i_{final}} \tag{24}$$

where α is the adaptive factor.

During the preliminary stage of the algorithm, each particle is separately evaluated for at least two function evaluations, and the best local solution achieved so far (corresponding to the first position in its specific solution archive) is designated as the parent of the next offspring. The same best-based local parent procedure is carried out on good particles, whereas the multi-parent strategy for each bad particle p, gives the parent x_p^{parent} as follows:

$$x_p^{\text{parent}} = x_k + \eta(x_i - x_j) \tag{25}$$

where x_i, x_j, and x_k represent the first (global best), last, and a randomly selected intermediate particle of the good particle group, respectively.

η is calculated accordingly:

$$\eta = 2\left(\text{rand} - \text{shift}\right) \tag{26}$$

$$\text{shift} = 0.5\left(1 - \alpha^2\right) \tag{27}$$

where α has been defined earlier in Eq. (23).

The number m representing mutation is decreased to reduce computational burden on convergence as

$$m = \text{round}(m_{final} + \text{rand}(m^* - m_{final})) \tag{28}$$

where

$$m^* = \text{round}(m_{ini} - \alpha(m_{ini} - m_{final})) \tag{29}$$

The mutation is guided by $m_{ini} = \text{round}(D/2) + 1$, $m_{final} = 1$.

The scaling factor f_s is adjusted for accuracy as

$$f_s = f_s^*\left(1 + \text{rand}\right) \tag{30}$$

$$f_s^* = f_{s_{ini}}^* + \alpha^2\left(f_{s_{final}}^* - f_{s_{ini}}^*\right) \tag{31}$$

Table 1: *Parameters of the wind turbine and DFIG.*

Parameters	Values
Installed capacity	1.5 MW
Cut-in wind speed	5 m/s
Rated wind speed	12 m/s
Rated turbine speed	1.21 pu
Cut-out wind speed	25 m/s
Maximum power coefficient	0.48
Extreme limit of pitch angle	45°
Slope of pitch change	2°/s
Frequency	60 Hz
Number of generator poles	6
Terminal voltage of generator	575 V
Equivalent inertia constant	5.04 s

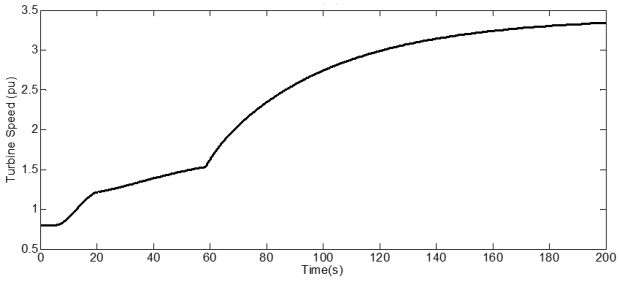

Fig. 4: *Turbine speed response without a controller.*

In all the equations, rand represents random numbers with uniform distribution in the range (0, 1).

The parameter tuning of the MVMO-SH algorithm and choice of a particular value for each parameter is presented in a logical way in Section 4 to address the problem under study.

3. CASE STUDY

The optimization algorithms and model are run using MATLAB® software version 2009b. The model parameters adopted are shown in Table 1. The performance of MVMO-SH is compared with PSO and GA. The various parameters for MVMO-SH, PSO, and GA applied to obtain the optimal controller gains are deliberated in Table 2.

The search bounds for both K_p and K_i were fixed after many trial runs. The wind speed deviation initiated at 5 s from 8 m/s to 14 m/s. The turbine speed response without any controller is shown in Fig. 4. The speed ω_r increases gradually, demonstrating that the system is unstable without a controller.

Hence, to achieve the desired good tracking speed and fast response, a PI controller is used in this study.

3.1 Tuning the MVMO-SH Parameters

The MVMO-SH has several parameters that should be chosen judiciously to address a specific problem; the setting of which the implemented algorithm performance depends. The specific control parameters consist of the variable increment Δd_o, the initial proportion of good particles $g^*_{p_{ini}}$,the final proportion of good particles $g^*_{p_{final}}$, initial value of scaling factor $f^*_{s_{ini}}$, and final value of the scaling factor $f^*_{s_{final}}$. A sequence of observations was used to properly tune the MVMO-SH parameters to optimize the PI parameters employing the ITAE objective function. Table 3 presents a tabulation of the best effect by varying the control parameters after 20 independent runs. It is evident from Table 2 that the best settings are $g^*_{p_{ini}} = 0.75$, $g^*_{p_{final}} = 0.25$, $f^*_{s_{ini}} = 1$, and $f^*_{s_{final}} = 15$ for the population = 10 and iterations = 25, depicted in bold font. After tuning the parameters for N = 10 and iterations = 25, the same procedure was performed for N = 12 and iterations = 25 which showed no change in the values of parameters for MVMO-SH. The variation in population size (NP) and iterations were also observed but that part is excluded for a concise presentation. Increasing the NP from 10 to 12 and iterations from 20 to 25 improves the average, maximum, and standard deviation values slightly at the expense of a significant increase in the computation time as can be observed from Tables 4 and 5. A further increase would be unlikely to result in any appreciable overall improvement.

