Fault-tolerant optimal pitch control of wind turbines using dynamic weighted parallel firefly algorithm

Mousavi , Yashar; Bevan, Geraint; Kucukdemiral, Ibrahim Beklan

Published in:
ISA Transactions

DOI:
10.1016/j.isatra.2021.10.019

Publication date:
2022

Document Version
Author accepted manuscript

Fault-Tolerant Optimal Pitch Control of Wind Turbines Using Dynamic Weighted Parallel Firefly Algorithm

Yashar Mousavi[1], Geraint Bevan[1], and Ibrahim Beklan Küçükdemiral[1]

[1] *Department of Applied Science, School of Computing, Engineering and Built Environment, Glasgow Caledonian University, Glasgow G4 0BA, UK*

Abstract

With steadily increasing interest in utilizing wind turbine (WT) systems as primary electrical energy generators, fault-tolerance has been considered decisive to enhance their efficiency and reliability. In this work, an optimal fault-tolerant pitch control (FTPC) strategy is addressed to adjust the pitch angle of WT blades in the presence of sensor, actuator, and system faults. The proposed scheme incorporates a fractional-calculus based extended memory (EM) of pitch angles along with a fractional-order proportional-integral-derivative (FOPID) controller to enhance the performance of the WT. A dynamic weighted parallel firefly algorithm (DWPFA) is also proposed to tune the controller parameters. The efficiency of the proposed algorithm is evaluated on the test functions adopted from 2017 IEEE congress on evolutionary computation (CEC2017). The merits of the proposed fault-tolerant approach are tested on a 4.8-MW WT benchmark model and compared to conventional PI and optimal FOPID approaches. Corresponding comparative simulation results validate the effectiveness and fault-tolerant capability of the proposed control paradigm, where it is observed that the proposed control scheme tends to be more consistent in the power generated at a given wind speed.

Keywords: Fault-tolerant control, Wind turbine, Pitch control, Optimization, Firefly algorithm, Fractional calculus.

1 Introduction

Renewable energy, especially wind turbine (WT) systems, have gained considerable attention during the past decade due to the energy shortage and environmental issues [36, 32, 37]. Since WTs have contributed a considerable portion of the world's power production, demands on the development of reliable control

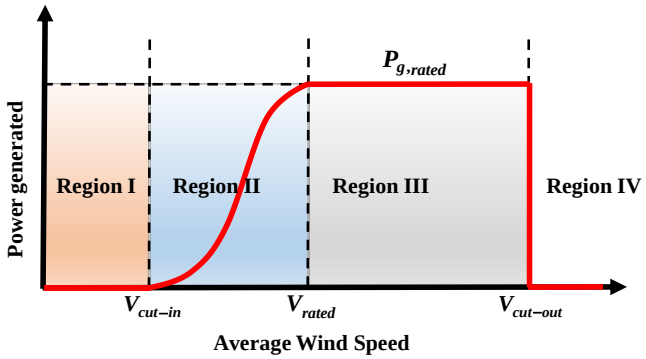

Figure 1: Wind turbine operational regions.

approaches that guarantee the power generation and reduce the operational and maintenance costs have increased substantially.

Variable-speed WT systems operate in four regions, which are related to wind speed [37]. Figure 1 demonstrates the operating regions, where V_{cut-in} and $V_{cut-out}$ stand for the wind speed at which the blades start and stop rotating, respectively. Also, V_{rated} denotes the wind speed that the blades reach the maximum speed, where the maximum power ($P_{g,rated}$) is generated. In regions I and IV, the wind speed is respectively, too low and too high, so that the WT does not operate. In region II (*i.e.* the partial-load region), the wind speed is lower than the rated wind speed. At the rated wind speed, the turbine is capable of generating electricity at its maximum capacity. Therefore, in this region, a generator torque controller is used to maximize the power capture. On the other hand, in region III (*i.e.* the full-load region), the wind speed exceeds the rated value. In this region, pitch actuation is critical for limiting the power capture in high wind speed situations. Hence, pitch control (PC) strategies are used to control the pitch angle and keep the WT operating at its rated power. Numerous studies in the literature have dealt with the PC problem in order to limit the aerodynamic power captured by the WT. For instance, in [50], the authors investigated the PC based on a nonlinear proportional-integral (PI) controller together with a state and perturbation observer. In [64], a PC scheme consisting of the conventional PI and two resonant compensators was developed. Authors in [56] proposed an advanced PC strategy based on fuzzy logic control (FLC), while in [57, 7] fuzzy proportional-integral-derivative (PID) and fractional-order fuzzy PID controllers were investigated to improve PC performance.

The aforementioned studies consider the ideal situation, where the variations in the actuator dynamics and sensor faults are assumed to be negligible. However, in real operations, WTs are prone to different sets of sensor, actuator, and system faults, which degrade the WT stability and power production performance and impose maintenance costs. Accordingly, various fault-tolerant

pitch control (FTPC) approaches have been investigated to compensate the fault effects in WT systems and achieve a robust system performance. To this end, in [12], a FTPC scheme is investigated based on an adaptive PI controller augmented with a fault detection strategy, while authors in [26] developed an adaptive PID. Although compared to other classical approaches, the controller demonstrated more acceptable performance in terms of handling non-linear dynamics, further improvements are required to mitigate the fault effects when unexpected actuator faults and wind speed fluctuations happen. Authors in [30] incorporated a conventional PI along with a sliding mode observer to compensate for faults. According to the authors, the proposed control strategy is capable of recovering the nominal pitch actuation; however, only the case of low pressure actuator fault occurrence is considered in the study, where the controller's performance in more harsh situations is yet to be investigated. A Kalman filter was used to assess the blade pitch angle of the WT, together with a PI to deal with the FTPC problem in [20]. Authors in [11] investigated the FTPC based on the incorporation of FLC and PI in the presence of sensor faults. An important issue in designing and evaluating the performance of controllers for WT systems is the fact that, WTs are not exposed to only one type of fault at a time, and concurrent faults occurrences are inevitable. Thus, it is important to evaluate the controller's proficiency in the presence of different fault scenarios, which is not considered in the aforementioned studies.

During the past decade, the concept and applications of fractional calculus have attracted growing interests of scholars in various engineering fields [42, 9, 40]. Fractional order (FO) derivatives induce an infinite series, presenting a long memory of the past [39], whereas integer-order derivatives are local operators that imply a finite number of terms. Since wind energy and direction have chaotic behavior, the pitch actuation system needs to provide an immediate precise response, which in practice, leads to some slight errors. Thus, it is desirable to preserve all the past effective pitch angles, representing the memory of the pitch system characteristics. Therefore, WT systems are quite suitable processes to be used with FO controllers. Similar to PID controllers, fractional-order PID (FOPID) controllers have been extensively implemented in many applications [5, 49, 44]. FOPID controllers not only inherit the advantages of conventional PIDs such as simple structure and strong robustness but also expand the control range by adding more flexibility to the control system [9, 5]; however, the existence of two more tunable parameters has made the design problem more complicated. A variety of tuning rules and design methods have been investigated in the literature [9, 4], while most of them suffer from the unavailability of the exact dynamic model in Laplace domain representation, especially for complex nonlinear systems [5]. As an alternative solution, evolutionary algorithms (EAs) have been playing a crucial role in determining FOPID parameters [38, 31]. A gain-scheduling FOPID was proposed in [6] to alleviate mechanical loads and improve pitch angle adjustment performance. The controller's parameters were chosen using radial basis function neural network (RBFNN), where a chaotic differential evolution algorithm was used to create the database to train the RBFNN. The authors have also used the same

method to design a PID controller, where as reported, the FOPID has outperformed the PID and the conventional gain-scheduling baseline PI controller in terms of load mitigation. In another study [7], an optimal fractional-order fuzzy PID controller was developed for pitch control of WTs. The controller's parameters were optimized taking advantage of chaotic optimization algorithms. As reported by the authors, compared with the gain-scheduling PID, the proposed controller demonstrated better performance with fewer fatigue damages in different wind speeds; however, the system was assumed to be fault-free. In this paper, a fractional-calculus based extended memory of pitch angles is augmented with the controller to enhance its performance for adjusting the desired pitch angle of WT blades and improving the power generation of the WT in the presence of faults.

Metaheuristic optimization algorithms have been extensively employed to optimally tune the controllers' parameters for WT control systems. Authors in [29] investigated the application of whale optimization algorithm for optimal design of PI controller in a WT pitch control system, aiming at alleviating the low-frequency torsional oscillations. A modified root tree optimization algorithm was proposed in [14] to adjust the PI controller parameters to minimize the chattering phenomenon in the active and reactive powers of WT. Authors in [61] developed two gravitational search algorithm (GSA) -optimized fractional-order sliding mode controllers (SMC) for PMSG to enhance its output power quality. One was proposed to control the rotor-side dq axis currents in the machine side converter, and the other one was proposed to regulate the output voltage of the GSC. According to the comparison results provided, the proposed control strategy demonstrated more tracking precision and stronger robustness against parametric disturbances over conventional SMC and PI control. An optimal fuzzy SMC was proposed in [17] for maximum power extraction of WT with zero stator reactive power regulation. Accordingly, a combination of particle swarm optimization (PSO) and GSA was proposed to tune the control parameters optimally. Firefly algorithm (FA), as one of the recently introduced EAs [63], has been effectively solved many optimization problems in recent years [35, 45, 23]. Authors in [53] developed a distributed parallel FA for parameter tuning of a variable pitch WT, where according to the authors, the proposed control scheme reduced the power fluctuation and improved the safety and reliability of WT. FA has certain superiorities over some of the most used EAs. To name a few, a) FA is able to tune its scaling parameter and hence adapt to problem landscape, b) FA can be counted as a generalization of PSO, differential evolution (DE), and simulated annealing (SA) [63], which takes all the three algorithms' advantages, c) unlike PSO, FA does not use velocities, and thus, can avoid the drawbacks associated with the velocity initialization [23], d) since the fireflies aggregate more closely around each optimum, it has shown superior performance over genetic algorithm (GA) that jumps around randomly, and e) since local attraction is more substantial than long-distance attraction, FA can automatically subdivide its population into subgroups, which makes it a suitable method to efficiently tackle nonlinear and multimodal problems [63]. However, the success of the search procedure in FA depends on a suitable trade-

off between global search (exploration) and local search (exploitation) abilities, which corresponds to the attractiveness formulation and variation of light intensity. Both factors allow significant scope for the algorithm's improvements. Thus, investigations have been performed to enhance its performance taking advantage of other search methodologies in order to achieve even better performance [59, 48, 3]. In this study, an enhanced FA is developed that explores the search space with a well-connected weighted parallel strategy that enriches the population diversity and increases the information exchange between the fireflies. The proposed strategy not only expedites the convergence speed of FA, but also reduces the possibility of getting trapped in local optima. A dynamic switching coefficient is also implemented that lets the algorithm perform more accurate exploitations. The switching coefficient makes the algorithm work more precisely and finds more reasonable solutions.

Even though various advanced control strategies have been developed for WTs such as sliding mode control, model predictive control, FLC-based control, etc. [27, 62, 41], PI/PID method is still the preferred approach in real-world applications with some improvements [30, 2] due to its simplicity. In this regard, many researchers have utilized a simple PI controller through the pitch angle regulation process in region III [26, 30, 20], and some studies have focused on applying rotor speed limitations [54, 33]. However, due to the existence of only two tuning parameters in PI controllers, the strategy of utilizing a simple PI does not guarantee the minimum steady-state error, especially when faults occur in the system [12, 11]. Besides, although PI/PID control has attracted a wide range of attention in WT control systems, a significant limitation still remains; how to determine the controller's gains. Accordingly, despite the existence of various methods for tuning PID gains [2, 15, 26], there is no specific way to determine such gains for WT control, as they need be chosen by the designer which is neither a straightforward task nor optimal. This has motivated our attempts to construct an optimal controller for WT control. Hence, to effectively maintain constant power generation, an optimized FOPID controller augmented with extended memory of the pitch angles is developed to regulate the pitch angle and prevent the WT from over-speeding.

In this paper, several performance evaluations of the proposed dynamic weighted parallel FA (DWPFA) in comparison to other conventional and modified EAs are investigated through solving well-defined 2017 IEEE congress on evolutionary computation (CEC2017) mathematical benchmark functions [8]. Non-parametric Friedman and Friedman Aligned statistical tests are also provided to statistically analyze the quality of the solution [24]. The proposed memory extension of pitch angles is incorporated in the FOPID controller (called EM-FOPID) to generate the desired WT pitch angle reference in the presence of sensor, actuator, and system faults, where the controller parameters are tuned using the proposed DWPFA algorithm. This study contributes the literature as follows:

1. Using the concept of fractional calculus, a fault-tolerant pitch control strategy with extended memory of pitch angles (EM-FOPID) is devel-

oped that improves the power generation of the WT, where the controller parameters are tuned using the proposed DWPFA.

2. A modified FA (DWPFA) is proposed that increases the convergence speed of the conventional FA, reduces the possibility of getting trapped in local optima, and increases the exploitation accuracy.

3. Comparative simulations are provided that reveal the remarkable performance of proposed optimal EM-FOPID with respect to optimal FOPID and conventional PI.

The paper is organized as follows. The problem statement, including the WT modeling, control objective, and fault scenarios, are described in Section 2. The proposed DWPFA algorithm is explored in Section 3. Section 4 presents the proposed EM-FOPID control strategy. The performance of DWPFA is evaluated, and the EM-FOPID design is verified in Section 5. Finally, Section 6 concludes the paper.

2 Problem Statement

In this section, first, the model of a three-bladed variable speed WT is investigated, and then, the control objective of the study is described. The last subsection is devoted to introducing the different sensor, actuator, and system fault scenarios with various levels of severity, which are considered to occur to the WT.

2.1 Wind Turbine Modeling

In this work, a 4.8 MW three-bladed variable speed horizontal axis wind turbine (HAWT) is considered [47]. The system consists of three main units: the generator-converter, drive-train, and the blade and pitch model. The aerodynamic and pitch system models are combined to form the blade and pitch model, where the former denotes the transformation of wind power to rotational energy, and the latter rotates the blades around their longitude axis. The drive-train provides the required rotational speed of the generator. To convert the mechanical wind energy to electrical energy, the coupled converter-generator system is utilized. Owing to the controlled pitching of the blades changing the aerodynamic efficiency of the WT, aerodynamic wind energy is transformed into effective mechanical energy. Thus, the captured power significantly depends on the available wind energy and the geometry of the blade aerofoils and their pitch, which thus affect the responding capability of the machine to wind fluctuations. The extracted aerodynamic power can be represented as follows [28].

$$P_a(t) = \frac{1}{2}\rho\pi R^2 \upsilon_w(t)^3 \mathcal{C}_P(\lambda(t), \beta(t)), \tag{1}$$

where R represents the rotor radius in [m], ρ denotes the air density in [kg/m^3], and $\upsilon_w(t)$ is the effective modeled wind speed at the rotor plane. $\mathcal{C}_P$

stands for the power coefficient, $\lambda = R\omega_r/\upsilon_w$ stands for the tip-speed ratio, where β denotes the pitch angle and ω_r represents the rotational speed of the rotor in [rad/s]. Finally, t is time in [s].

