A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF ÇANKIRI KARATEKİN UNIVERSITY

CONTROL OF INDIVIDUAL PITCH ANGLE IN VARIABLE WIND SPEED USING MODERN OPTIMIZATION TECHNIQUES

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
ELECTRICAL AND ELECTRONICS ENGINEERING

BY

MAHDI OSMAN ABDI

ÇANKIRI

2022

CONTROL OF INDIVIDUAL PITCH ANGLE IN VARIABLE WIND SPEED USING MODERN OPTIMIZATION TECHNIQUES

By Mahdi OSMAN ABDI

August 2022

We certify that we have read this thesis and that in our opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science

Advisor **:** Asst. Prof. Dr. Göksu GÖREL

Examining Committee Members:

Chairman **:** Assoc. Prof. Dr. Fatih KORKMAZ
Electrical and Electronics Engineering
Çankırı Karatekin University

Member **:** Asst. Prof. Dr. Göksu GÖREL
Electrical and Electronics Engineering
Çankırı Karatekin University

Member **:** Asst. Prof. Dr. Mehmet GÜÇYETMEZ
Electrical and Electronics Engineering
Sivas University of Science and Technology

Approved for the Graduate School of Natural and Applied Sciences

Prof. Dr. İbrahim ÇİFTÇİ
Director of Graduate School

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Mahdi OSMAN ABDI

ABSTRACT

CONTROL OF INDIVIDUAL PITCH ANGLE IN VARIABLE WIND SPEED USING MODERN OPTIMIZATION TECHNIQUES

Mahdi OSMAN ABDI

Master of Science in Electrical and Electronics Engineering

Advisor: Asst. Prof. Dr. Göksu GÖREL

August 2022

The renewable energy source used in many different applications is wind power. Wind energy is one of the most expensive forms of energy today. The purpose of wind power is to convert kinetic energy into electrical energy. MATLAB/SIMULINK simulations allowed us to clearly distinguish the three control methods used to optimise the FAST wind turbine; the N-Z PI method, the PSO-PI method and the GA-PI method produced similar results. Increasing the rotor speed quality of the wind turbines and decreasing the mechanical loads on the turbines were investigated. By adjusting the blade angle to the nominal wind speed, the rotor speed of the wind turbine is maintained at its nominal value. By using control methods (such as PI, genetic algorithms and particle swarm optimisation), different results can be achieved. In addition, the individual control of the blade tilt angle allowed us to reduce the mechanical loads of the wind turbine through control methods. The wind turbine was modelled in Matlab/Simulink. The simulation results show that the individual control of the blade tilt angle ensures the quality of the rotor speed of the wind turbine and reduces the balanced periodic loads on the wind turbine.

2022, 74 pages

Keywords: Fast, Genetic algorithm, Pitch angle, Particle swarm optimization

ÖZET

MODERN OPTİMİZASYON TEKNİKLERİ İLE DEĞİŞKEN RÜZGÂR HIZINDA BİREYSEL HATVE AÇISININ KONTROLÜ

Mahdi OSMAN ABDI

Elektrik ve Elektronik Mühendisliği, Yüksek Lisans

Tez Danışmanı: Dr. Öğr. Üyesi Göksu GÖREL

Ağustos 2022

Birçok farklı uygulamada kullanılan yenilenebilir enerji kaynağı rüzgâr enerjisidir. Rüzgâr enerjisi günümüzde en pahalı enerji türlerinden biridir. Rüzgâr gücünün amacı kinetik enerjiyi elektrik enerjisine dönüştürmektir. MATLAB/SIMULINK simülasyonları, FAST rüzgâr türbinini optimize etmek için kullanılan üç kontrol yöntemini net bir şekilde ayırt etmemizi sağladı; N-Z PI yöntemi, PSO-PI yöntemi ve GA-PI yöntemi benzer sonuçlar üretti. Rüzgâr türbinlerinin rotor hızı kalitesinin artırılması ve türbinler üzerindeki mekanik yüklerin azaltılması araştırılmıştır. Kanat açısını nominal rüzgâr hızına göre ayarlayarak rüzgâr türbininin rotor hızının nominal değerinde tutulması sağlanmıştır. Kontrol yöntemleri (PI, genetik algoritmalar ve parçacık sürüsü optimizasyonu gibi) kullanılarak farklı sonuçlar elde edilebilir. Buna ek olarak, kanat eğim açısının bireysel kontrolü, kontrol yöntemleri aracılığıyla rüzgar türbininin mekanik yüklerini azaltmamızı sağladı. Rüzgâr türbini Matlab/Simulink'te modellenmiştir. Simülasyon sonuçları, kanat eğim açısının bireysel kontrolünün rüzgâr türbininin rotor hızının kalitesini sağladığını ve rüzgâr türbini üzerindeki dengeli periyodik yükleri azalttığını göstermektedir.