Table 2: *Parameters of the optimization algorithms.*

Algorithm	Parameter	Value
MVMO-SH	Data set size	10/12
	No. of iterations	20/25
	Δd_o	0.05
	$g^*_{p_{ini}}$	0.75
	$g^*_{p_{final}}$	0.25
	$f^*_{s_{ini}}$	1
	$f^*_{s_{final}}$	15
PSO	Swarm size	10/12
	Max. no. of generations	20/25
	c_1	2.0
	c_2	2.0
	w_{max}	0.9
	w_{min}	0.4
GA	Population size	10/12
	Max. no. of generations	20/25
	p_{cross}	0.5
	p_{mut}	0.01
Search space bounds [min, max]	K_p	[400, 2000]
	K_i	[30, 50]

3.2 Parameter Optimization of the PI Controller

The controller gains K_p and K_i perform optimally using various error-derived objective functions for a wind speed deviation from 8 to 14 m/s. The simulation model was run for 50 s during optimization.

Table 3: *Parameter tuning of the MVMO-SH.*

Parameters		Min.	Avg.	Max.	S.D.	Others
Δd_o	0.03	128.268	128.337	128.410	0.121	$g^*_{p_{ini}}=0.7$
	0.04	128.193	128.279	128.328	0.118	$g^*_{p_{final}}=0.2$
	0.05	128.154	128.196	128.289	0.112	$f^*_{s_{ini}}=2$
	0.06	128.183	128.265	128.306	0.115	$f^*_{s_{final}}=20$
	0.07	128.213	128.284	128.342	0.124	
$g^*_{p_{ini}}$	0.7	128.176	128.201	128.248	0.115	$\Delta d_o=0.05$
	0.75	128.149	128.194	128.236	0.109	$g^*_{p_{final}}=0.2$
	0.8	128.182	128.218	128.256	0.114	$f^*_{s_{ini}}=2$
	0.85	128.187	128.226	128.274	0.118	$f^*_{s_{final}}=20$
	0.9	128.196	128.234	128.286	0.121	
$g^*_{p_{final}}$	0.2	128.158	128.238	128.336	0.115	$\Delta d_o=0.05$
	0.25	128.138	128.198	128.274	0.105	$g^*_{p_{ini}}=0.75$
	0.3	128.164	128.216	128.326	0.118	$f^*_{s_{ini}}=2$
	0.35	128.176	128.260	128.358	0.123	$f^*_{s_{final}}=20$
	0.4	128.189	128.244	128.392	0.128	
$f^*_{s_{ini}}$	**1**	128.126	128.212	128.324	0.098	$\Delta d_o=0.05$
	1.5	128.139	128.222	128.342	0.104	$g^*_{p_{ini}}=0.75$
	2	128.154	128.236	128.356	0.109	$g^*_{p_{final}}=0.25$
	2.5	128.173	128.249	128.374	0.114	$f^*_{s_{final}}=20$
	3	128.196	128.265	128.368	0.118	
$f^*_{s_{final}}$	**15**	128.116	128.186	128.253	0.086	$\Delta d_o=0.05$
	16	128.123	128.184	128.265	0.092	$g^*_{p_{ini}}=0.75$
	17	128.129	128.190	128.274	0.099	$g^*_{p_{final}}=0.25$
	18	128.135	128.217	128.287	0.102	$f^*_{s_{ini}}=1$
	19	128.136	128.227	128.313	0.107	
	20	128.139	128.236	128.358	0.112	

The program was executed 20 times since randomness is inherent in each of the algorithms and the finest gains are shown. The NP and maximum number of iterations I were increased to observe their effect. Tables 4 and 5 present a tabulated comparison of the two cases of NP and I, for different objective functions. The best response in terms of rise time is obtained using ITAE as the objective function for the MVMO-SH with NP = 12 and iterations = 25 in the final row of Table 5. The PSO and GA techniques were also tested using a similar search space, population, and generation to the MVMO-SH, with the comparative results presented in Table 6 for the second set of NP and I.

In Fig. 5, ω_r the optimal gains are determined using the MVMO-SH with different objective functions. As in Table 5, these are plotted to compare the step deviation in wind speed from 8 to 14 m/s. As can be observed, despite overshoot, the response for optimized gains using the ITAE criteria exhibits the least settling time. The error is in the range of 10^{-7} which, although well within the requirement, is slightly more than that found in the ITSE error criteria. For a set point of 1.21 pu, steady-state error variation in the range of 10^{-7} is of little concern. This optimal PI controller for pitch control with $K_p = 547.82$ and $K_i = 44.36$ has been tested under ten different dynamic conditions of wind speed