Remark 1. *The dimensionless parameter $\mathcal{C}_P$ is an experimental coefficient, which is generally adopted from a look-up table in terms of (λ,β). However, taking advantage of some curve-fitting techniques [1, 21, 43], it can be approximated as follows [13].*

$$\mathcal{C}_P(\lambda,\beta) = \mathcal{C}_1\left(\frac{\mathcal{C}_2}{\Lambda} - \mathcal{C}_3\beta - \mathcal{C}_4\right)e^{-\mathcal{C}_5/\Lambda} + \mathcal{C}_6\lambda, \tag{2}$$

where $\mathcal{C}_1 = 0.5176$, $\mathcal{C}_2 = 116$, $\mathcal{C}_3 = 0.4$, $\mathcal{C}_4 = 5$, $\mathcal{C}_5 = 21$, $\mathcal{C}_6 = 0.0068$, *and*

$$\Lambda = \left[\frac{1}{\lambda + 0.08\beta} - \frac{0.035}{\beta^3 + 1}\right]^{-1}. \tag{3}$$

To maximize the power captured from the wind, the pitch angle and tip-speed ratio should be controlled to optimize the power coefficient $\mathcal{C}_P$ *i.e.* $\mathcal{C}_{P,\max} \triangleq \mathcal{C}_P(\beta_{opt}, \lambda_{opt})$, where $\mathcal{C}_{P,\max}$ is the maximum power coefficient. Thus, considering $\omega_r = \eta_g\omega_g$, where η_g represents the generator's efficiency, and ω_g and ω_r define the generator and rotor rotational speed, respectively, the optimal rotor speed $\omega_{r,opt}$ can be achieved as $\omega_{r,opt} = \lambda_{opt}\upsilon_w/R$.

Figure 2 demonstrates the variations of power coefficient $\mathcal{C}_P$ for different values of λ and β. According to Fig. 2 it can be observed that the WT has a maximum efficiency of approximately $\mathcal{C}_{P,\max} = 0.4797$ for a tip speed ratio $\lambda_{opt} = 8.2$.

On the other hand, the aerodynamic torque produced by the WT can be expressed by $T_a = P_a/\omega_r$. Besides, the pitch actuator model consists of a hydraulic and a mechanical machinery, and can be expressed as the following second-order system [47]:

$$\frac{\beta(s)}{\beta_r(s)} = \frac{\omega_n^2}{s^2 + 2\xi\omega_n s + \omega_n^2}, \tag{4}$$

where β_r denotes the command signal for the pitch angle being produced by the WT controller, β stands for the actual pitch angle produced by the actuator, ξ denotes the damping factor, and ω_n represents the natural frequency in [rad/s].

Remark 2. *In a pitch actuator, the pitch actuator constraints play a critical role. In this regard, the pitch rate constraints are considered between $-8^{\circ/s}$ and $8^{\circ/s}$, while the operational range of the pitch angle is considered as $-3^\circ \leq \beta \leq 90^\circ$ [10].*

The mechanical part of WT, namely the drive-train, is a rather complex system which consists of a gearbox, low-speed shaft, and high-speed shaft that converts the low-speed torque of the rotor-side shaft to a high-speed torque of the generator-side shaft. Since the WT is coupled to the generator through a

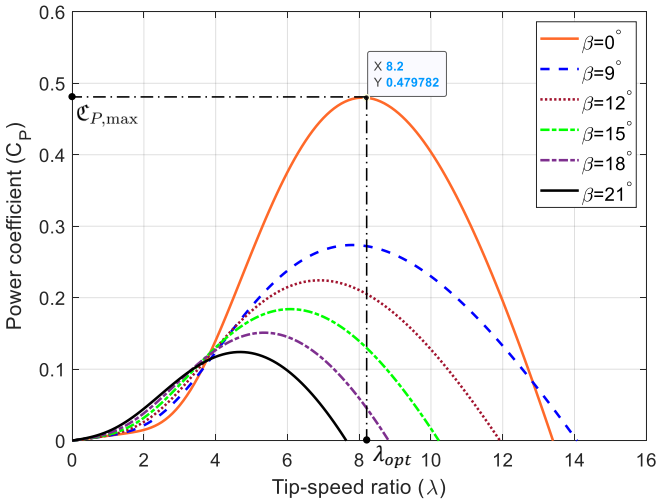

Figure 2: The power coefficient curve based on the optimal tip-speed ratio for different pitch angles.

gearbox, the generator torque T_g can regulate the rotor speed. The rotor inertia J_r [kg m^2] is driven at speed ω_r by the aerodynamic torque T_a [N m], and the generator inertia J_g [kg m^2] is driven by high-speed torque of the generator-side shaft at speed ω_g and is braked by the generator torque T_g. This study considers the following dynamic model of a two-mass drive-train model [54]:

$$\begin{bmatrix} \dot{\omega}_r \\ \dot{\omega}_g \\ \dot{\theta}_\Delta \end{bmatrix} = \begin{bmatrix} \vartheta_{11} & \vartheta_{12} & \vartheta_{13} \\ \vartheta_{21} & \vartheta_{22} & \vartheta_{23} \\ \vartheta_{31} & \vartheta_{32} & \vartheta_{33} \end{bmatrix} \begin{bmatrix} \omega_r \\ \omega_g \\ \theta_\Delta \end{bmatrix} + \begin{bmatrix} \psi_1 \\ 0 \\ 0 \end{bmatrix} T_a + \begin{bmatrix} 0 \\ \psi_2 \\ 0 \end{bmatrix} T_g, \quad (5)$$

where $\vartheta_{11} = -D_{dt} + B_r/J_r$, $\vartheta_{12} = D_{dt}/J_r N_g$, $\vartheta_{13} = -K_{dt}/J_r$, $\vartheta_{21} = \eta_{dt} D_{dt}/J_g N_g$, $\vartheta_{22} = -\eta_{dt} D_{dt}/J_g N_g^2 - B_g/J_g$, $\vartheta_{23} = \eta_{dt} K_{dt}/J_g N_g$, $\vartheta_{31} = 1$, $\vartheta_{32} = -1/N_g$, $\vartheta_{33} = 0$, $\psi_1 = 1/J_r$, $\psi_2 = -1/J_g$. K_{dt} [Nm/rad] and D_{dt} [Nms/rad] denote the low-speed shaft stiffness and damping coefficient, respectively. B_g stands for the viscous friction of the high-speed shaft in [Nms/rad], N_g denotes the gearbox ratio, and η_{dt} and θ_Δ represent the drive-train efficiency and torsion angle, respectively.

The generator is responsible for converting the shaft kinetic energy into electrical energy. Since compared to the WT dynamics, the dynamics of the electrical system are noticeably faster, the following first-order model with fast dynamics can be used to model the generator and converter dynamics [47].

$$\frac{T_g(s)}{T_{g,ref}(s)} = \frac{\alpha_{gc}}{s + \alpha_{gc}}, \quad (6)$$

where $T_{g,ref}$ stands for the torque reference to the generator, and $\alpha_{gc} = 1/\tau_{gc}$ denotes the generator and converter unit coefficient, with τ_{gc} representing the

time constant. The generated power by the generator can be achieved by $P_g = \eta_g T_g \omega_g$.

2.2 Control Objectives

The baseline control system in WTs consists of two individual control sections to regulate the generator torque and the pitch angle of the blades, where the overall performance of the WT directly depends on the performance of both controllers.

The wind energy is not always constant, and it holds different flow profiles (laminar, turbulent, etc.), and gusts, which can result in deceleration of the rotor speed to a critical speed, which brings instability and damage to the WT. Besides, as the power generated depends on the generator torque, it is evident from (6) that any change (due to faults or uncertainties) in the generator would directly affect the rated power. Thus, generator faults can impose severe problems in tracking the maximum power point and rated power in regions II and III, respectively. In this regard, in region II, the power reference P_{ref} tracking is switched to maximum power point tracking (MPPT) to stabilize the WT while maximizing the power capture [54]. The switch works with respect to the wind speed, as in Fig. 3, the dotted arrow from the Wind Profile block to the switch block shows its dependency on wind speed. Accordingly, reference torque to the converter can be represented as $T_{g,ref} = K_{opt}\omega_r^2$, where $K_{opt} = 1/2(\rho\pi R^5 \mathcal{C}_{P,\max}/\lambda_{opt}^3)$. Figure 3 depicts the block diagram of the proposed WT control scheme, which comprises three main blocks: (a) the WT model which is prone to actuator, system, and sensor faults, (b) the proposed DWPFA-based pitch control block, and (c) the generator torque assignment block, which includes the power reference tracking (P_{ref}/ω_r) and MPPT ($K_{opt}\omega_r^2$).

2.3 Fault Scenarios

Faults occurring in a WT can affect the system characteristics or lead to inoperable conditions. Wind turbine faults may be classified in terms of those that are highly serious, where the WT needs to be shut-down in order to prevent irreparable damage, and those faults that can be accommodated by suitable controllers, leading the WT to stay operational with some possible performance detriment. The faults modeled in this work include sensor (F1-F4), actuator (F5,F6), and system (F7) faults with various levels of severity, as summarized in Table 1; where each one can cause performance degradation or slight damage to the WT. The development time for F1-F5 is considered medium, while it is slow and very slow for F6 and F7, respectively.

Sensor faults mainly originate from mechanical or electrical faults in the sensors, due to drift, noise, and external factors such as lightning, heavy rain, moisture, storms, and corrosion; and also misalignment of one or more blades at the installation step or blade imbalancement during operation [25]. Considering the fact that the pitch position measurements act as a reference for the internal pitch system controller, sensor faults can negatively affect the pitch positions if

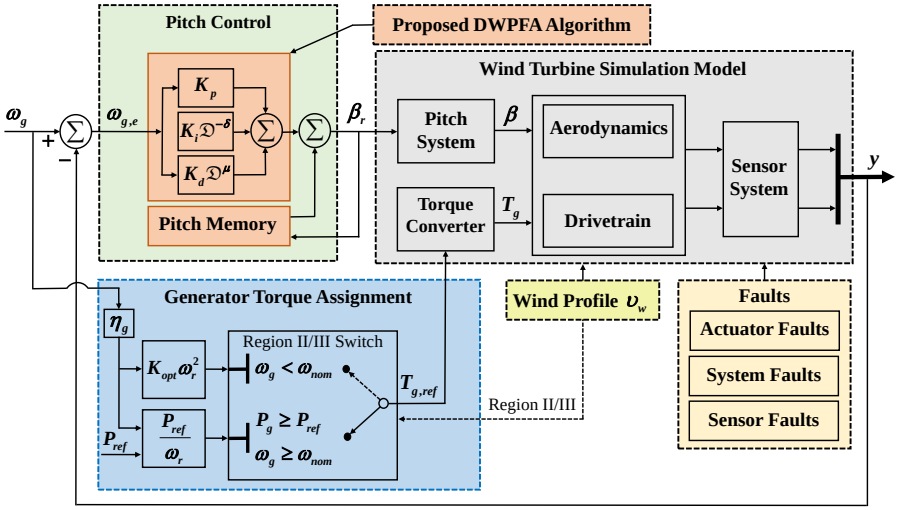

Figure 3: Proposed control scheme diagram for a wind turbine system.

Table 1: Sensor, actuator, and system fault scenarios considered.

Fault	Faulted (Occurrence time (s))	Severity
F1	$\beta_{1,m1} = 5°$ (2000-2100), $\beta_{1,m1} = 5°$ (2700-2900)	low
F2	$\beta_{2,m2} = 1.2°$ (2400-2500)	low
F3	$\beta_{3,m2} = 5°$ (2600-2700)	low
F4	$\omega_{r,m2} = 1.1$ (3805-4400), $\omega_{g,m1} = 0.9$ (3805-4400)	low
F5	Hydraulic pressure drop (2900-3000)	high
F6	Air content increment in the oil (3500-3600)	medium
F7	Friction changes in the drive-train (4100-4300)	medium

the control system fails to handle them properly, which leads to performance degradation of WT [20]. Additionally, since the generator and rotor speed measurements are carried out utilizing encoders, and due to possible malfunctions of the electrical components of the encoders, they can be faulty as well. The faults can be in the form of a fixed value that prevents the encoder from being updated with new values, or a changed gain factor on the measurements which causes the encoder to read more marks on the rotating part than are actually present [25].

At a basic level, in a WT system, faults can occur on the converter and the pitch actuator system. Faults in the pitch actuator cause changes in the dynamics due to three factors; a hydraulic leakage, a drop pressure in the pump wear, or a high air content in the hydraulic oil, where the latter may happen in various levels due to compressible nature of the air [51]. The source of converter faults is either in changed dynamics of the converter arising from an internal

fault in the converter's electronic components, or an offset in the converter torque estimation, which is more severe. The converter controller can deal with the faults in the electronic components, and since the torque balance in the WT power train is changed by torque offset, it is possible to detect and accommodate them [47]. System faults result in changes in the dynamic of parts of the system, which mainly happen in the drive-train. Although compared to the system dynamics, the drive-train friction coefficient changes more slowly with respect to time, it could be detected by observing the changes in the frequency spectrum of the vibration measurements. In this work, this fault is considered as a small change of the friction coefficient.