2022, 74 sayfa

Anahtar Kelimeler: Fast, Genetik algoritma, Hatve Açısı, Parçacık sürüsü optimizasyonu.

PREFACE AND ACKNOWLEDGEMENTS

I would like to express my deep gratitude to my thesis advisor, Asst. Prof. Dr. Göksu GÖREL, who helped me throughout my period by supporting me morally and academically. Secondly, I would like to thank my family for their financial and emotional support while I was at university. A major asset for my further academic career was my former professor, Asst. Prof. Dr. Zafer Civelek, who allowed me to expand my knowledge of the FAST system.

Mahdi OSMAN ABDI
Çankırı-2022

CONTENTS

LIST OF SYMBOLS

ρ	Air density
β	Blade pitch angle
Ω	Mechanical speed at the rotor shaft of the wind turbine
C_{aero}	Couple aerodynamic
P_{aero}	Power aerodynamic

LIST OF ABBREVIATIONS

A	Area swept by the rotor blades
Cp	Power coefficient
FAST	Fatigue, aerodynamic, structure, and turbulence
GA	Genetic algorithm
HAWT	Horizontal axis wind turbine
IPC	Individual pitch control
J	Total inertie
NREL	National renewable energy laboratory
P	Mechanical power in the moving air
PSO	Particule swarm optimization
R	Radios of the blade
TSR	Tip speed ratio
V	Velocity of the air
VAWT	Vertical axis wind turbine
WT	Wind turbine

LIST OF FIGURES

LIST OF TABLES

1. INTRODUCTION

1.1 Introduction

A wind turbine is a device that converts the kinetic energy of the wind into mechanical energy, known as wind power, which is then most often converted into electrical energy(Manyonge *et al.* 2012). Wind turbines that produce electricity are called wind generators, while wind turbines that pump water directly are sometimes called pumping turbines or wind pumps. An old form of wind turbine is the windmill(Omer 2008). The terms "wind power plant", "wind farm" or "wind farm" are used to describe grouped production units, installed on land or at sea(Piasecka *et al.* 2019).

Wind energy existed at the time of the windmill, which appeared in the 9th century. The wind turbine has been improved over the centuries to fit into the industrial area(Hau 2013). The very first 'industrial' wind turbine to generate electricity was developed by the Dane Paul La Cour in 1890, to produce hydrogen by electrolysis. Later, in the years following, two types of wind turbines were developed: horizontal and vertical. A horizontal wind turbine is a propeller perpendicular to the wind, mounted on a mast. The height is usually 20m for small wind turbines, and for large ones double the height of the blades. The horizontal wind turbine is the most common, consisting of a mast, a nacelle and a rotor. Such a wind turbine is usually designed based on its aerodynamic, mechanical and electrical specifications. The mast allows the rotor to be placed at a satisfactory height to allow its movement (utilitarian for horizontal axis wind turbines), or at a height of the wind that will be sufficient and constant than at ground level(Sharpe 2010). The nacelle is placed at the top of the mast and houses the mechanical, pneumatic, electrical and electronic components important to the operation of the machine. The rotor is composed of the front side of the wind turbine receiving the blades, fixed on a shaft rotating in bearings installed in the nacelle. A vertical wind turbine in contrast to the classical horizontal axis wind turbine, which has an axis perpendicular to the wind direction(Hau 2013). The axis is mostly vertical, but some of these types are horizontal and can be described in several ways (such as Darrieus, Savonius, rotary wing...).