Table 4: *Comparison of the different objective functions NP = 10, I = 20.*

Objective function	K_p	K_i	Value of objective function	Run time (s)	Settling time (s)	Steady-state error
ISE	610.34	42.14	4.1991	304.36	68.37	2.93×10^{-6}
IAE	584.96	42.85	10.2458	302.74	67.64	1.77×10^{-6}
ITSE	592.36	44.28	52.5279	294.74	67.55	1.53×10^{-6}
ITAE	548.47	45.27	128.1209	296.25	67.49	4.85×10^{-7}

Table 5: *Comparison of the different objective functions NP = 12, I = 25.*

Objective function	K_p	K_i	Value of objective function	Run time (s)	Settling time (s)	Steady-state error
ISE	602.28	42.16	4.1973	503.48	68.17	2.98×10^{-6}
IAE	587.89	42.72	10.2431	502.46	67.72	2.04×10^{-6}
ITSE	591.47	44.23	52.3974	468.45	67.55	1.53×10^{-7}
ITAE	547.82	44.36	128.1162	473.58	67.24	6.54×10^{-7}

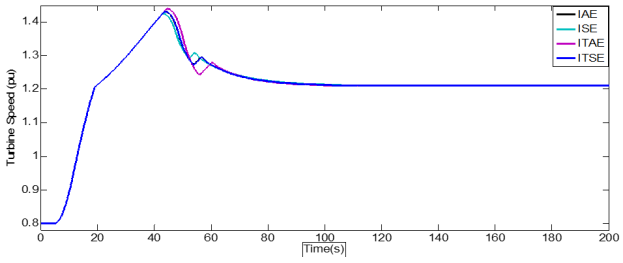

Fig. 5: *Comparison of Turbine Speed Response for Different Objective Functions with MVMO.*

change and transients under grid fault to verify robustness. The settling time and steady-state error are summarized in Table 7.

3.3 Controller Performance Analysis under Transient Conditions

Case 1: Random wind speed variation within the range of 8 to 14 m/s

Although the controller has been optimized for step deviation, the wind is intermittent in nature and varies slowly. Therefore, the performance of the controller was observed for random variation within 8 to 14 m/s. The wind speed profile and the variation in wind turbine speed response and time are shown in Figs. 6 and 7, respectively. The settling time and steady-state error are presented in Table 7.

Since the wind speed is taken at random, in every run the steady-state error and settling time differ. Two to three cases of random variation in this range were studied and the steady-state error remained within the range of 10^{-4}.

Table 6: *Comparison of the different optimization algorithms.*

Objective function	Optimization Technique	K_p	K_i	Value of objective function	Run time (s)	Settling time (s)	Steady-state error
ISE	MVMO-SH	602.28	42.16	4.1973	503.48	68.17	2.98×10^{-6}
	PSO	631.42	42.83	4.2735	624.96	68.66	4.11×10^{-6}
	GA	640.64	43.01	4.2831	642.64	68.84	4.56×10^{-6}
IAE	MVMO-SH	587.89	42.72	10.2431	502.46	67.72	2.04×10^{-6}
	PSO	622.91	42.83	10.6355	634.19	68.47	3.59×10^{-6}
	GA	640.74	43.42	10.7253	672.04	68.70	4.17×10^{-6}
ITSE	MVMO-SH	591.47	44.23	52.3974	468.45	67.55	1.53×10^{-7}
	PSO	597.26	44.31	53.1841	579.04	67.61	1.66×10^{-6}
	GA	598.34	44.15	53.1926	616.08	67.64	1.75×10^{-6}
ITAE	MVMO-SH	547.82	44.36	128.1162	473.58	67.24	6.54×10^{-7}
	PSO	571.71	46.18	129.1162	582.86	67.58	6.96×10^{-7}
	GA	583.92	46.69	129.4283	608.52	67.58	7.77×10^{-6}

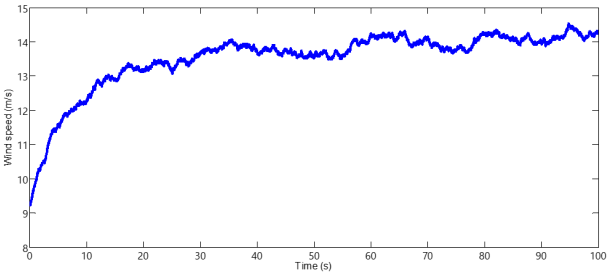

Fig. 6: *Wind speed profile for random variation within the range of 8 to 14 m/s.*

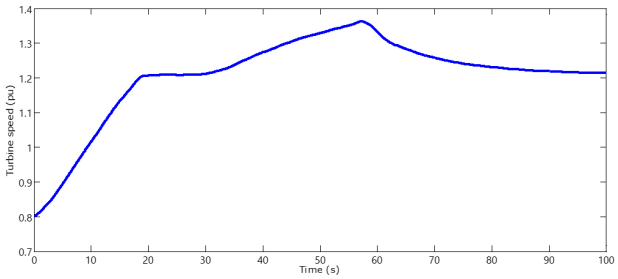

Fig. 7: *Variation of wind turbine speed under random wind variation within the range of 8 to 14 m/s.*

Case 2: Random wind speed in the range 8 to 24 m/s

In this case, since the wind speed crosses that of the rated, the pitch controller controls the speed of the wind turbine rotor and maintains it at 1.21 pu as shown in Fig. 8. The steady-state error and settling time are fairly good, as indicated in Table 7.