2.4 Fault Injection

This section briefly explains the type of changes happening in the fault-free model as each fault occurs. According to Table 1, faults F1, F2, and F3 correspond to fixed values on $\beta_{1,m1}$, $\beta_{2,m2}$, and $\beta_{3,m2}$, respectively. Each fault occurs in a certain time interval, as shown in Table 1. The fault F4 happens in the time interval of 3805-4400 seconds by changing the gain factors on $\omega_{r,m2}$ and $\omega_{g,m1}$ to 1.1 and 0.9, respectively. As two of the main pitch actuator faults, the hydraulic pressure drop (F5) and high air content in the oil (F6) are considered in this work. The effects of these faults are reflected in the damping ratio and natural frequency of the pitch system, where each one influences the system dynamics differently. A drop in the hydraulic pressure changes ω_n and ξ from their nominal values $\omega_{n,0}$ and ξ_0 to their low pressure values $\omega_{n,f}$ and ξ_f, which influences the pitch system dynamics. Under this gradual low-pressure fault, ω_n^2 and $\xi\omega_n$ in (4) can be modeled as follows [26]:

$$\omega_n^2 = \omega_{n,0}^2 + f.\left(\omega_{n,f}^2 - \omega_{n,0}^2\right), \tag{7a}$$

$$\xi\omega_n = \xi_0\omega_{n,0} + f.\left(\xi_f\omega_{n,f} - \xi_0\omega_{n,0}\right), \tag{7b}$$

where $f \in [0, 1]$ represents the fault indicator at which $f = 0$ and $f = 1$ correspond to the normal pressure and low pressure up to 50% pressure drop, respectively.

Changing the fault indicator corresponds to changes in the natural frequency and the damping ratio, where Table 2 presents the effects of their changes on the hydraulic pressure drop (F5) and the air content in the oil (F6). For a better demonstration of hydraulic pressure drop, the fault indicator is changed gradually, which corresponds to different values for ω_n and ξ, where in each step, the percentage of change in hydraulic pressure drop is given. Besides, $\omega_n = 5.73$ and $\xi = 0.45$ correspond to the maximum percentage of 15% change for the air content in the oil occurring during the time period of 3500-3600 seconds. Another considered fault is the friction changes in the drive-train (F7) which will be investigated in Section 5 with different levels of severity with 5%, 10%, 50%, and 100% increase in the coefficient.

Table 2: Different faults effect on the pitch system dynamics.

Faults	Fault indicator	ω_n (change %)	ξ (change %)	Fault (%)
Fault free	0.0	11.1100	0.600	0.0 %
Hydraulic pressure drop (HPD)	0.1	10.5952 (-4.63 %)	0.5953 (-0.78 %)	5 %
	0.2	10.0541 (-9.50 %)	0.5916 (-1.40 %)	10 %
	0.3	9.4822 (-14.65 %)	0.5895 (-1.75 %)	15 %
	0.4	8.8734 (-20.13 %)	0.5895 (-1.75 %)	20 %
	0.5	8.2197 (-26.01 %)	0.5927 (-1.21 %)	25 %
	0.6	7.5094 (-32.40 %)	0.6010 (+0.16 %)	30 %
	0.7	6.7244 (-39.47 %)	0.6178 (+2.96 %)	35 %
	0.8	5.8347 (-47.48 %)	0.6505 (+8.41 %)	40 %
	0.9	4.7823 (-56.95 %)	0.7187 (+19.78 %)	45 %
	1.0	3.4200 (-69.21 %)	0.9000 (+50.00 %)	50 %
High air content in the oil (HAC)	–	5.7300 (-48.42 %)	0.4500 (-25.00 %)	15 %

Figure 4 depicts the step responses of the pitch system to different fault situations. Accordingly, as the hydraulic pressure drops, it slows the pitch actuator dynamics, resulting in the degradation of pitching performance.

Remark 3. *The effects of pitch actuator faults on the pitch system are reflected in ω_n and ξ from their nominal values to faulty values by changing the fault indicator f, as expressed in (7). Accordingly, fault-free and faulty situations are being considered in the design process of the developed controller (16).*

3 Proposed Dynamic Weighted Parallel Firefly Algorithm

In this section, the conventional FA is first introduced, and then, the proposed dynamic weighted parallel FA will be investigated in detail.

3.1 Basic Principles of FA

FA is an optimization algorithm that mimes the social behavior of fireflies and their flashing light patterns [63]. The swarm of fireflies is randomly located in the search space, where each one represents a possible solution to the problem. Fireflies with better solutions acquire more light intensity, while other swarm members update their positions by moving toward brighter and more attractive fireflies. For simplicity of development, the FA utilizes the three rules: (a) the fireflies are unisex; thus, they only get drawn to brighter ones, (b) attractiveness corresponds to brightness. The less bright ones always move toward the brighter fireflies, and if no brighter one is left, it moves randomly, and (c) the analytical form of the problem affects the brightness of a firefly, where, brightness is proportional to the value of the objective function.

In the firefly algorithm, attractiveness is proportional to the light intensity seen by adjoining fireflies. Accordingly, since decreasing the distance from the

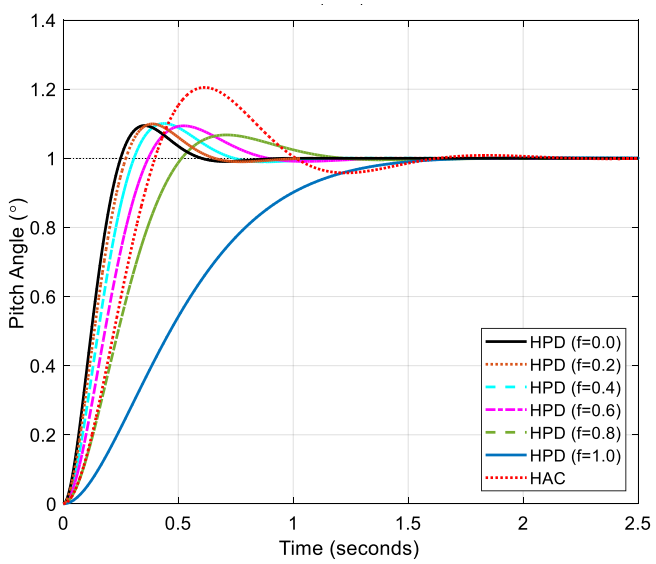

Figure 4: Step response of the pitch system under various fault conditions.

source leads to increment of light intensity, attractiveness increases as the distance between any two fireflies decreases. Two critical issues to be considered in FA are the attractiveness formulization and the light intensity variation. The light intensity $\mathfrak{I} = \mathfrak{I}_0 e^{-\gamma \mathfrak{r}^2}$ alters with the distance $\mathfrak{r}$, where γ and $\mathfrak{I}_0$ represent the light absorption coefficient and the initial light intensity, respectively. The firefly's attractiveness χ is defined as $\chi_{\mathfrak{r}} = \chi_0 e^{-\gamma \mathfrak{r}^2}$, where χ_0 denotes the attractiveness at $\mathfrak{r} = 0$. The Euclidian distance $\mathfrak{r}_{ij} = \|\mathfrak{X}_i - \mathfrak{X}_j\|_2$ or ℓ_2 norm can express the distance r_{ij} between any two fireflies i and j at $\mathfrak{X}_i$ and $\mathfrak{X}_j$.

Remark 4. *Considering the attractiveness $\chi_{\mathfrak{r}} = \chi_0 e^{-\gamma \mathfrak{r}^2}$, it can be seen that there are two limiting cases with firefly algorithm related to small and large values of γ (i.e. $\gamma \to 0$ and $\gamma \to \infty$). When γ tends to zero, the brightness and attractiveness become constant; in other words, a firefly is visible to all other fireflies. In contrast, when γ is very large, the attractiveness considerably decreases, and the fireflies are short-sighted or equivalently fly in a dense foggy environment. Large values of γ imply an almost randomly movement of fireflies, that refers to a random search procedure. As a result, the FA usually performs between these two cases, where the attractiveness coefficient plays a critical role in fireflies' movements.*

According to Remark 4, firefly i moves toward a more attractive (brighter) one as follows [63],

$$\mathfrak{X}_i(t+1) = \mathfrak{X}_i(t) + \chi_{\mathfrak{r}}\left(\mathfrak{X}_j(t) - \mathfrak{X}_i(t)\right) + \eta \cdot (\psi - 0.5), \tag{8}$$

where randomization is performed with η and ψ being random numbers within the interval [0,1].

It is worth noting that, the case $\gamma \to 0$ corresponds to a special case of PSO with $\chi_0 \approx 2$ [63]. However, although according to [63], $\chi_0 = 1$ is considered in the standard FA for most cases, other ranges of χ_0 are reported and used in the literature, where $0 < \chi_0 < 2$ has found to deliver the best performance [39, 35, 45, 23, 59, 53, 55]. Hence, initializing χ_0 within the interval $(0, 2)$ (corresponding to $|1 - \chi_{\mathfrak{r}}| < 1$) is a reasonable choice to achieve better performances from the firefly algorithm. However, other values of $\chi_0 \geq 2$ can also be set for the algorithm.

3.2 Proposed Dynamic Weighted Parallel FA

Although the conventional FA has its advantages, it also has some shortcomings, such as premature convergence leading to being trapped in local minima and lack of a suitable trade-off between exploitation and exploration abilities [23]. Besides, in FA, the brightest member always moves randomly in the search area which tends to decrease its intensity, especially at high dimensions. In this regard, many studies have incorporated external global/local search procedures (algorithms) into the FA, in order to enhance its search abilities and performance [46, 48, 3].

This study proposes a modified version of FA that explores the search space with a well-connected weighted parallel strategy, which effectively accelerates the convergence speed of the conventional FA while reducing the possibility of becoming trapped in local optima. In addition, taking advantage of a dynamic switching coefficient, as the damping coefficient decreases, the switching coefficient increases to let the algorithm to perform more accurate exploitations.

A population of randomly generated fireflies is firstly initialized, where each individual stands for a possible solution. The damping coefficient $R = 2\left(1 - n_{it}/n_{it,\max}\right)$ is incorporated into (8) to improve the movement pattern of individuals in the exploration process as they move towards the brightest one, as follows:

$$\mathfrak{X}_i\left(t+1\right) = \mathfrak{X}_i\left(t\right) + R\chi_{\mathfrak{r}}\left(\mathfrak{X}_j\left(t\right) - \mathfrak{X}_i\left(t\right)\right) + \eta \cdot \left(\psi - 0.5\right), \tag{9}$$

where n_{it} and $n_{it,\max}$ denote the number of the current iteration and max iteration, respectively, and R linearly decreases from two to zero over the number of iterations.

The objective value of each individual is evaluated to determine the best solution. In the exploitation process, the population can be divided into n number of semi-independent subgroups with equal number of members, where within each subgroup, the individuals are updated in parallel aiming at finding better solutions. The weighting coefficient associated with each subgroup can be calculated by $w_{sg} = \frac{2.5 \times n_{sg}}{n_{sg}+1}$, where n_{sg} denotes the number of predefined groups.

Remark 5. *It should be noted that for each individual, the weighting coefficient associated with its current group should be twice the weighting coefficient associated with other groups. In addition, the total number of members in the*

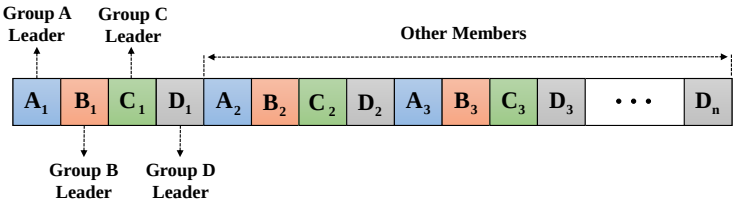

Figure 5: The subgroups members assignment procedure.

populations (n_p) should be divisible by the defined number of subgroups (n_{sg}); thereby, all subgroups would have the same number of fireflies through the individual assignment procedure.

Remark 6. *In this paper, the subgroups members assignment procedure presented by (10) and Fig. 5, as well as the position update (11) are presented for the case $n_{sg} = 4$. The procedure for other values of n_{sg} can be carried out analogously.*

The population is divided into four semi-independent subgroups $\{A, B, C, D\}$, where the best four individuals of the exploration process are assigned as subgroup leaders (see Fig. 5), and other members join the subgroups as follows:

$$A = \{A_1, A_2, A_3, ..., A_n\} = \{1, 5, 9, ..., A_n\}, \tag{10a}$$
$$B = \{B_1, B_2, B_3, ..., B_n\} = \{2, 6, 10, ..., B_n\}, \tag{10b}$$
$$C = \{C_1, C_2, C_3, ..., C_n\} = \{3, 7, 11, ..., C_n\}, \tag{10c}$$
$$D = \{D_1, D_2, D_3, ..., D_n\} = \{4, 8, 12, ..., D_n\}, \tag{10d}$$

where $n = n_p/n_{sg} = 4$.

The weighting coefficients w_A, w_B, w_C, and w_D are defined for subgroups in order to establish a connection between the whole population and subgroups leaders to exchange their location information. The weighting coefficients are chosen in such a way that $w_{total} = w_A + w_B + w_C + w_D = 10$. Accordingly, in every step of the exploitation process, the individuals update their positions, taking into account the position of the leaders with different weights. The switching coefficient $\Theta = \text{round}\left(\kappa/e^R\right)$ represents the number of exploitation steps and dynamically makes a trade-off between the exploitation and exploration functions of the algorithm, where κ is the switching coefficient sensitivity parameter, an arbitrary positive number that determines the depth of the exploitation process. Members of each subgroup are updated according to (11),

taking into consideration their subgroup leader and other leaders as follows:

$$\mathfrak{X}(t+1) = \mathfrak{X} + R\chi_{\mathfrak{r}}(\mathfrak{X}_{GL} - \mathfrak{X}) + \left(\begin{array}{l} w_A r_1 (\mathfrak{X}_A - \mathfrak{X}) + w_B r_2 (\mathfrak{X}_B - \mathfrak{X}) \\ +w_C r_3 (\mathfrak{X}_C - \mathfrak{X}) + w_D r_4 (\mathfrak{X}_D - \mathfrak{X}) \end{array} \right) / w_{total} + \eta \cdot (\psi - 0.5), \tag{11}$$

where $\mathfrak{X}_{GL}$ represents the subgroup leader's position and $r_i \in (0,1), i = 1,2,3,4$ is a random number that directs the movement.

Remark 7. *Considering (11), suppose that the position of a firefly in group B is to be updated. Since the firefly belongs to group B, $\mathfrak{X}_{GL} \equiv \mathfrak{X}_B$ denotes the position of its subgroup leader. The damping coefficient R in the second term of (11) linearly decreases from two to zero over the number of iterations; hence, R increases the exploration performance at the beginning of the algorithm, and as the algorithm proceeds and the fireflies approach the subgroup leader, it increases the exploitation performance around the leader. This helps the individuals perform a semi-local search behavior to better search space around the optimum solution. The third term in (11) benefits from the position of other subgroups' leaders and their associated weighting coefficients (w_{sg}), giving more weight to each firefly's current leader as mentioned in Remark 5. This helps the fireflies consider other leaders' positions as they move towards their own group's leader, resulting in increased diversity of movements.*

After each exploitation step, the individuals are evaluated, and the best solution becomes the leader of the subgroup. By completing the exploitation process, the four subgroups are merged into one big group so that agents can share location information amongst the search space. According to (11), it is observed that utilizing the subgroup method increases the information exchange between individuals, and effectively increases the algorithm's convergence speed. Besides, despite the conventional FA, the brightest member's movement is not entirely random, and it performs a more sophisticated search through the exploitation process. Algorithm 1 presents the pseudo-code of the DWPFA, where NFE represents the number of function evaluations.