The FAST (Fatigue, Aerodynamic, Structure, and Turbulence) code is modular and can be adapted to most existing wind turbines(Beltran *et al.* 2008). This flexibility comes from its multi-body structure which allows only the necessary components to be included via description files. The code is created by NREL (National Renewable Energy Laboratory), The FAST code can simulate the dynamic response of standard horizontal axis wind turbines with two or three blades(Tran et al.2015). Rotor-furling, tailfurling, and tail aerodynamics elements are optional in the wind turbine configuration and are useful in the analysis of most small wind turbines. Germanischer Lloyd WindEnergie assessed the code and deemed it adequate for "the calculation of onshore wind turbine loads for design and certification." (Jonkman and Buhl Jr 2005).

Topics covered in this thesis; wind turbines that produce electrical power from wind energy and in particular wind turbines with variable pitch angle and variable generator rotor speed. In addition, the generators used in these wind turbines and the electric actuators used to change the pitch angle are also considered in this thesis. Because the operating modes and transfer functions of these actuators are part of the control process, they need to be investigated and developed. In addition, the classical PID control methods, modern control methods, the PID-GA (Genetic Algorithm) control method and the PID-PSO (Particle Swoar optimization) control method, which is obtained by combining the two control methods, fall within the scope of this thesis.

1.2 Literature Review

This part will show a last check of the text detailing the pitch angle control (PAC) based wind turbine frameworks. Previous studies and research on demonstration, control approaches, and the optimal inverter topologies, in particular. In wind turbines, control techniques and optimal angle optimization are important.

To reduce aeroelastic flutter instability, this work offers a non-linear model predictive controller (NMPC) for pitch control of Horizontal-Axis Wind Turbines (HAWT) in Region 3. The pre-twist angle and pitch angle of each blade were included in an aeroelastic HAWT (Horizontal-Axis Wind Turbine) model. The response of the created model's time simulation was comparable to that of the turbine simulation utilizing the HAWT aero-elastic FAST simulation. When there is a lot of wind and the turbine rotor is spinning too fast. The created model's time simulation response was in good accord with that obtained from a turbine simulation utilizing the HAWT aero-elastic FAST simulator. Following that, an NMPC pitch controller was created using the above-mentioned aeroelastic rotor model for prediction purposes. However, the simple gain PI pitch controller and an NMPC pitch controller that uses the simplified transmission prediction model failed to do so due to a considerable rise in coupled deviations, implying that flutter instability exists. (El-Baklish *et al*. 2020).

The goal of this study is to compare the performance of several linear quadratic Gaussian (LQG) controllers in terms of decreasing the fatigue load produced by spatial wind speed turbulence on the most expensive wind turbine (WT) components. Five LQGs are created, each with a bigger number of objectives and a more complex control model, the first four with collective pitch control (CPC) and the fifth with individual pitch control (IPC). With a CPU in the loop, these five controllers are put to the test (PIL). Their control model takes into account the WT structure's various degrees of complexity, and the control objectives include amplitude reduction. Powertrain load reduction is clearly influenced by taking into account powertrain torsion and integrating the relevant torsional torque in the minimisation criterion. (Camblong *et al*. 2012).

The goal of pitch control is to maintain the best blade angle for a given power output. A traditional wind turbine controller, such as a PID controller, will provide a consistent nominal WT power output despite variable wind. Different types of PIDs are utilized in this study to evaluate which is most efficient. The suggested intelligent pitch angle controller for wind turbine blades produces results that can stabilize wind speed fluctuations. The controller must be tuned for this purpose, as well as the wind turbine's performance. (Sarkar *et al*. 2020).

When the wind power system varies, this study exhibits the performance of a PMSG-based industrial turbine with variable speed. The PMSG wind turbine's planned design can be expanded to include a continual grid connection. When the wind energy system is connected to the RL load via a DC link capacitor and thyristor converter control, the performance of a PMSG-based variable speed wind turbine is shown in this study. It keeps the control voltage practically constant at 310 V for constant and variable wind speed, which is extremely close to the target voltage on the RL load. Furthermore, the output is smooth and has the fewest oscillations, both of which are critical for grid compatibility. As a result, this setup allows for wind turbine generator management as well as DC bus voltage stabilization. (Hemeida, Farag and Mahgoub 2011).