The results obtained for settling time and steady-state error vary depending on the wind speed in different runs. However, the variation in steady-state error is within the range of 10^{-4}.

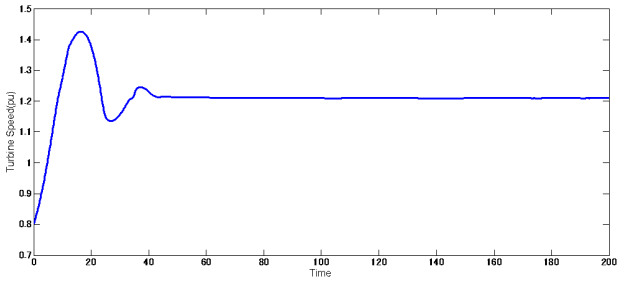

Fig. 8: *Variation in wind turbine speed under random wind variation within the range of 8 to 24 m/s.*

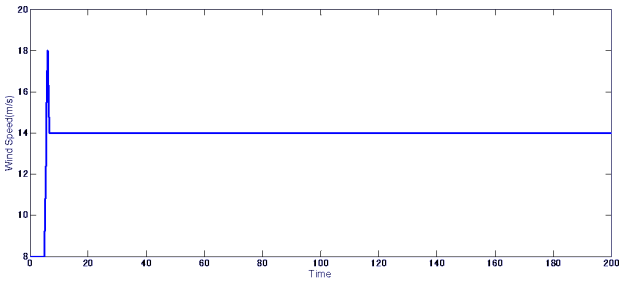

Fig. 9: *Wind gust from 8 to 18 m/s at 5 s.*

Case 3: Wind gust (8 to 18 m/s)

The wind gust is applied at 5 s from 8 to 18 m/s to study the effect of a sudden significant change in wind speed beyond the rated level and the optimization range. The gust finally settles at 14 m/s as shown in Fig. 9. The performance of the MVMO-SH, PSO, and GA tuned controllers are compared for wind gust. The variation in turbine speed response is depicted in Fig. 10. As revealed in Table 6 (rows 10–12), the MVMO-SH settling time (2% band) is less for a step change in wind speed with the ITAE criterion. When the wind speed takes the form of a wind gust, the corresponding settling time is still

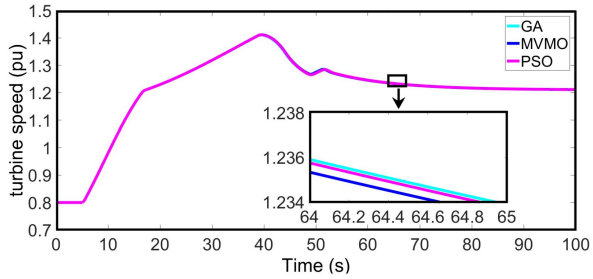

Fig. 10: *Comparison of the wind turbine speed response using different optimization techniques for wind gust.*

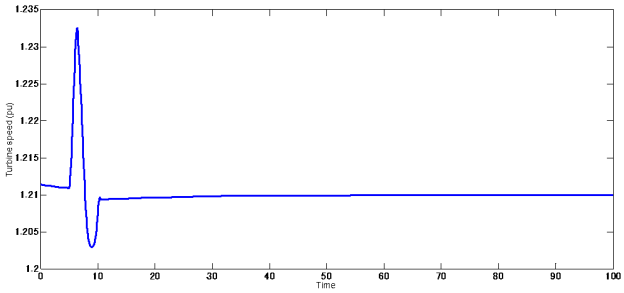

Fig. 11: *Variation in wind turbine speed with time under wind gust ranging from 14 to 20 m/s.*

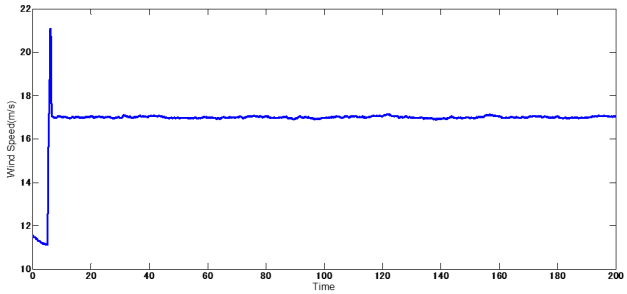

Fig. 12: *Wind speed with time under a wind gust of 8 to 18 m/s with a random wind variation.*

improved (changing to 64–65 s from an earlier range of 67–68 s).

Therefore, the MVMO-SH can claim a better performance than the other two techniques in settling the angular speed of the turbine. In view of the results presented in Table 6 and Fig. 10, although the MVMO-SH demonstrates a marginal improvement in performance, it is still better than the PSO and GA.