3.3 On the computational complexity of the proposed DWPFA

The computational complexity of an EA is a pointer of its execution time and is controlled by its structure. Let $O(F)$ denote the computational complexity of the fitness evaluation function $F(\cdot)$. The conventional FA has a computational complexity of $O\left(It_{max} \times n_p^2 \times F\right)$ [60], where It_{max} represents the maximum number of iterations and n_p denotes the population size. However, compared to the relatively small n_p, the study of the number of attractions and movements during It_{max} iterations could be more important. It is noteworthy that although larger n_p can result in significant benefits in terms of the algorithm's performance, its negative consequence is a substantial increase in calculation time.

Algorithm 1 Pseudo-code of DWPFA.

```
Objective function f(𝔛),   𝔛 = (𝔛_1, 𝔛_2, ..., 𝔛_d)
NFE=0;
Initialize the population randomly;
Define the light absorption coefficient γ;
while not stopping criterion do
    for i = 1 : n_p do
        for j = 1 : n_p do
            if ℑ_j > ℑ_i then
                Move individual i towards j in all dimensions;
            end if
            Update 𝔛_i using (8);
            Update light intensity ℑ_i;
            NFE=0;
        end for
    end for
    Rank the individuals and find the current best;
    Define the subgroups and assign the leaders and members;
    for k = 1 : Θ do
        for i = 2 : n_p do
            if ℑ_GL > ℑ_i then
                Move individual i towards the subgroup leader in all dimensions;
            end if
            Update 𝔛_i using (11);
            Update light intensity ℑ_i;
            NFE=0;
            Rank the individuals and define the subgroup best as leader;
        end for
    end for
    Merge the subgroups into one group;
    Rank the individuals and find the current best;
end while
```

In the conventional FA, each firefly is compared with all other members of the population, and at each comparison step, one of the agents is moved. Hence, it can be concluded that each agent is moved with an average of $(n_p - 1)/2$ times per iteration [60]. Consequently, at each iteration of the conventional FA, $n_p \times (n_p - 1)/2$ attractions are performed. It should be noted that, although the attraction enables the agents to find new optimal solutions, if a high number of attractions does not come along with better exploitation performance (leading to higher convergence speed of the algorithm), excessive attractions can induce oscillations during the search process, and simultaneously impose a high computational burden with less optimal solutions. Accordingly, an effective EA should provide a trade-off between the accuracy and computational cost. In this regard, efforts have been made to enhance the FAs performance considering the abovementioned objectives (*i.e.* better convergence and keeping the computational complexity as low as possible). Authors in [35] proposed a modified FA with Gaussian disturbance and local search (GDLSFA). As reported, the solutions' accuracy and convergence speed has increased; however, compared to the conventional FA, a higher number of attractions $3/2 \times n_p \times (n_p - 1)$ is achieved.

Authors in [45] developed a memetic FA (MFA) to enhance the solutions' accuracy of FA; however, the optimization objective is achieved with an increased number of attractions as $n_p \times (n_p - 1)$ due to the embedded exploitation process.

In the proposed DWPFA algorithm, the exploration process has the same number of attractions as the conventional FA has. During the exploitation process, each firefly moves with an average of Θ times per iteration. Consequently, at each iteration, $\Theta \times n_{sg} \times (n_p - n_{sg})$ attractions are performed during the exploitation process. That is to say, the total number of each agent's movement per iteration is $(n_p - 1)/2 + \Theta$, with an attraction number of $n_p \times (n_p - 1)/2 + (\Theta \times n_{sg} \times (n_p - n_{sg}))$. As a result, the total number of attractions per iteration is within the range of $[n_p \times (n_p - 1)/2 + (n_p - n_{sg}), n_p \times (n_p - 1)/2 + (\kappa \times n_{sg} \times (n_p - n_{sg}))]$, which is slightly more than that of FA with the same computational complexity. In our experiments, $\kappa = 4$ is a good selection for DWPFA. Figure 6 illustrates a comparative study on the number of attractions associated with FA, GDLSFA, MFA, and the proposed DWPFA with different numbers of populations. According to the foregoing analysis and Fig. 6, it can be observed that the number of attractions of DWPFA under the full attraction model is much lower than the abovementioned studies, showing much less imposed computational burden. In addition, considering the DWPFA's superior exploitation performance and solutions accuracy compared to FA, the slight increase in its computational burden is negligible.

4 Proposed Extended Memory Pitch Control Strategy

PID controllers have been extensively applied in industrial applications owing to their design and implementation simplicity, low computational complexity, and robustness in the presence of external disturbances. FOPID controllers have demonstrated more flexibility to controller design, and more robustness in comparison with conventional PIDs [38, 58]. FOPIDs involve two additional degrees of freedom to the conventional PID; namely, the non-integer integral δ and derivative μ orders, leading to a more promising performance with five adjustable parameters [38]. Since the wind energy level and direction changes continuously, and the pitch actuation system cannot provide immediate precise responses, considering the slight time delay between these changes can play an effective role in enhancing the control performance. Accordingly, it may be desirable to keep track of past effective pitch angles, as they serve as memory of the pitch system characteristics. Thus, extending the memory of pitch angles results in acquiring more data and a more comprehensive perspective of the system behavior. In this regard, in this paper, the incorporation of fractional-calculus-based extended memory of pitch angles with an optimal FOPID controller is proposed in order to generate the desired pitch angle reference for the WT in region III.

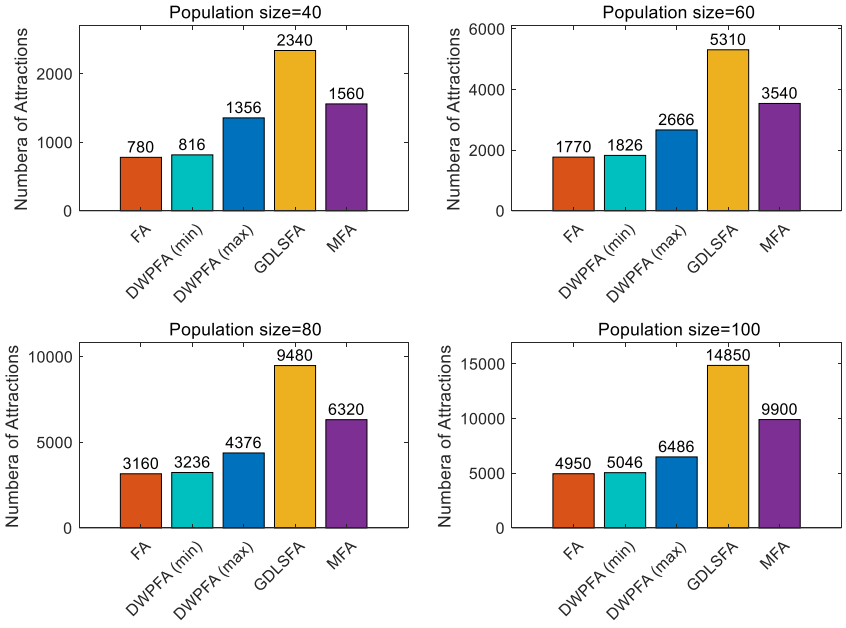

Figure 6: Comparative illustration on changes in the number of attractions associated with FA, GDLSFA [35], MFA [45], and DWPFA.

Fractional calculus generalizes the integration and differentiation of a function to non-integer order, represented by $\mathfrak{D}^{\sigma}$ operator, where σ denotes the fractional order. Fractional-order derivatives and integrals can be derived through various definitions [52]. However, the Grunwald–Letnikov (G-L) approximation is the most prominent definition in fractional-order calculus [39, 5, 49, 44]. The G-L fractional derivative of a function $x(t)$ in discrete-time is expressed as follows.

$$\mathfrak{D}^{\sigma}\left[x(t)\right] = \frac{1}{T^{\sigma}} \sum_{k=0}^{\Upsilon} (-1)^{k} \frac{\Gamma(\sigma+1)\, x(t-kT)}{\Gamma(k+1)\,\Gamma(\sigma-k+1)}, \tag{12}$$

where T and Υ denote the sampling period in [s] and the truncation order, respectively. $\Gamma(\cdot)$ is Euler's gamma function, where $\Gamma(z) = \int_0^{\infty} \mathfrak{n}^{z-1} e^{-\mathfrak{n}} \mathrm{d}\mathfrak{n}$, $\mathrm{Re}(z) > 0$.

The main control actions in region III are carried out by the pitch system, through designing a controller to minimize the error $e(t) = \omega_{nom} - \omega_g(t)$. The proposed EM-FOPID is implemented in the time domain as follows,

$$\beta_r(t) = \beta_r(t-1) + K_p e(t) + K_i \mathfrak{D}^{-\delta} e(t) + K_d \mathfrak{D}^{\mu} e(t). \tag{13}$$

Remark 8. *According to fractional calculus concepts, despite the integer-order derivative that represents a finite series, the fractional-order derivative involves an infinite number of terms [52]. This characteristic leads to acquiring a memory of all past pitch angles and can be controlled by the fractional order $0 \leq \sigma \leq 1$. It is noteworthy that the controller is implemented in discrete time to benefit from the memory preservation characteristics of the discrete-time G-L fractional derivative. Hence, all considered time-dependent variables discrete-time.*

In this perspective, (13) can be rearranged as

$$\beta_r(t) - \beta_r(t-1) = K_p e(t) + K_i \mathfrak{D}^{-\delta} e(t) + K_d \mathfrak{D}^{\mu} e(t). \tag{14}$$

Remark 9. *Assuming $T = 1$, the left side of (14) represents the G-L fractional derivative with order $\sigma = 1$, which leads to:*

$$\mathfrak{D}^{\sigma}\left[\beta_r(t)\right] = K_p e(t) + K_i \mathfrak{D}^{-\delta} e(t) + K_d \mathfrak{D}^{\mu} e(t). \tag{15}$$

Thus, considering the first $\Upsilon = 4$ terms of differential derivative (12), (15) can be rewritten as follows:

$$\begin{aligned} \beta_r(t) = {} & \sigma \beta_r(t-1) \\ & + \frac{1}{2!}\sigma(1-\sigma)\beta_r(t-2) + \frac{1}{3!}\sigma(1-\sigma)(2-\sigma)\beta_r(t-3) \\ & + \frac{1}{4!}\sigma(1-\sigma)(2-\sigma)(3-\sigma)\beta_r(t-4) \\ & + K_p e(t) + K_i \mathfrak{D}^{-\delta} e(t) + K_d \mathfrak{D}^{\mu} e(t). \end{aligned} \tag{16}$$

Remark 10. *Experimental tests have shown that larger values of $\Upsilon > 4$ lead to similar results [38].*

In this work, the proposed DWPFA is utilized to achieve the optimum parameters $\{K_p, K_i, \delta, K_d, \mu\}$ for the proposed EM-FOPID controller. To simultaneously improve the transient and steady-state error, the integral of time multiplied squared error (ITSE) index is incorporated in the objective function. Also, to avoid large control signals and simultaneously reduce its deviations, which may lead to actuator saturation, the integral of squared control signal (ISCO) is embedded alongside the ITSE. The objective function evaluated to determine the controller parameters is defined as $J = \int_0^\infty \left[te^2(t) + u^2(t)\right] \mathrm{dt} + V_c$ subjected to the pitch actuator constraints stated in Remark 2, where $u(t)$ denotes the controller's output, and V_c is the constraints violation coefficient such that V_c is a very small value when the constraints are respected during the optimal design procedure, and $V_c = \infty$ in case of any violations. Figure 7 depicts the procedure of tuning the proposed EM-FOPID using DWPFA algorithm.

Remark 11. *In this study, the pitch actuator (4) is modeled as a second-order transfer function with constraints to ensure the feasible operational range of the WT [47]. A common drawback associated with some studies, such as [57], is that the pitch actuator constraints are not considered in the controller's design procedure, which may cause the wind-up phenomenon and consequently degrade the WT performance if the control input reaches the saturation limits. In this regard, in many studies magnitude and rate limiters are implemented to deal with the constraints [47, 30, 53]. In this work, constraints are explicitly checked at each iteration, with high penalties added to the objective function in case of any violations, to ensure adherence to constraints in future iterations.*

5 Simulation Results and Discussions

The performance of the proposed DWPFA algorithm is first evaluated in comparison with other EAs through solving the CEC2017 mathematical benchmark functions. Then, the proposed DWPFA-optimized EM-FOPID controller is utilized to adjust the pitch angle of WT blades, where its performance is compared to PI and DWPFA-optimized conventional FOPID approaches under sensor, actuator, and system faults. Simulations are carried out using MATLAB R2020a (9.8.0.1417392) 64-bit, on a ASUS laptop with 64-bit win10 operating system, processor: Intel® core™ i7-8550U CPU 2.50 GHz, installed memory: 8.00 GB, and VGA: GeForce NVidia 620M-4GB.