These control algorithms boost blade pitch actuation, primarily to lower the 1P (once per revolution) aerodynamic load component. However, it is well recognized that blade pitch system failure is a primary cause of turbine downtime. Pitch actuation will only be exacerbated by control algorithms that increase pitch actuation. As a result, increasing blade pitch actuation to minimize mechanical stresses may not always be the best option. Individual pitch control allows for reduced pitch actuation while following the rotor speed rate. IPC (individual pitch control) can regulate rotor speed at greater speeds with less pitch actuation. In addition, the IPC can limit forward and backward movement as well as rotation. In comparison to the basic controller, the suggested IPC can maintain the nominal rotor speed with less pitch actuation. At increasing wind speeds, performance improves. Reduced blade pitch actuation is proposed to improve rotor speed control. In addition, out-of-plane blade displacement, tower fore-aft displacement, and platform pitch rotation can all be reduced. (Sarkar, Fitzgerald and Basu 2020).This research examines a simple wind turbine model with a non-linear receding horizon management system in depth. For the development of the system, the comparison work on wind turbine dynamics is based on the NREL 5MW reference wind turbine model, where the results reveal that a non-linear real-time receding horizon control on the wind turbine regulates the dynamics (in real time). The importance of real-time dynamic control of wind turbines is to ensure that the energy of the wind turbine is stabilized without losing too much energy. This illustrates the significance of selecting a specialized controller that allows for dynamic control. This paper gives a detailed examination of a small wind turbine model that was validated before being utilized to create a non-linear receding horizon control system. The comparison work on wind turbine dynamics is based on the NREL 5 MW reference wind turbine model for system development, where the results show the capacity and usefulness of a real-time non-linear receding-horizon control methodology on wind turbine control dynamics (in real-time). (Galvani, Sun and Turkoglu 2016).

The aim of this study is to create two control strategies for a variable pitch wind turbine system and see the difference in performance in real operation. There are two linear and non-linear models, implemented in SIMULINK with the wind turbine calculation code FAST. The control strategies are developed from a linear model and a non-linear model both made in SIMULINK with the wind turbine calculation code FAST. They are then implemented in the control module of the wind turbine and operating data are collected for about 1 month for each. The first control strategy is a proportional integral (PI) control strategy. It is chosen because of its simplicity of design and application and its proven good performance. To achieve good angular velocity control, gain scheduling had to be used. The operating data of the PI strategy is used to validate the behaviour of the non-linear model. The second strategy is a disturbance accommodation strategy (DAC). It is chosen because of its greater flexibility of adjustment and its simplicity of application to wind turbines. The objective of this thesis was to design two control strategies for the WESNet variable wind turbine stall system and compare their performance under real operating conditions to determine which one provides the best output. The first control strategy to be designed is a proportional-integral (PI) type. It is chosen because of its simplicity of design and application and its proven good performance. The second is a disturbance accommodation strategy (DAC) chosen because of the high flexibility of adjustment of its behaviour and its simplicity of application compared to other modern control strategies. Both strategies were in operation for about 1 month each. In the end, the DAC strategy proved to be more suitable for production than the PI strategy. The production plateau visible in region 3 on its power curve is more constant and higher: the nominal production value of 10 kW was only reached on the curve of the DAC strategy. Moreover, it has a higher annual energy production than the PI strategy, despite a lower availability caused by a high number of overspeed outages. (Dubé 2014).

The increase in installed wind power capacity around the world, as well as academics' interest in integrating it into the energy grid, has prompted this study. However, the randomness and unpredictability of its main source (wind), the grid's limited capacity, the possibility of accidental disconnections, and the decline of power quality all have an impact on this integration. The turbine's speed must be changed in response to the wind speed in order to maximize converted power. Pitch Control: This is a control method that involves adjusting the pitch angle of the blades around the longitudinal axis in response to wind speed and active power extraction. The power coefficient Cp no longer varies as a function of "lambda" as it did previously, but as a function of wind speed at constant rotation speed, allowing the power to be held constant in the zone 3 of its maximum value. The wind turbine's blade pitch angle can be changed to allow for a variety of different functions. (Sylla 2013).

This method is achievable through individual control of the pitch angle of the wind turbine in order to lower the load of the wind turbine in order to anticipate the loads to be reduced. A more advanced version of the previously stated individual height control approach has been created. Further large load reductions are accomplished, notably for fixed frame loads, by adding an unique anticipation term in parallel with the PI controllers for the d- and q-axes, at the expense of a further rise in pitch activity. (Bossanyi 2005).