Case 4: Wind gust (14 to 20 m/s)

The speed range of a gust at 5 s is beyond the range of wind speed for which the controller is tuned, i.e., 14 m/s. The initial steady-state generated using a proportional controller with $K_p = 500$ gives a higher error of 0.0015, as can be observed from Fig. 11. With the optimized PI controller, the steady-state error is considerably reduced and the response converges. The peak overshoot with such an upsurge is also

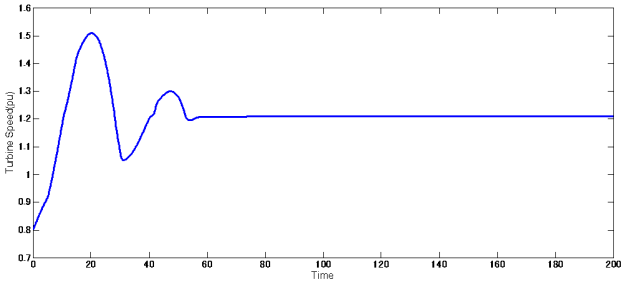

Fig. 13: *Variation in wind turbine speed under a wind gust range of 8 to 18 m/s with random wind variation.*

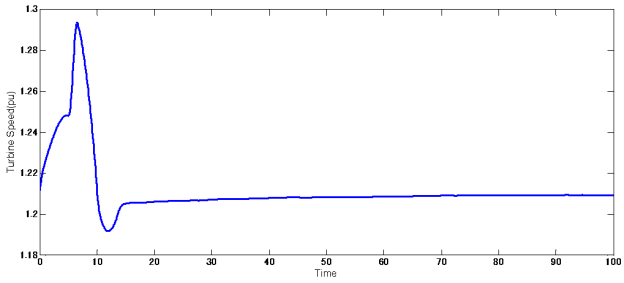

Fig. 14: *Variation in wind turbine speed under a wind gust range of 14 to 20 m/s and random wind variation.*

1.2325 pu which is within 2% band of the steady-state value.

Case 5: Wind gust (8 to 18 m/s) and a random wind speed change

With both wind randomness and the occurrence of a wind gust at 5 s, the performance is good, as presented in Table 7. The wind speed and turbine speed response can be observed in Figs. 12 and 13. The response is quantified in Table 7.

Case 6: Wind gust (14 to 20 m/s) and random wind speed change

In this case, the performance of the controller is tested for a highly unpredictable wind speed beyond the optimization range. The response of the wind turbine speed is shown in Fig. 14.

In steady-state, the response settles well and is quantified in Table 7.

Case 7: Wind gust (8 to 14 m/s) under a random line to ground fault

Since the wind farm is connected to the grid, a random change in wind speed with a gust occurring at 5 s at low wind speed is simulated followed by a line to ground (LG) fault on "A" phase of the grid at 10 s for nine cycles (0.15 s). This scenario is frequent during a storm. The developed controller could accomplish control as depicted in Table 7 while the wind turbine speed plot is shown in Fig. 15.

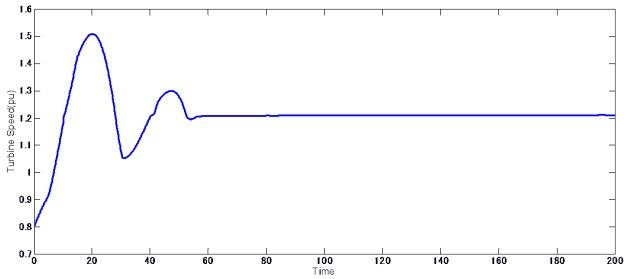

Fig. 15: *Wind turbine speed under a wind gust in the range of 8 to 14 m/s, random wind variation and an LG fault.*

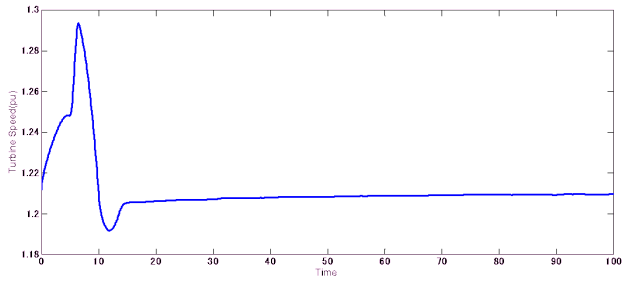

Fig. 16: *Wind turbine speed under high wind gust, random variation of wind, and LG fault.*

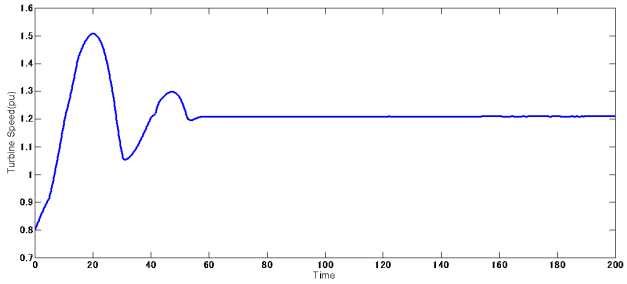

Fig. 17: *Wind turbine speed under low wind gust 8 to 14 m/s and LG fault at 5 s with random variation of wind.*