5.1 Performance Evaluation of DWPFA

In this section, well-defined CEC2017 special session mathematical benchmark functions [8] are used as objective functions, to assess the performance of the proposed DWPFA compared to other EAs. In this regard, 30 test functions are used and are categorized as follows: unimodal functions ($f_1 - f_3$), simple multimodal functions ($f_4 - f_{10}$), hybrid functions ($f_{11} - f_{20}$), and composition

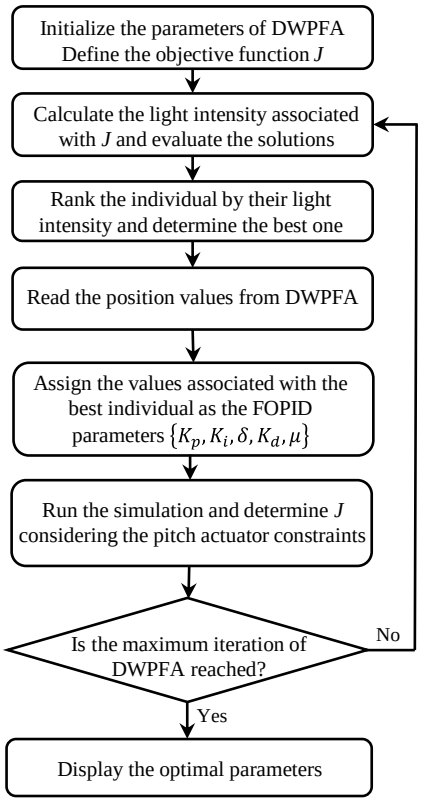

Figure 7: The flowchart of the proposed EM-FOPID tuning using DWPFA.

functions ($f_{21} - f_{30}$). To testify the performance of the proposed DWPFA algorithm, it is compared with PSO, fractional PSO-based memetic algorithm (FPSOMA) [38], grey wolf optimizer (GWO), enhanced GWO (EGWO) [34], enhanced BFO (CCGBFO) [19], FA, and fractional order FA [39]. In the proposed algorithm, $\gamma = 1$ and $\chi_0 = 1.2$ are considered. In the performed experiments, the benchmark functions' dimension is set to $D = 50$, and all EAs have a population size of 40. Each algorithm is run 200 times independently for each test instance, and the allowed number of maximum function evaluation (NFE) is set to $10000 \times D$. Since achieving zero error on CEC2017 functions is a demanding task for the algorithms, the constant ε is defined as an acceptable threshold value of a satisfactory solution near the optimal solution for each function. In this work, $\varepsilon = 50$ is set for ($f_1 - f_{20}$) and $\varepsilon = 500$ is set for ($f_{21} - f_{30}$). In Table 3, the first column illustrates the sequence of 30 CEC2017 benchmark functions, and the next columns show the mean results achieved, while the minimum value obtained for each function is emphasized in bold. In addition, a comparison is carried out by reporting the experimental results of CEC2017 test suit as a logarithmic radar graph (spider plot) in Fig. 8.

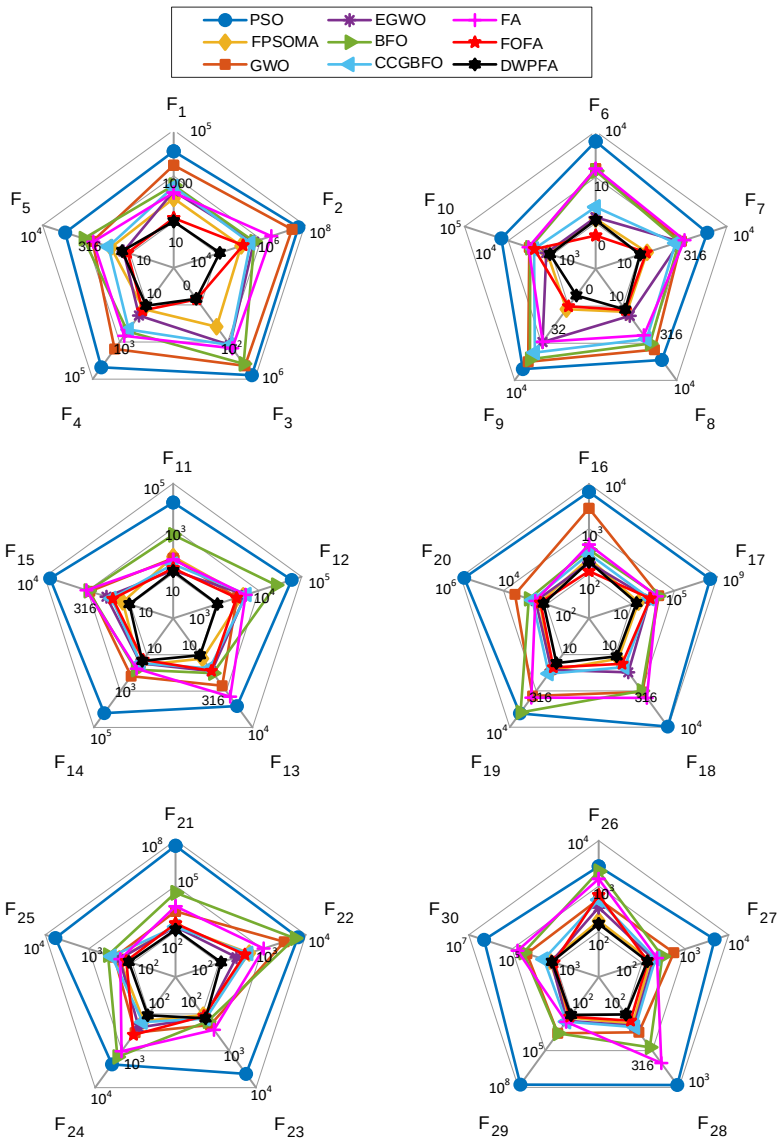

Figure 8: Spider plot of results achieved on CEC2017 benchmark functions.

Table 3: Obtained results for CEC2017 test functions with $D = 50$.

Fun.	PSO	FPSOMA	GWO	EGWO	BFO	CCGBFO	FA	FOFA	DWPFA
f_1	1.20×10^4	1.06×10^2	2.95×10^3	2.17×10^2	3.94×10^2	3.41×10^2	1.88×10^2	1.47×10^1	1.05×10^1
f_2	5.08×10^7	1.22×10^5	2.61×10^7	4.70×10^5	4.90×10^5	3.22×10^5	2.93×10^6	1.49×10^5	1.29×10^4
f_3	3.75×10^5	2.25×10^0	3.64×10^4	2.13×10^2	2.23×10^4	2.14×10^2	4.69×10^2	2.50×10^{-3}	2.19×10^{-3}
f_4	2.30×10^4	1.69×10^1	2.37×10^3	3.73×10^1	2.67×10^2	2.12×10^2	4.63×10^2	1.99×10^1	1.10×10^1
f_5	1.68×10^3	4.49×10^1	1.94×10^2	1.67×10^1	3.89×10^2	5.53×10^1	1.82×10^2	1.21×10^1	1.86×10^1
f_6	2.19×10^3	1.89×10^{-2}	3.65×10^1	2.40×10^{-2}	2.60×10^1	1.13×10^{-1}	3.60×10^1	1.53×10^{-3}	1.58×10^{-2}
f_7	2.14×10^3	1.84×10^1	2.78×10^2	2.44×10^2	2.30×10^2	1.75×10^2	3.54×10^2	1.73×10^1	1.06×10^1
f_8	1.50×10^3	1.71×10^1	5.75×10^2	2.47×10^1	3.26×10^2	2.13×10^2	1.50×10^2	1.43×10^1	1.38×10^1
f_9	1.74×10^3	1.61×10^{-1}	5.59×10^2	2.63×10^1	3.80×10^2	1.34×10^2	2.54×10^1	1.03×10^{-1}	1.92×10^{-2}
f_{10}	1.44×10^4	1.28×10^3	3.30×10^3	1.35×10^3	3.16×10^3	2.57×10^3	3.44×10^3	2.51×10^3	1.11×10^3
f_{11}	1.43×10^4	5.57×10^1	4.75×10^1	1.37×10^1	5.19×10^2	2.57×10^1	4.10×10^1	1.69×10^1	1.26×10^1
f_{12}	5.87×10^4	4.32×10^3	3.10×10^3	4.89×10^3	2.51×10^4	5.21×10^3	4.93×10^3	3.02×10^3	1.08×10^3
f_{13}	1.31×10^3	1.45×10^1	1.89×10^2	4.77×10^1	5.75×10^1	3.65×10^1	5.38×10^2	4.62×10^1	1.05×10^1
f_{14}	1.63×10^4	3.77×10^1	1.51×10^2	2.96×10^1	6.65×10^1	2.92×10^1	5.83×10^1	1.89×10^1	2.17×10^1
f_{15}	6.52×10^3	1.84×10^1	2.63×10^2	6.87×10^1	2.95×10^2	4.51×10^1	3.47×10^2	4.13×10^1	1.09×10^1
f_{16}	6.62×10^3	2.19×10^2	2.85×10^3	1.75×10^2	3.39×10^2	2.72×10^2	4.41×10^2	1.13×10^2	1.83×10^2
f_{17}	2.24×10^8	4.28×10^1	3.43×10^3	4.58×10^2	2.86×10^3	5.03×10^2	2.30×10^3	5.45×10^2	2.85×10^1
f_{18}	9.42×10^3	1.47×10^1	3.49×10^2	5.39×10^1	3.12×10^2	3.30×10^1	6.12×10^2	2.39×10^1	1.18×10^1
f_{19}	2.68×10^3	4.52×10^1	5.10×10^2	4.29×10^1	2.44×10^3	6.09×10^1	6.01×10^2	3.36×10^1	2.23×10^1
f_{20}	6.94×10^5	1.97×10^2	2.94×10^3	3.14×10^2	6.57×10^2	4.40×10^2	3.27×10^2	1.87×10^2	1.34×10^2
f_{21}	4.36×10^7	2.36×10^2	2.11×10^3	2.01×10^2	3.51×10^4	2.01×10^2	4.54×10^3	3.37×10^2	1.29×10^2
f_{22}	6.61×10^3	4.99×10^2	3.18×10^3	2.53×10^2	5.60×10^3	4.75×10^2	1.07×10^3	3.95×10^2	1.12×10^2
f_{23}	4.24×10^3	1.08×10^2	1.99×10^2	1.24×10^2	1.58×10^2	1.34×10^2	2.70×10^2	1.14×10^2	1.32×10^2
f_{24}	2.34×10^3	1.55×10^2	3.23×10^2	2.34×10^2	1.47×10^3	1.67×10^2	1.03×10^3	3.71×10^2	1.09×10^2
f_{25}	5.86×10^3	1.97×10^2	2.56×10^2	1.95×10^2	3.59×10^2	2.95×10^2	1.87×10^2	1.36×10^2	1.19×10^2
f_{26}	2.71×10^3	1.72×10^2	5.12×10^2	3.39×10^2	2.11×10^3	4.99×10^2	1.44×10^3	6.35×10^2	1.47×10^2
f_{27}	4.75×10^3	1.54×10^2	5.42×10^2	1.84×10^2	2.95×10^2	2.04×10^2	2.22×10^2	1.36×10^2	1.33×10^2
f_{28}	9.28×10^2	1.37×10^2	1.78×10^2	1.45×10^2	2.85×10^2	1.52×10^2	4.67×10^2	1.24×10^2	1.02×10^2
f_{29}	5.96×10^7	1.95×10^2	3.89×10^3	3.67×10^2	3.81×10^3	4.33×10^2	4.47×10^2	1.70×10^2	1.27×10^2
f_{30}	1.96×10^6	1.79×10^3	2.08×10^4	1.52×10^3	2.97×10^4	3.43×10^3	5.61×10^4	1.16×10^3	1.46×10^3

According to Table 3, it can be seen that the optimization performance obtained by the proposed DWPFA are markedly less than other algorithms in solving real-parameter optimization problems in the CEC2017 suite. Results illustrated in Table 3 and Fig. 8 demonstrates the superiority of DWPFA, yielding outstanding performance with 24 best solutions achieved out of 30 test problems, followed by FOFA and FPSOMA with 5 and 1 best solutions, respectively.

To intuitively show the performances of DWPFA, Fig. 9 is plotted to illustrate the box-and-whisker diagrams of solutions obtained on different selected problems of each category, for all 200 runs with $D = 50$. The vertical and horizontal axes indicate the optimal solution and the nine algorithms, respectively. From Fig. 9 it is observed that despite the high complexity of functions, DWPFA provides promising results maintaining fewer values and shorter distribution of solutions comparing to other algorithms under evaluation, indicating excellent and steady performances of it. This implies that DWPFA is more effective for optimizing functions, and thus its superiority is apparent. In order to determine whether to accept or reject the null hypothesis, non-parametric tests can be utilized. Non-parametric tests determine whether the data sets to be compared have the same variance [18]. Accordingly, to statistically compare and analyze the quality of the solution, two non-parametric statistical hypothesis tests were used to compare the results, namely the Friedman test and the Friedman Aligned Ranks test [18]. The null hypothesis for the Friedman test represents the equality of medians between the populations, while the ranks

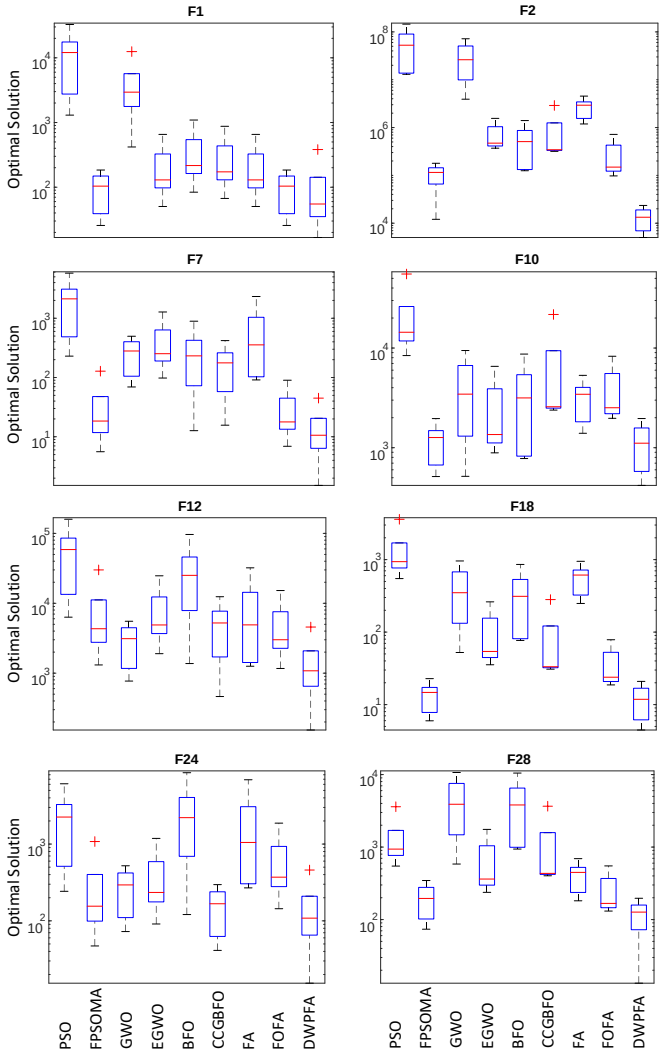

Figure 9: The box-and-whisker comparative performance diagrams on the selected functions, $D = 50$.

assigned to the resulting differences (aligned observations) are called Friedman aligned ranks. Table 4 demonstrates the Friedman and Friedman Aligned test results sorted by the performance order "Rank". The results indicate that DWPFA obtains the best rank, followed by FOFA and FPSOMA. The ranking of all algorithms in both tests is the same, except for EGWO, where its rank improved from fifth to fourth, taking CCGBFO's place.