In this thesis, a study is conducted on increasing the electrical power output and torque load on the turbine by controlling the individual pitch angle with different types of PID controllers. This was done in the MATLAB/SIMULINK environment. The FAST model was used as a wind turbine design, as it allows both collective and individual control of the wind turbine blade opening. The study conducted to control the power output of the wind turbine and its mechanical load reduction. This was done using both classical and modern control methods were carried out on the model wind turbine. The different results were compared to see the power output and mechanical load with the studies done on individual pitch and collective control. This shows the potential results from the individual pitch control side with the classical control methods. With modern control methods the stability of the output power and the mechanical load have decreased. It uses the error and its derivative for input, the change of the inclination angle for output. Then an algorithm comes to optimize the controller, an intelligent genetic algorithm (IGA). By comparing the results with and without the IGA method, the stability of the output power and the reduction of the mechanical loads brought by the optimisation to the system control. The target in controlling the pitch angle of the wind turbine blades; the power output is stable. At wind speeds above the rated wind speed, it is a matter of trying to keep the power output as stable as possible. However, as the size of wind turbines increases, studies have shown that mechanical loads should also be taken into account when controlling the power output. Individual control of the tilt angle is recommended, which provides a solution for some of the mechanical loads. Excess ratio improves torque, but worsens power output control. A low ratio does not provide good mechanical load reduction. Good control is achieved by ensuring a balance between these two dimensions. Good power quality is ensured by the stable output power, and the useful life of the wind turbine is increased by reducing the moments. Thus, the energy cost also decreases (Civelek 2017).

The major goal was to create a model that accounted for the most relevant dynamics affecting the wind turbine and the hydraulic transmission system in question, so that the model could be used to assess the dynamic feasibility of a hydraulic transmission concept. MATLAB Simulink is used to create the nonlinear dynamic model. The eigenperiods of a linearised model may be calculated analytically, and the simulations of the whole system are accurate. A valve control system is proposed to reduce pressure and power changes when the wind turbine is running below and above its rated wind speed. As a result, during operating over the rated wind speed, a blade pitch control system based on an aerodynamic power estimator is proposed. The system's dynamic response is stable in both cases below and beyond the nominal wind speed, and the turbine variables are within the typical ranges of conventional variable speed wind turbines with mechanical transmission, according to simulations. It is proposed that a valve control technique be used to offer extra dampening of the crucial oscillation period during operation below the rated wind speed and to improve power control during operation above the rated wind speed. For steady functioning of the entire system above the nominal wind speed, a new blade pitch control approach based on an aerodynamic power estimate is proposed. (Skaare, Hörnsten and Nielsen 2013).

The gains of a proportional-integral pitch angle controller for a 5 MW wind turbine may be calculated using two techniques, according to this study. The first method is analytical, whereas the second is simulation-based. When the wind speed falls below a nominal value, the speed controller can modify the rotor speed constantly to keep it at a level that gives the highest power coefficient, increasing the turbine efficiency. The power output can be affected by minor tilt angle adjustments. The control of the blade tilt angle in a variable speed wind turbine has been presented using a PI-based control creation system. The PI increases were calculated using both an analytical and a simulation-based method. The simulation results revealed that both proposed PI systems performed well. (Hwas and Katebi 2012).

An adaptive disturbance rejection control theory is used in this research to construct an individual blade pitch controller for a large-scale wind turbine to reduce load. The adaptive disturbance rejection control's goal is to modulate blade pitch angle independently in order to lessen asymmetric load in the blade caused by vertical wind shear while also rejecting unneeded disturbances. An individual blade pitch controller for load reduction is designed using a direct adaptive control approach. The fundamental goal of adaptive control was to build a controller that was as efficient as the present basic controller with the least amount of information about the wind turbine parameters. The suggested adaptive controller is simulated using a 5 MW nonlinear, high-fidelity wind turbine model from NREL. Using both pitch and turbulent wind profiles, the simulation results were compared to the existing reference single blade pitch controller. The adaptive controller is successful in reducing blade load, however this results in higher pitch actuation, according to simulation data. (Magar, Balas and Frost 2015).