Case 8: Wind gust (14 to 20 m/s) and a random LG fault

With a gust occurring at 5 s at a high wind speed of 14 to 20 m/s under random wind conditions, a line to ground fault on the grid side for phase A was created at 10 s, and the PI controller performance checked. This situation enables the reliability of the power supply to be evaluated as it occurs under stormy weather conditions. The settling time was found to be significantly reduced, as demonstrated in Table 7. The response is shown in Fig. 16.

Case 9: Simultaneous wind gust (8 to 14 m/s) and LG fault in presence of random variation of wind speed

A case of wind speed random change with gust stirring at 5 s and at the same time a line to ground fault on A phase of the grid is modeled. The potential accomplishment of the controller design is shown in Table 7 while the plot of the wind turbine speed is depicted in Fig. 17.

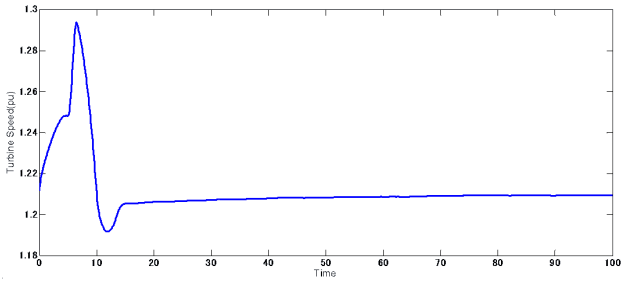

Fig. 18: *Wind turbine speed under simultaneous wind gust, LG fault with random variation of wind.*

Table 7: *Performance of optimal controller under transient conditions.*

Case study	Settling time (s)	Steady-state error
1	–	2.67×10^{-4}
2	33.15	-1.13×10^{-4}
3	64.52	1.3×10^{-7}
4	0*	-1.66×10^{-6}
5	51.88	2.92×10^{-5}
6	4.36	-4.71×10^{-4}
7	51.74	2.91×10^{-5}
8	4.36	-4.71×10^{-4}
9	51.64	3.17×10^{-5}
10	4.36	-4.71×10^{-4}

*Does not exceed 2% band

Case 10: Simultaneous wind gust (14 to 20 m/s) and LG fault in presence of random variation of wind speed

A case of wind gust and LG fault occurring simultaneously at 5 s in presence of randomness of wind is simulated. The result is tabulated, which shows no deterioration of performance as compared to occurrence of a disturbance at different instants of time. The response is shown in Fig. 18.

4. RESULTS AND DISCUSSION

In this study, an optimal PI controller was designed using a new optimization technique. When comparing the overall response of a PI controller optimized using a genetic algorithm (GA) as in [8] and apply it to a different system specification, this method gives a better steady-state error. Furthermore, the controller performance in the previous study was not analyzed for different disturbances. When comparing the response of the proportional (P) controller in [5], it gives a steady-state error without disturbance in the range of 10^{-4} whereas

here it is 10^{-7} which indicates good tracking. It can be observed from Table 6 that the MVMO-SH is faster than the other two optimization techniques, and gives a better steady-state error performance as well as settling time. Thus, the optimal controller designed using the MVMO-SH performs well under the stochastic nature of wind. From the observations in Table 6, it can be concluded that the designed controller using the MVMO-SH for one operating range of wind speed (8 to 14 m/s) performs well for different operating ranges of wind speed (8 to 18 m/s or 14 to 20 m/s). The steady-state error is in the range of 10^{-4} in the most critical condition during the occurrence of simultaneous disturbances. However, the peak overshoot is slightly high in some cases as can be observed from the figures for wind turbine speed response ω_r with time. The settling time is dependent on the occurrence of disturbance.

Bekakra *et al.* [12] optimized the PI controller gains using PSO in a similar approach to that applied here, but with the maximum power point tracking (MPPT) control being confined to the cut-in at below the rated wind speed level (region II) active and reactive power control. An improvement in the performance of the PSO optimized PI controller compared to the manually tuned PI controller has been demonstrated. Although the focus is different, the findings reflect the better performance of the optimization algorithm compared to the conventional Ziegler Nichols (ZN) tuning method. This supports our results.

In contrast to the previous study [12], the proposed PI controller with the optimization algorithms can be said to be simple and produces a better performance by reducing mechanical fatigue under practical actuator and wind speed conditions in region III. However, due to the varying system configurations, the gains are different and cannot be compared graphically.