Table 4: The Friedman and Friedman Aligned test results over CEC2017 test functions, $D = 50$.

Friedman			Friedman Aligned		
Algorithm	Score	Rank	Algorithm	Score	Rank
DWPFA	2.1925	1	DWPFA	33.263	1
FOFA	2.3852	2	FOFA	34.174	2
FPSOMA	2.5682	3	FPSOMA	34.368	3
CCGBFO	2.7268	4	EGWO	34.404	4
EGWO	3.1481	5	CCGBFO	34.671	5
FA	3.1692	6	FA	35.816	6
BFO	3.3744	7	BFO	36.221	7
GWO	3.5760	8	GWO	36.414	8
PSO	3.8767	9	PSO	38.162	9

5.2 Numerical Example

This section investigates the closed-loop performance of the proposed EM-FOPID approach compared to the results obtained by other relevant methods in terms of ITSE performance criterion. Consider the following system adopted from literature [38],

$$m_1 \mathfrak{D}^{n_1} x(t) + m_2 \mathfrak{D}^{n_2} x(t) + m_3 x(t) = u(t), \tag{17}$$

where $m_1 = 0.8$, $m_2 = 0.9$, $m_3 = 1$, $n_1 = 2.2$, and $n_2 = 0.5$.

The control variable $u(t)$ can be considered as

$$u(t) = u(t-1) + K_p e(t) + K_i \mathfrak{D}^{-\delta} e(t) + K_d \mathfrak{D}^{\mu} e(t). \tag{18}$$

Considering Remark 9 and employing the extended memory characteristics, (18) can be rewritten as,

$$\begin{aligned} u(t) &= \sigma u(t-1) \\ &+ \frac{1}{2!}\sigma(1-\sigma) u(t-2) + \frac{1}{3!}\sigma(1-\sigma)(2-\sigma) u(t-3) \\ &+ \frac{1}{4!}\sigma(1-\sigma)(2-\sigma)(3-\sigma) u(t-4) \\ &+ K_p e(t) + K_i \mathfrak{D}^{-\delta} e(t) + K_d \mathfrak{D}^{\mu} e(t). \end{aligned} \tag{19}$$

Table 5 summarizes the step responses obtained by the controllers under study. Accordingly, it is observed that, compared to PID controllers, the FOPIDs demonstrate superior performance in terms of maximum overshoot and rise time. In addition, the proposed DWPFA algorithm delivers better tuning performance compared to other algorithms. Comparison study involving DWPFA-optimized EM-FOPID and other methods validates the effectiveness of embedding the memory characteristics to the controller. Accordingly, the DWPFA-optimized EM-FOPID outperforms all other methods and demonstrates more preferable performance with a maximum overshoot of 0.75% and

Table 5: Performance comparison of the controllers.

Controller	K_p	K_i	δ	K_d	μ	σ	Overshoot (%)	Rise Time [s]
PID Ziegler–Nichols [38]	16.6281	9.4422	-	7.2230	-	-	25.92	0.223
PID-DWPFA	5.3100	10.8620	-	18.1150	-	-	12.28	0.137
FOPID-FPSOMA [38]	393.9550	353.9850	0.12	117.8490	1.2240	-	1.73	0.003
FOPID-DE [16]	21.2200	1.3700	0.92	12.0500	0.93	-	7.69	0.023
FOPID-DWPFA	36.2200	27.0500	0.24	112.2000	1.13	-	1.25	0.003
EM-FOPID-DWPFA	24.8400	19.8540	0.16	84.6800	1.34	0.7	0.75	0.002

Table 6: WT model parameters.

Parameter	Value	Unit	Parameter	Value	Unit
R	57.5	m	D_{dt}	775.49	Nms/rad
ρ	1.225	kg/m^3	B_g	46.6	Nms/rad
ω_n	11.11	rad/s	N_g	95	-
ξ	0.6	-	η_{dt}	0.97	-
J_r	55E+06	kg m^2	α_{gc}	50	rad/s
J_g	390	kg m^2	ω_{nom}	162	rad/s
K_{dt}	2.7E+09	Nm/rad	K_{opt}	1.2171	-

rise time of 0.002 seconds, followed by the DWPFA-optimized FOPID with 1.25% overshoot and 0.003 seconds. In addition, although the FPSOMA-based FOPID [38] provides an acceptable performance, large controller gains K_p, K_i, and K_d are required, whereas, with similar fractional integral and derivative orders δ and μ, the impact of incorporating the memory effects to the controller has led to smaller controller gains with an even better control performance.

5.3 Optimal Pitch Angle Control of WT

In this section, the proposed DWPFA-optimized EM-FOPID is applied to generate the desired pitch angle reference of WT blades in region III, and its performance is evaluated with respect to the conventional PI and DWPFA-optimized conventional FOPID approaches under fault-free and faulty conditions. In this study, it is assumed that all required system signals are available for the controller. In practice, it is often necessary to estimate the wind speed and other signals. The WT model parameters are listed in Table 6, and the WT is subjected to sensor, actuator, and system faults described in Table 1. The proposed DWPFA algorithm is used to tune the controller parameters (illustrated in Table 7), and comparative simulations are conducted to validate the efficiency of the proposed optimal EM-FOPID.

A critical issue in designing a controller is to ensure closed-loop stability; this applies whether the system is linear or non-linear. However, although it can generally be achieved for linear systems and some classes of non-linear systems, analytical investigation of the closed-loop stability for FOPID controllers for a 4.8-MW WT is not a straightforward task due to the unavailability of the exact dynamic model in Laplace domain representation and the nonlineari-

Table 7: Controllers parameters for the WT system.

Controller	K_p	K_i	δ	K_d	μ	σ
Conventional PI [47]	4	1	-	-	-	-
DWPFA-FOPID	8.2	3	0.7	2	0.5	-
DWPFA-EM-FOPID	5.5	2.1	0.8	1.8	0.45	0.7

ties associated with the WT system. On the contrary, PID/FOPID controllers generally have the ability to destabilize a system if they are poorly designed. However, since the objective function uses ITSE, the optimization algorithm is expected to ensure stability and avoid any parameters that destabilize the system. Hence, if a set of control parameters cause instability at any wind speeds, the cost function would be a large value, and unstable modes would not be found to be optimal and will be avoided over successive generations. Consequently, the closed-loop system can be guaranteed to remain stable during the optimization process.

The design parameters are considered within the search ranges $-500 < (K_p, K_i, K_d) < 500$, and $0 < (\delta, \mu) < 2$. It is often essential that the controller's parameters are not chosen close to the marginal stability regions, which may lead to performance degradation in this particular case. Hence, to guarantee the system's stability, the controller's gains K_p, K_i, and K_d must be non-negative, and the fractional orders δ and μ should be chosen such that they maintain a trade-off between a) getting as far as possible from stability margins and b) providing desirable integration (leading to higher precision) and derivation (leading to more stability) performances. The marginal stability regions are depicted in Fig. 10(a) via Venn diagram. It is worth mentioning that, although the whole area $0 < (\delta, \mu) < 2$ (as shown in Fig. 10(b)) maintain the stability and acceptable performance, the golden zone $0.3 < (\delta, \mu) < 0.95$ has found to deliver the best performance, where, as shown in Fig. 10(b) the optimal integral and derivative orders $\delta = 0.8$ and $\mu = 0.45$ are found within this zone.

The wind profile covering a wind speed range of 5-20 m/s along with the occurrence time intervals of each fault scenario is depicted in Fig. 11. It consists of slow wind variations $\upsilon_m(t)$, stochastic wind behavior $\upsilon_s(t)$, the wind shear effects $\upsilon_{ws}(t)$, and the tower shadow effects $\upsilon_{ts}(t)$ [22] expressed as follows:

$$\upsilon_w(t) = \upsilon_m(t) + \upsilon_s(t) + \upsilon_{ws}(t) + \upsilon_{ts}(t). \tag{20}$$

According to Table 1, four different sensor fault scenarios (F1-F4) with low levels of severity occur between the time intervals of 2000-4400 seconds. Also, two actuator faults (F5, F6) and one system fault (F7) with medium and high levels of severity occur between the time intervals of 2900-3600 seconds and 4100-4300 seconds, respectively. Simulations are performed using MATLAB/Simulink environment for the WT model presented in Section 2.

Remark 12. *When actuator faults (hydraulic pressure drop or increment of air content in the oil) occur, the pitch angle changes accordingly, which degrades*

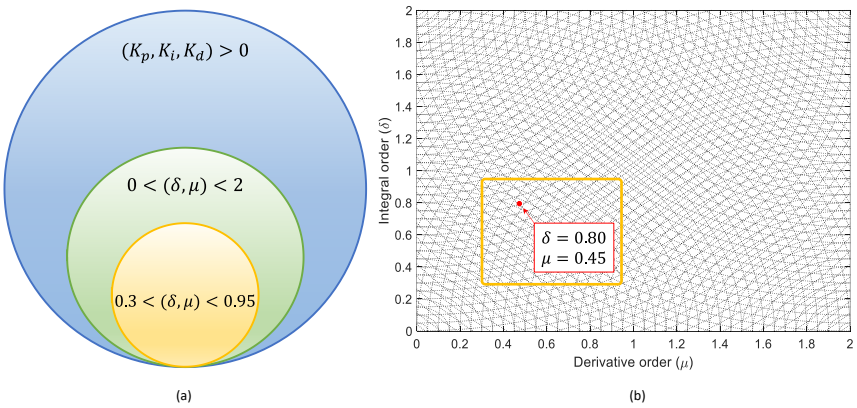

Figure 10: The stability regions for K_p, K_i, K_d, δ, and μ.

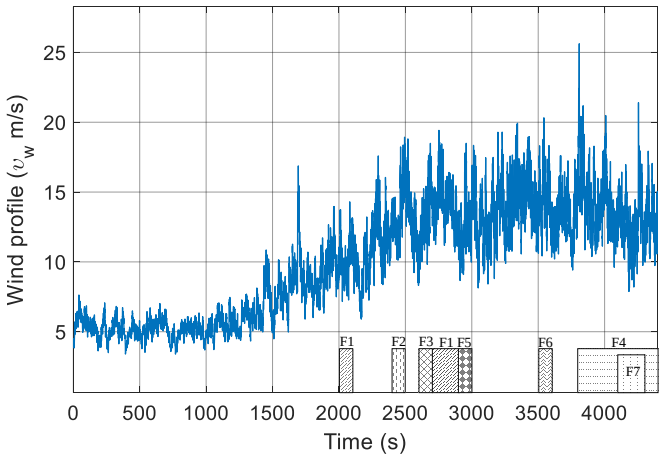

Figure 11: The wind speed profile and the occurrence time intervals of each fault scenarios.

the reference tracking of the generator. This tracking ability degradation can lead to large fluctuations in the generator speed. It is worth mentioning that these faults occur relatively slow; thus, the pitch angle can track the reference. However, they need to be compensated in order to prevent the deterioration of tracking performance.

Figure 12 depicts the power generated by the WT under the control of all three controllers under consideration. It can be seen that the profiles cover the full range of operation, demonstrating the suitability of this profile for comparison under the various fault scenarios. It is noteworthy to mention that, due to the stochastic wind behavior and its deviations, whenever the wind speed decreases largely in region III, the generated power also decreases largely. However, faults also impose the effects based on their severity and result in the decrement of generated power. According to Fig. 12, it can be seen that all three control approaches can compensate the effects of the fault, delivering different levels of performance. Here, the conventional PI controller's parameters are chosen as in [47]. From Fig. 12, it is evident that the effects of faults (F1-F3) with low severity are satisfactorily accommodated using all three controllers. However, when the highly severe fault (F5) happens due to hydraulic pressure drop, the conventional PI and optimal FOPID controllers could not deliver a satisfactorily fault accommodation performance as the proposed EM-OFOPID did. In the event of the air content increment in the oil with a medium level of severity (F6), the same performance is achieved with the conventional PI; however, in this case, the optimal FOPID has performed as well as the proposed EM-OFOPID, outperforming the conventional PI. It can also be seen that while the low sever fault (F4) is occurring in the time interval of 3805-4400 seconds, at some points, the conventional PI fails to accommodate the effects of the fault. Besides, a significant decrease in the wind speed happens within the interval of 4180-4260 seconds, which associates with the previously happened fault (F4) and another fault (F7) during 4100-4300 seconds. Accordingly, the generated power is decreased; however, from the zoomed-in inset, it is evident that the proposed EM-OFOPID demonstrates superior performance compared to other controllers in terms of fault accommodation and power generation. Figure 13 compares the performance of PI and EM-OFOPID controllers in terms of power generation when fault F7 occurs to the system. The fault corresponds to the friction changes in the drive-train with different levels of severity with a 5%, 10%, 50%, and 100% increase in the coefficient during the time interval of 4100-4300 seconds. Accordingly, it can be observed that increasing the severity of F7 degrades the performance of PI, while EM-OFOPID can effectively compensate for the fault and demonstrate superior performance. In addition, the comparative $\|P_g - P_{g,opt}\|_2$ in all controllers validate the superior performance of EM-OFOPID with 1.4591e+08 over OFOPID with 1.4923e+08 and PI with 1.6814e+08. It is readily observed that taking advantage of more design parameters and DWPFA to tune them; the FOPID schemes increase the power generation compared to the conventional PI scheme. The results also reveal that utilizing a memory of pitch angles with the optimal FOPID effectively

Table 8: The comparative $\|P_g - P_{g,opt}\|_2$ of EM-OFOPID and OFOPID controllers tuned by optimization algorithms.