5. CONCLUSION

In this study, a proportional-integral (PI) controller in a pitch angle control loop has been tuned through hybrid mean-variance mapping optimization (MVMO-SH) technique to provide optimal PI control gains K_p and K_i. Ziegler Nichols (ZN) tuning has also been implemented for obtaining K_p and K_i gains, although a detailed comparison is omitted here for brevity. A comparative study has been undertaken using different fitness evaluation criteria. The performance of the designed optimal controller is presented here in the time domain. The performance of the optimal PI controller designed for one operating region was analyzed under different operating conditions for wind gust, random wind variation, and disturbance from the grid side using a line to ground fault. In all the different operating conditions, the controller design performed satisfactorily, with fewer steady-state errors, whereas the controller designed using the ZN tuning rule failed under wind gust and random wind speed conditions in cases 8–10.

REFERENCES

[1] Y. Mishra, S. Mishra, and F. Li, "Coordinated Tuning of DFIG-Based Wind Turbines and Batteries Using Bacteria Foraging Technique for Maintaining Constant Grid Power Output," *IEEE Systems Journal*, vol. 6, no. 1, pp. 16–26, Mar. 2012.

[2] S. Heier, *Grid Integration of Wind Energy Conversion Systems*. New York, USA: John Wiley & Sons, 1998.

[3] K. J. Åström and T. Hägglund, "The future of PID control," *Control Engineering Practice*, vol. 9, no. 11, pp. 1163–1175, Nov. 2001.

[4] J. Jonkman, S. Butterfield, W. Musial, and G. Scott, "Definition of a 5-MW Reference Wind Turbine for Offshore System Development," NREL, Golden, CO, USA, Tech. Rep. NREL/TP-500-38060, Feb. 2009.

[5] J. G. Slootweg, S. W. H. de Haan, H. Polinder, and W. L. Kling, "General model for representing variable speed wind turbines in power system dynamics simulations," *IEEE Transactions on Power Systems*, vol. 18, no. 1, pp. 144–151, Feb. 2003.

[6] W. Qiao, G. K. Venayagamoorthy, and R. G. Harley, "Design of Optimal PI Controllers for Doubly Fed Induction Generators Driven by Wind Turbines Using Particle Swarm Optimization," in *The 2006 IEEE International Joint Conference on Neural Network Proceedings*, 2006, pp. 1982–1987.

[7] I. Poultangari, 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, no. 5, pp. 641–648, Sep. 2012.

[8] H. F. Boroujeni, B. K. Boroujeni, A. Memaripour, and M. Eghtedari, "Unified Control of Doubly Feed Induction Generator in Wind Farm by using Genetic Algorithms," *Research Journal of Applied Sciences, Engineering and Technology*, vol. 4, no. 8, pp. 859–862, 2012.

[9] F. O. Echavarri, E. Zulueta, J. A. R. Hernanz, J. M. L. Guede, and J. M. L. Lesaka , "Particle Swarm Optimization Based Tuning for a Small Wind Turbine Pitch Control," *International Journal on Technical and Physical Problems of Engineering*, vol. 4, no. 4, pp. 164–170, Dec. 2012.

[10] S. A. Raza and A. H. M. A. Rahim, "Smart Pitch Control Strategy of Wind Generation System Using Differential Evolution and Neural Network," in *2012 International Conference on Power and Energy Systems*, 2012, pp. 202–207.

[11] A. S. Oshaba, and E. S. Ali, "Speed Control of Induction Motor Fed from Wind Turbine via Particle Swarm Optimization Based PI Controller," *Research Journal of Applied Sciences, Engineering and Technology*, vol. 5, no. 18, pp. 4594–4606, 2013.

[12] Y. Bekakra and D. B. Attous, "Optimal tuning of PI controller using PSO optimization for indirect power control for DFIG based wind turbine with MPPT," *International Journal of System Assurance Engineering and Management*, vol. 5, no. 3, pp. 219–229, 2014.

[13] F. A. R. Abbas and M. A. Abdulsada, "Simulation of Wind-Turbine Speed Control by MATLAB," *International Journal of Computer and Electrical Engineering*, vol. 2, no. 5, pp. 912–915, Oct. 2010.

[14] A. S. Yilmaz and Z. Ozer, "Pitch angle control in wind turbines above the rated wind speed by multi-layer perceptron and radial basis function neural networks," *Expert Systems with Applications*, vol. 36, no. 6, pp. 9767–9775, Aug. 2009.

[15] Y. El-Tous, "Pitch Angle Control of Variable Speed Wind Turbine," *American Journal of Engineering and Applied Sciences*, vol. 1, no. 2, pp. 118–120, 2008.

[16] W. Qiao, J. Liang, G. K. Venayagamoorthy, and R. Harley, "Computational intelligence for control of wind turbine generators," in *2011 IEEE Power and Energy Society General Meeting*, 2011.

[17] C. Jauch, S. M. Islam, P. Sørensen, and B. B. Jensen, "Design of a wind turbine pitch angle controller for power system stabilisation," *Renewable Energy*, vol. 32, no. 14, pp. 2334–2349, Nov. 2007.

[18] J. Wang, N. Tse, and Z. Gao, "Synthesis on PI-based pitch controller of large wind turbines generator," *Energy Conversion and Management*, vol. 52, no. 2, pp. 1288–1294, Feb. 2011.

[19] A. Musyafa, A. Harika, I. M. Y. Negara, and I. Robandi, "Pitch Angle Control of Variable Low Rated Speed Wind Turbine Using Fuzzy Logic Controller," *International Journal of Engineering & Technology*, vol. 10, no. 5, pp. 22–25, 2010.

[20] Y. Qi and Q. Meng, "The Application of Fuzzy PID Control in Pitch Wind Turbine," *Energy Procedia*, vol. 16, Part C, pp. 1635–1641, 2012.

[21] J. L. Rueda and I. Erlich, "Hybrid Mean-Variance Mapping Optimization for solving the IEEE-CEC 2013 competition problems," in *2013 IEEE Congress on Evolutionary Computation*, Cancun, Mexico, 2013, pp. 1664–1671.

[22] J. L. Rueda and I. Erlich, "Evaluation of the mean-variance mapping optimization for solving multimodal problems," in *2013 IEEE Symposium on Swarm Intelligence (SIS)*, Singapore, 2013, pp. 7–14.

[23] F. Gonzalez-Longatt, J. L. Rueda, I. Erlich, D. Bogdanov, and W. Villa, "Identification of Gaussian Mixture Model using Mean Variance Mapping Optimization: Venezuelan Case," in *2012 3rd IEEE PES Innovative Smart Grid Technologies Europe (ISGT Europe)*, Berlin, Germany, Oct. 2012, pp. 1–6.

[24] J. L. Rueda, J. C. Cepeda, and I. Erlich, "Estimation of location and coordinated tuning of PSS based on mean-variance mapping optimization," in *2012 IEEE Power and Energy Society General Meeting*, 2012.

[25] R. M. Pringles and J. L. Rueda, "Optimal transmission expansion planning using Mean-Variance Mapping Optimization," in *2012 Sixth IEEE/PES Transmission and Distribution: Latin America Conference and Exposition (T&D-LA)*, 2012.

[26] R. Gagnon, B. Saulnier, and A. Forcione, "Wind Farm (DFIG Phasor Model)." Mathworks.com. `https://www.mathworks.com/help/physmod/sps/ug/wind-farm-dfig-phasor-model.html`

[27] I. J. Nagrath and M. Gopal, *A Textbook of Control Systems Engineering*. New Delhi, India: New Age International, 2010.

[28] B. Mohanty, S. Panda, and P. K. Hota, "Differential evolution algorithm based automatic generation control for interconnected power systems with non-linearity," *Alexandria Engineering Journal*, vol. 53, no. 3, pp. 537–552, Sep. 2014.

Sasmita Behera received her B.E. in EE from University College of Engineering (UCE), Burla, India in 2000, M.E. in Power System Engg in 2003 from UCE, Burla under Biju Patnaik University of Technology, Rourkela, India, and her Doctorate in the area of Wind Energy from Veer Surendra Sai University of Technology, Burla, India in 2017. She is currently an Assistant Professor in the Department of Electrical & Electronics Engineering, Veer Surendra Sai University of Technology, Burla, India She has nine years' experience in the power distribution company, WESCO Odisha, India where she worked in the capacity of Executive Engineer and more than 10 years of teaching and research experience. Her areas of interest are renewable energy control and applications. She has more than 12 publications in international journals and has attended and presented papers in 30 national and international conferences. She has three book chapters to her credit. She is an active member of many professional bodies. She is a reviewer of various International Journals. She is a Life Member ISTE, Member of Institute of Engineers and Life member of Odisha Bigyan Academy.

Subham Sahoo received his B.Tech. in Electrical and Electronics Engineering from Veer Surendra Sai University of Technology, Burla, India in 2014 and Ph.D. degree in Electrical Engineering at Indian Institute of Technology, Delhi, New Delhi, India in 2018. He has worked as a Visiting Student with the Department of Electrical and Electronics Engineering in Cardiff University, UK in 2017. Prior to completion of his Ph.D., he worked as a postdoctoral researcher in the Department of Electrical and Computer Engineering in National University of Singapore during 2018-19 and in Aalborg University (AAU), Denmark during 2019-2020. He is currently an Assistant Professor in the Department of Energy Technology, AAU, Denmark.

He is a recipient of the Indian National Academy of Engineering (INAE) Innovative Students Project Award for his Ph.D. thesis across all the institutes in India for the year 2019. He was also a distinguished reviewer for IEEE Transactions on Smart Grid in the year 2020. He currently serves as a secretary of IEEE Young Professionals Affinity Group, Denmark and Joint IAS/IES/PELS in Denmark section. He is an active member of IEEE PELS technical committee (TC #10) on automated design methodologies and cybersecure solutions for power electronics.

His research interests are control, optimization, and stability of power electronic dominated grids, renewable energy integration, physics-informed AI tools for cyber-physical power electronic systems.