OFOPID			EM-OFOPID		
Algorithm	$\|P_g - P_{g,opt}\|_2$	Rank	Algorithm	$\|P_g - P_{g,opt}\|_2$	Rank
DWPFA	1.4923e+08	1	DWPFA	1.4591e+08	1
FPSOMA	1.4940e+08	2	FOFA	1.4610e+08	2
FOFA	1.4961e+08	3	FPSOMA	1.4667e+08	3
CCGBFO	1.4996e+08	4	CCGBFO	1.4704e+08	4
EGWO	1.5058e+08	5	EGWO	1.4735e+08	5
FA	1.5094e+08	6	FA	1.4758e+08	6
GWO	1.5143e+08	7	BFO	1.4777e+08	7
BFO	1.5196e+08	8	GWO	1.4792e+08	8
PSO	1.5224e+08	9	PSO	1.4811e+08	9

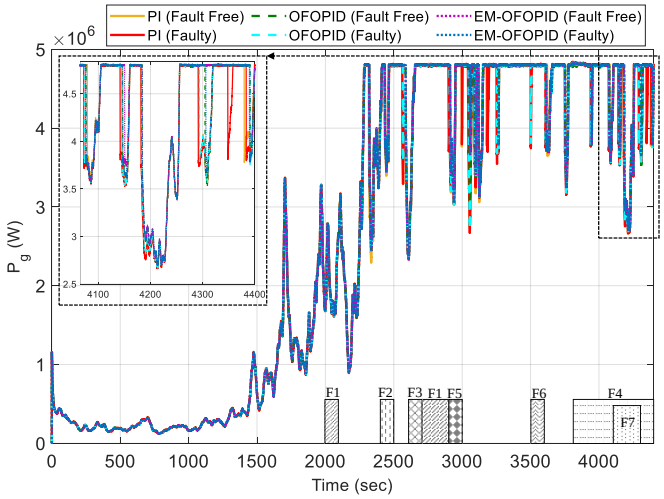

Figure 12: Generator power under fault-free and faulty conditions; a comparison between PI, OFOPID, and EM-OFOPID methods. The insets exhibit the dashed-line-highlighted regions.

enhances the control performance, while the superiority of DWPFA-optimized EM-FOPID is apparent in comparison with the DWPFA-optimized conventional FOPID. To further testify the performance of DWPFA compared to other EAs, the comparative $\|P_g - P_{g,opt}\|_2$ of EM-OFOPID and OFOPID controllers tuned by optimization algorithms is demonstrated and sorted by the performance order "Rank" in Table 8. The results manifest the superior performance of DWPFA over other algorithms, obtaining the best rank.

Figures 14 and 15 show the rotor and generator speed, respectively. As it is observed, despite the occurrence of different faults, the conventional PI and optimal FOPID approaches deliver sort of acceptable performance. In this

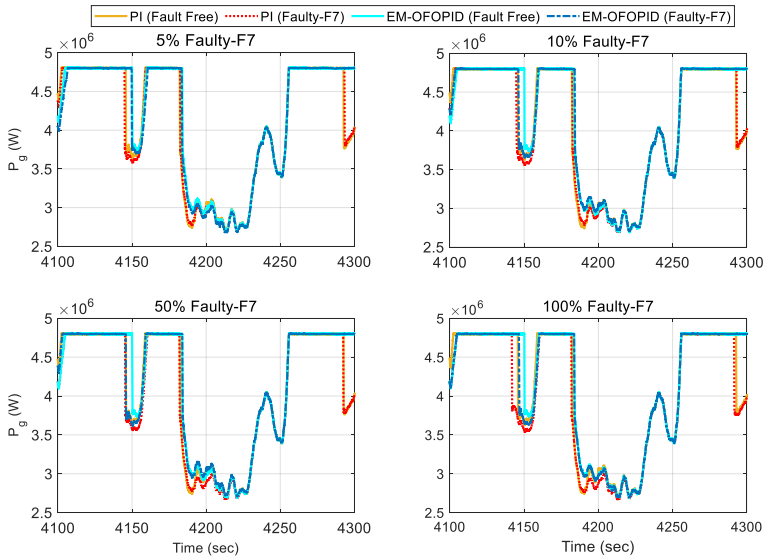

Figure 13: Generator power under occurrence of faults F7 during the time interval of 4100-4300 s; a comparison between PI and EM-OFOPID methods.

regard, as the zoomed-in insets show in Figs. 14 and 15, compared to the fault-free case, the performance of conventional PI degrades as the faults occur. The optimal FOPID also shows similar behavior; however, its performance is significantly better than the conventional PI. Considering Figs. 14 and 15 it is obvious that although each fault imposes its effects based on its severity level, the EM-OFOPID evidently demonstrates improved performance, that is to say, the proposed fault-tolerant EM-OFOPID controller with extended memory of pitch angles can work well even at the situation of simultaneous sensor, actuator, and system faults. Figure 16 shows the scatter plot of generated power, where it is observed that, in comparison to other methods, the proposed control scheme tends to be more consistent in the power generated at a given wind speed. Figures 17, 18, and 19 respectively depict the measured pitch angle of blade 1 from sensor 1, blade 2 from sensor 2, and blade 3 from sensor 2 in the presence of different fault scenarios. As it is observed, the operational constraints on pitch angle ($-3^\circ \leq \beta \leq 90^\circ$) are respected.

To sum up, according to the simulation results illustrated, the investigated conventional PI, DWPFA optimized FOPID, and DWPFA optimized EM-FOPID controllers, efficiently tolerate the effects of sensor, actuator, and system faults. However, as investigated, the EM-OFOPID demonstrated the best performance in mitigating the effects of fault scenarios and improving the power generation of the WT.

Remark 13. *Although non-PID approaches can often produce superior behaviour, but there is a strong industrial preference for PID controllers due to*

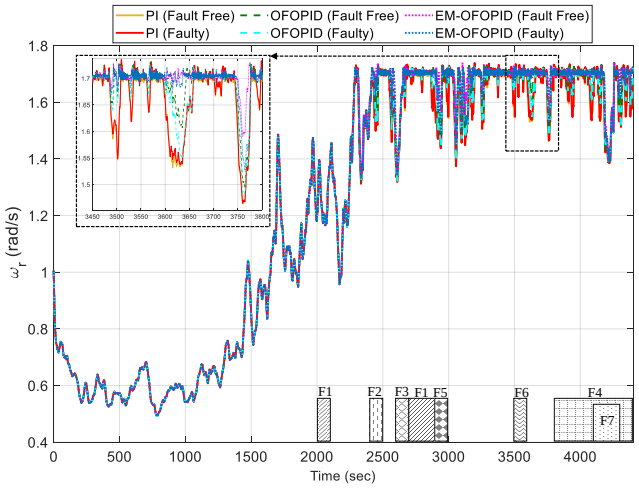

Figure 14: Rotor speed under fault-free and faulty conditions; a comparison between PI, OFOPID, and EM-OFOPID methods. The insets exhibit the dashed-line-highlighted regions.

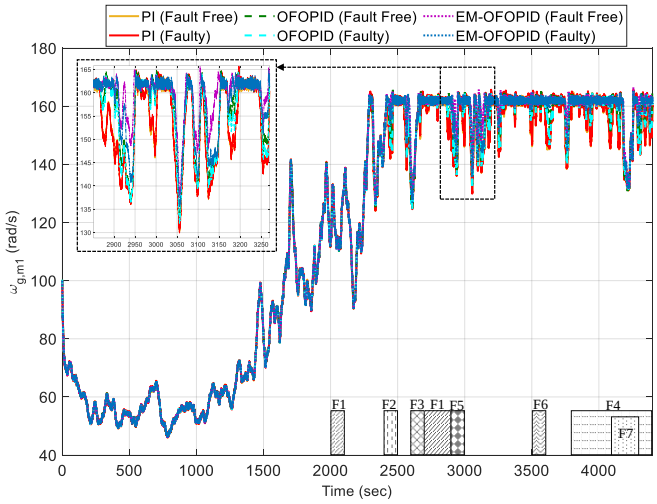

Figure 15: Generator speed under fault-free and faulty conditions; a comparison between PI, OFOPID, and EM-OFOPID methods. The insets exhibit the dashed-line-highlighted regions.

some main reasons such as: a) simplicity of design and implementation, so PID controller do not require overly complex mathematical models for the design process, b) they can be re-tuned in the field if necessary, by operators who can make small changes to improve performance without having to go back to re-do com-

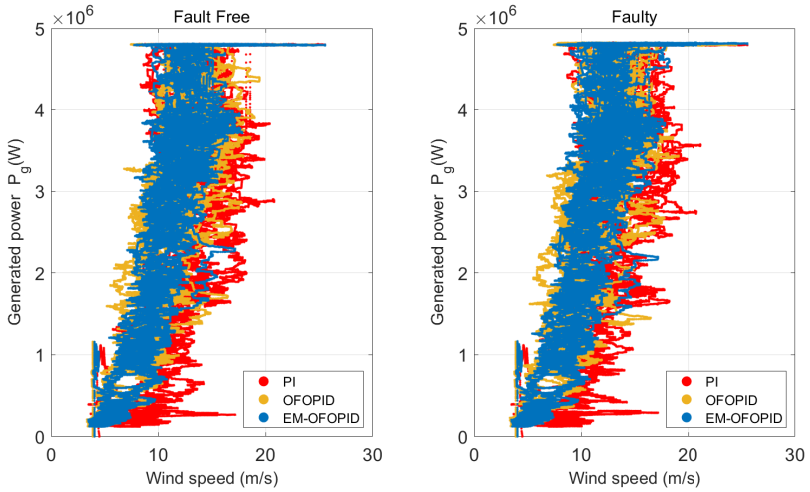

Figure 16: Scatter plot of generated power; a comparison between PI, OFOPID, and EM-OFOPID methods.

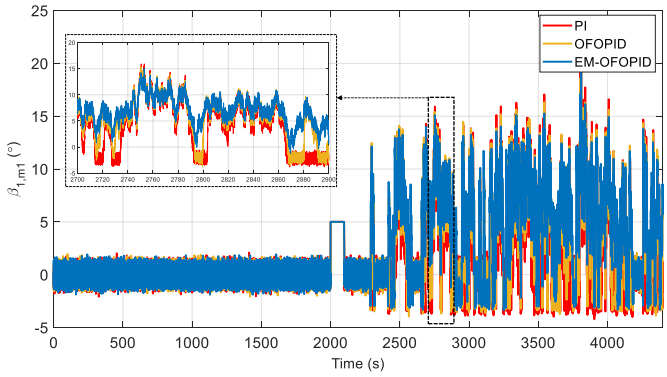

Figure 17: Pitch angle of blade 1 under faulty condition; a comparison between PI, OFOPID, and EM-OFOPID methods. The insets exhibit the dashed-line-highlighted regions.

plicated analysis, and c) considering the industry's current infrastructures and the hardship and costly efforts that have to be done to install the required hardware for other complex methods, the benefits of the proposed method in terms of no requirements for costly or complex hardware installments will be salient.

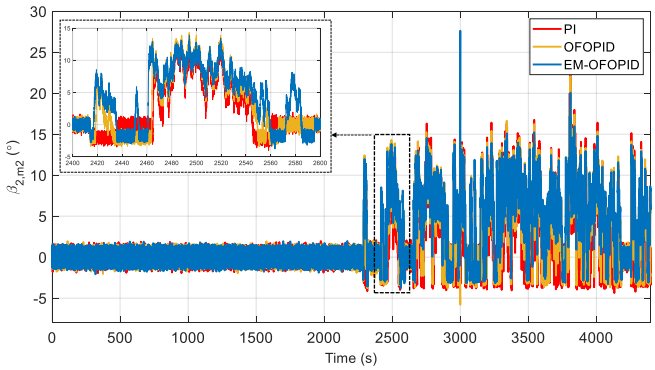

Figure 18: Pitch angle of blade 2 under faulty condition; a comparison between PI, OFOPID, and EM-OFOPID methods. The insets exhibit the dashed-line-highlighted regions.

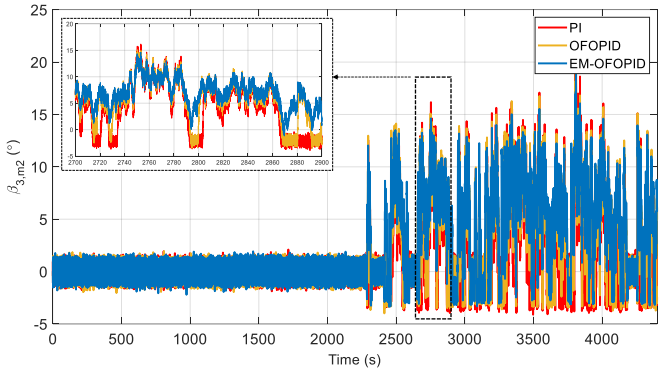

Figure 19: Pitch angle of blade 3 under faulty condition; a comparison between PI, OFOPID, and EM-OFOPID methods. The insets exhibit the dashed-line-highlighted regions.

6 Conclusions

This study proposed a new fault-tolerant pitch control scheme to adjust the pitch angle of WT blades subjected to sensor, actuator, and system faults. The proposed FTPC scheme comprises a fractional-calculus-based extended memory of pitch angles augmented with FOPID controller to maintain improvement in power generation performance of WT. Furthermore, a novel dynamic weighted parallel firefly algorithm (DWPFA) has been proposed, and its performance was evaluated through well-defined CEC2017 benchmark functions in comparison with other EAs. Non-parametric Friedman and Friedman Aligned statistical tests were utilized to analyze the quality of solutions. Comparative simulation

results revealed the superiority of DWPFA over other EAs. The performance of the proposed fault-tolerant EM-FOPID has been investigated in comparison to conventional PI and optimal FOPID approaches, on a 4.8-MW WT model in region III, where the controller parameters were tuned using DWPFA. Simulation results demonstrated the efficaciousness of the proposed FTPC strategy under fault-free and faulty conditions. Accordingly, the proposed DWPFA optimized EM-FOPID not only demonstrated the best performance in mitigating the effects of fault scenarios, but also improved the power generation of the WT.

The most significant limitation in this work is that all necessary signals, including immediate wind speed at appropriate resolution along with rotor speed, are assumed to be available. However, these signals have been found to be easily measured/estimated in practice by taking advantage of estimators. Also, a limitation for realization of the proposed EM-FOPID control scheme for the blade pitch control system is a slight increase in the computational complexity due to the memory requirements based on the fractional-order operators and the higher number of parameters that must be tuned, compared to the conventional controllers. Since approximations must be considered to implement such controllers, fractional-order operators' implementations are relatively complex and costly compared to their integer-order counterparts. However, for the specific controller presented here, it only takes 1.8% and 6.4% more time as compared to that of FOPID and PI controllers, respectively, to perform the blade pitch control. Besides, although the conventional PI and the optimal FOPID approaches provide lower computational complexities with respect to the proposed EM-FOPID, prioritizing more power generation and better fault-tolerant performances will make the proposed method a more preferred candidate providing a viable solution that can be implemented with ease without the needs of costly or complex hardware installments.

Acknowledgments

The authors gratefully acknowledge the support of the Glasgow Caledonian University (GCU) SMART Centre and the GCU Doctoral Scholarships.

References

[1] Abolvafaei, M., and Ganjefar, S. Maximum power extraction from wind energy system using homotopy singular perturbation and fast terminal sliding mode method. *Renewable Energy 148* (2020), 611–626.

[2] Aissaoui, A. G., Tahour, A., Essounbouli, N., Nollet, F., Abid, M., and Chergui, M. I. A fuzzy-PI control to extract an optimal power from wind turbine. *Energy Conversion and Management 65* (2013), 688–696.

[3] Altabeeb, A. M., Mohsen, A. M., and Ghallab, A. An improved hybrid firefly algorithm for capacitated vehicle routing problem. *Applied Soft Computing 84* (2019), 105728.

[4] Amoura, K., Mansouri, R., Bettayeb, M., and Al-Saggaf, U. M. Closed-loop step response for tuning PID-fractional-order-filter controllers. *ISA Transactions 64* (2016), 247–257.

[5] Angel, L., and Viola, J. Fractional order PID for tracking control of a parallel robotic manipulator type delta. *ISA Transactions 79* (2018), 172–188.

[6] Asgharnia, A., Jamali, A., Shahnazi, R., and Maheri, A. Load mitigation of a class of 5-MW wind turbine with RBF neural network based fractional-order PID controller. *ISA Transactions 96* (2020), 272–286.

[7] Asgharnia, A., Shahnazi, R., and Jamali, A. Performance and robustness of optimal fractional fuzzy PID controllers for pitch control of a wind turbine using chaotic optimization algorithms. *ISA Transactions 79* (2018), 27–44.

[8] Awad, N. H., Ali, M. Z., and Suganthan, P. N. Ensemble sinusoidal differential covariance matrix adaptation with euclidean neighborhood for solving CEC2017 benchmark problems. In *2017 IEEE Congress on Evolutionary Computation (CEC)* (2017), IEEE, pp. 372–379.

[9] Azarmi, R., Tavakoli-Kakhki, M., Sedigh, A. K., and Fatehi, A. Analytical design of fractional order PID controllers based on the fractional set-point weighted structure: Case study in twin rotor helicopter. *Mechatronics 31* (2015), 222–233.

[10] Azizi, A., Nourisola, H., and Shoja-Majidabad, S. Fault tolerant control of wind turbines with an adaptive output feedback sliding mode controller. *Renewable Energy 135* (2019), 55–65.

[11] Badihi, H., Zhang, Y., and Hong, H. Wind turbine fault diagnosis and fault-tolerant torque load control against actuator faults. *IEEE Transactions on Control Systems Technology 23*, 4 (2014), 1351–1372.

[12] Badihi, H., Zhang, Y., Pillay, P., and Rakheja, S. Fault-tolerant individual pitch control for load mitigation in wind turbines with actuator faults. *IEEE Transactions on Industrial Electronics 68*, 1 (2020), 532–543.

[13] Beltran, B., Ahmed-Ali, T., and Benbouzid, M. E. H. Sliding mode power control of variable-speed wind energy conversion systems. *IEEE Transactions on Energy Conversion 23*, 2 (2008), 551–558.

[14] Benamor, A., Benchouia, M., Srairi, K., and Benbouzid, M. A new rooted tree optimization algorithm for indirect power control of wind turbine based on a doubly-fed induction generator. *ISA Transactions 88* (2019), 296–306.

[15] Bianchi, F. D., Sánchez-Peña, R. S., and Guadayol, M. Gain scheduled control based on high fidelity local wind turbine models. *Renewable Energy 37*, 1 (2012), 233–240.

[16] Biswas, A., Das, S., Abraham, A., and Dasgupta, S. Design of fractional-order $PI^{\lambda}D^{\mu}$ controllers with an improved differential evolution. *Engineering Applications of Artificial Intelligence 22*, 2 (2009), 343–350.

[17] Bounar, N., Labdai, S., and Boulkroune, A. PSO–GSA based fuzzy sliding mode controller for DFIG-based wind turbine. *ISA Transactions 85* (2019), 177–188.

[18] Carrasco, J., García, S., Rueda, M., Das, S., and Herrera, F. Recent trends in the use of statistical tests for comparing swarm and evolutionary computing algorithms: Practical guidelines and a critical review. *Swarm and Evolutionary Computation 54* (2020), 100665.

[19] Chen, H., Zhang, Q., Luo, J., Xu, Y., and Zhang, X. An enhanced bacterial foraging optimization and its application for training kernel extreme learning machine. *Applied Soft Computing 86* (2020), 105884.

[20] Cho, S., Gao, Z., and Moan, T. Model-based fault detection, fault isolation and fault-tolerant control of a blade pitch system in floating wind turbines. *Renewable Energy 120* (2018), 306–321.

[21] Deraz, S., and Kader, F. A. A new control strategy for a stand-alone self-excited induction generator driven by a variable speed wind turbine. *Renewable Energy 51* (2013), 263–273.

[22] Dolan, D. S., and Lehn, P. W. Simulation model of wind turbine 3p torque oscillations due to wind shear and tower shadow. *IEEE Transactions on Energy Conversion 21*, 3 (2006), 717–724.

[23] Fister, I., Fister Jr, I., Yang, X.-S., and Brest, J. A comprehensive review of firefly algorithms. *Swarm and Evolutionary Computation 13* (2013), 34–46.

[24] Gui, L., Xia, X., Yu, F., Wu, H., Wu, R., Wei, B., Zhang, Y., Li, X., and He, G. A multi-role based differential evolution. *Swarm and Evolutionary Computation 50* (2019), 100508.

[25] Habibi, H., Howard, I., and Simani, S. Reliability improvement of wind turbine power generation using model-based fault detection and fault tolerant control: A review. *Renewable Energy 135* (2019), 877–896.

[26] Habibi, H., Nohooji, H. R., and Howard, I. Adaptive PID control of wind turbines for power regulation with unknown control direction and actuator faults. *IEEE Access 6* (2018), 37464–37479.

[27] Jain, T., and Yamé, J. J. Fault-tolerant economic model predictive control for wind turbines. *IEEE Transactions on Sustainable Energy 10*, 4 (2018), 1696–1704.

[28] Johnson, K. E., Pao, L. Y., Balas, M. J., and Fingersh, L. J. Control of variable-speed wind turbines: standard and adaptive techniques for maximizing energy capture. *IEEE Control Systems Magazine 26*, 3 (2006), 70–81.

[29] Kumar, R., Singh, R., Ashfaq, H., Singh, S. K., and Badoni, M. Power system stability enhancement by damping and control of subsynchronous torsional oscillations using whale optimization algorithm based type-2 wind turbines. *ISA Transactions 108* (2021), 240–256.

[30] Lan, J., Patton, R. J., and Zhu, X. Fault-tolerant wind turbine pitch control using adaptive sliding mode estimation. *Renewable Energy 116* (2018), 219–231.

[31] Lee, C.-H., and Chang, F.-K. Fractional-order PID controller optimization via improved electromagnetism-like algorithm. *Expert Systems with Applications 37*, 12 (2010), 8871–8878.

[32] Li, D.-Y., Li, P., Cai, W.-C., Song, Y.-D., and Chen, H.-J. Adaptive fault-tolerant control of wind turbines with guaranteed transient performance considering active power control of wind farms. *IEEE Transactions on Industrial Electronics 65*, 4 (2017), 3275–3285.

[33] Luo, C., Banakar, H., Shen, B., and Ooi, B.-T. Strategies to smooth wind power fluctuations of wind turbine generator. *IEEE Transactions on Energy Conversion 22*, 2 (2007), 341–349.

[34] Luo, K. Enhanced grey wolf optimizer with a model for dynamically estimating the location of the prey. *Applied Soft Computing 77* (2019), 225–235.

[35] Lv, L., and Zhao, J. The firefly algorithm with gaussian disturbance and local search. *Journal of Signal Processing Systems 90*, 8 (2018), 1123–1131.

[36] Ma, Z., Yan, Z., Shaltout, M. L., and Chen, D. Optimal real-time control of wind turbine during partial load operation. *IEEE Transactions on Control Systems Technology 23*, 6 (2015), 2216–2226.

[37] Madsen, F., Nielsen, T., Kim, T., Bredmose, H., Pegalajar-Jurado, A., Mikkelsen, R., Lomholt, A., Borg, M., Mirzaei, M., and Shin, P. Experimental analysis of the scaled DTU10MW TLP floating wind turbine with different control strategies. *Renewable Energy 155* (2020), 330–346.

[38] Mousavi, Y., and Alfi, A. A memetic algorithm applied to trajectory control by tuning of fractional order proportional-integral-derivative controllers. *Applied Soft Computing 36* (2015), 599–617.

[39] Mousavi, Y., and Alfi, A. Fractional calculus-based firefly algorithm applied to parameter estimation of chaotic systems. *Chaos, Solitons & Fractals 114* (2018), 202–215.

[40] Mousavi, Y., Alfi, A., and Kucukdemiral, I. B. Enhanced fractional chaotic whale optimization algorithm for parameter identification of isolated wind-diesel power systems. *IEEE Access 8* (2020), 140862–140875.

[41] Mousavi, Y., Bevan, G., Küçükdemiral, I. B., and Fekih, A. Maximum power extraction from wind turbines using a fault-tolerant fractional-order nonsingular terminal sliding mode controller. *Energies 14*, 18 (2021), 5887.

[42] Mousavi, Y., Zarei, A., and Jahromi, Z. S. Robust adaptive fractional-order nonsingular terminal sliding mode stabilization of three-axis gimbal platforms. *ISA Transactions* (2021).

[43] Muhando, E., Senjyu, T., Yona, A., Kinjo, H., and Funabashi, T. Disturbance rejection by dual pitch control and self-tuning regulator for wind turbine generator parametric uncertainty compensation. *IET Control Theory & Applications 1*, 5 (2007), 1431–1440.

[44] Naidu, R. P. K., and Meikandasivam, S. Power quality enhancement in a grid-connected hybrid system with coordinated PQ theory & fractional order PID controller in DPFC. *Sustainable Energy, Grids and Networks 21* (2020), 100317.

[45] Nayak, J., Naik, B., Dash, P. B., Souri, A., and Shanmuganathan, V. Hyper-parameter tuned light gradient boosting machine using memetic firefly algorithm for hand gesture recognition. *Applied Soft Computing 107* (2021), 107478.

[46] Niknam, T., Azizipanah-Abarghooee, R., and Roosta, A. Reserve constrained dynamic economic dispatch: A new fast self-adaptive modified firefly algorithm. *IEEE Systems Journal 6*, 4 (2012), 635–646.

[47] Odgaard, P. F., Stoustrup, J., and Kinnaert, M. Fault-tolerant control of wind turbines: A benchmark model. *IEEE Transactions on Control Systems Technology 21*, 4 (2013), 1168–1182.

[48] Pazhoohesh, F., Hasanvand, S., and Mousavi, Y. Optimal harmonic reduction approach for pwm AC–AC converter using nested memetic algorithm. *Soft Computing 21*, 10 (2017), 2761–2776.

[49] Ren, H.-P., Fan, J.-T., and Kaynak, O. Optimal design of a fractional-order proportional-integer-differential controller for a pneumatic position servo system. *IEEE Transactions on Industrial Electronics 66*, 8 (2018), 6220–6229.

[50] Ren, Y., Li, L., Brindley, J., and Jiang, L. Nonlinear PI control for variable pitch wind turbine. *Control Engineering Practice 50* (2016), 84–94.

[51] Ruiz, M., Mujica, L. E., Alferez, S., Acho, L., Tutiven, C., Vidal, Y., Rodellar, J., and Pozo, F. Wind turbine fault detection and classification by means of image texture analysis. *Mechanical Systems and Signal Processing 107* (2018), 149–167.

[52] Sabatier, J., Agrawal, O. P., and Machado, J. T. *Advances in fractional calculus*, vol. 4. Springer, 2007.

[53] Shan, J., Pan, J.-S., Chang, C.-K., Chu, S.-C., and Zheng, S.-G. A distributed parallel firefly algorithm with communication strategies and its application for the control of variable pitch wind turbine. *ISA Transactions* (2021).

[54] Tang, X., Yin, M., Shen, C., Xu, Y., Dong, Z. Y., and Zou, Y. Active power control of wind turbine generators via coordinated rotor speed and pitch angle regulation. *IEEE Transactions on Sustainable Energy 10*, 2 (2018), 822–832.

[55] Trachanatzi, D., Rigakis, M., Marinaki, M., and Marinakis, Y. A firefly algorithm for the environmental prize-collecting vehicle routing problem. *Swarm and Evolutionary Computation 57* (2020), 100712.

[56] Van, T. L., Nguyen, T. H., and Lee, D.-C. Advanced pitch angle control based on fuzzy logic for variable-speed wind turbine systems. *IEEE Transactions on Energy Conversion 30*, 2 (2015), 578–587.

[57] Venkaiah, P., and Sarkar, B. K. Hydraulically actuated horizontal axis wind turbine pitch control by model free adaptive controller. *Renewable Energy 147* (2020), 55–68.

[58] Viola, J., Angel, L., and Sebastian, J. M. Design and robust performance evaluation of a fractional order PID controller applied to a DC motor. *IEEE/CAA Journal of Automatica Sinica 4*, 2 (2017), 304–314.

[59] Wang, C.-F., and Song, W.-X. A novel firefly algorithm based on gender difference and its convergence. *Applied Soft Computing 80* (2019), 107–124.

[60] Wang, H., Wang, W., Zhou, X., Sun, H., Zhao, J., Yu, X., and Cui, Z. Firefly algorithm with neighborhood attraction. *Information Sciences 382* (2017), 374–387.

[61] Xiong, L., Li, P., Ma, M., Wang, Z., and Wang, J. Output power quality enhancement of pmsg with fractional order sliding mode control. *International Journal of Electrical Power & Energy Systems 115* (2020), 105402.

[62] Xu, N., Chen, Y., Xue, A., Zong, G., and Zhao, X. Event-trigger-based adaptive fuzzy hierarchical sliding mode control of uncertain under-actuated switched nonlinear systems. *ISA Transactions* (2019).

[63] Yang, X.-S. Firefly algorithms for multimodal optimization. In *International Symposium on Stochastic Algorithms* (2009), Springer, pp. 169–178.

[64] Zhang, Y., Cheng, M., and Chen, Z. Load mitigation of unbalanced wind turbines using PI-R individual pitch control. *IET Renewable Power Generation 9*, 3 (2014), 262–271.