The potential for offshore wind energy is large and to harvest this energy, turbines need to be placed further offshore. A floating wind turbine has additional motions that affect energy generation and loads in the structure. Thus, the control system of the wind turbine must be capable of improving energy generation and keeping the wind turbine operating safely under incidental wind and wave conditions. A PI controller with programmed gain is applied as a reference to evaluate the behavior of newly developed controllers. A collective linear and quadratic pitch controller, intended to regulate solely rotor speed and blade pitch, adjusts the performances of the system, but it is believed that this improvement is a result of improved controller tuning because both controllers utilize the same system to reset the blade pitch and control the speed. Individual blade pitch control utilising periodic control model is applied as it uses a distinct system to adjust the blade pitch of the platform. A collective blade pitch control utilizing a linear quadratic regulator (LQR) provides improved behavior in terms of rotor speed control as well as blade pitch control over a PI controller with programmed gain intended for rotor speed control solely. For collective pitch controllers, the blade pitch restoration device has been designed to provide a better control of both rotor speed and platform pitch than a programmed gain PI controller designed for rotor speed control only. This performance improvement is due to the use of constant torque control instead of constant power control for speed control, and possibly to better tuning of the controller. For collective pitch controllers, the blade pitch restoration mechanism is accomplished by switching the thrust uniformity on the rotor. In the case of an individual blade pitch controller, though, the mechanics used are a combined modification of symmetrical lift and induction of an asymmetrical rotor load, producing a pitch restoration moment. The initial simulations run using a low-fidelity model suggest that the individual blade pitch controller is capable of achieving even better performance than the collective pitch LQR controller. (Namik, Stol and Jonkman 2008).

As wind turbines get bigger, advanced control systems that lower wind turbine loads become more appealing, with the potential to significantly cut energy prices. Individual pitch control (IPC) and individual blade control (IBC) are two of the greatest choices for achieving this among the different advanced control systems. Their effectiveness should be measured not just in terms of load reduction, but also in terms of pitch system utilization. Even if large load reductions are achieved, an excessive rise in pitch activity is not acceptable. The wind turbine loads are regulated using the Coleman transformation based IPC controller and the loop shaping based IBC controller. The load decrease obtained alone does not provide enough information to evaluate the controllers' performance. The association between load reduction and pitch activity gives useful information to support the controller design for a specific design criterion. (Han and Leithead 2015).

This paper illustrates sensor selection and state analysis methods for vibration monitoring in wind turbine constructions, as well as various options for individual blade pitch control. Individual blade pitch control can reduce blade root torques under some conditions, according to simulations of the NREL Offshore Baseline Turbine (5MW), but the dynamics of the pitch actuators and assumptions in the modeling can restrict the control's performance under other scenarios. Because aeroelastic models are becoming more mature, it is becoming easier to include them into wind turbine automation. State variable controllers perform worse than a simple technique for regulating an estimated angle of attack. Individual blade pitch control is limited in both circumstances by the pitch actuators' speed restrictions. (Ehlers 2009).

The wind energy industry has focused on reducing the discounted average cost of energy through rotor scaling. Therefore, the usefulness of load reduction technology incorporated in the design methodology or for retrofit purposes becomes relevant. One of these is individual blade pitch control, a recurrent theme in the study, with known benefits and disadvantages, specifically the pitch actuator and bearing wear. The linear system is based on black-box identification of non-linear models and filtering is used for the input and output. Various configurations of the blade-independent control scheme are implemented on a 10 MW baseline turbine, with a very flexibly large rotor representing the present industry state, in wind situations as specified by the respective certification standard. Life extension, IPC as an upgrade technique might be a catalyst as no new sensor or actuators are required. This study presents a new IPC concept and examines the potentials and disadvantages of load reduction in a full design load base. Three independently operating SISO controllers are introduced using the blade root moment measured in the flap direction as input to an IBC design. The main controller is decoupled with high-pass filters and black-box identification designs are applied to tune the basic feedback structure. The results show that a greater reduction is achieved on the blade root moments in the flap direction, the torsional and lateral moments of the lower turn, and the non-rotating pitch and yaw loads of the upper turn. A control bandwidth greater than 3P benefits only the top of the tower and, to a lesser extent, the tower channels, but comes with a very high price tag on the pitch actuation (Pettas *et al.* 2018).

In this study, a new adaptive control approach for pitch control of wind turbines is developed. When operating under high wind speeds and internal structural uncertainties, wind turbines might suffer from reduced life due to excessive stresses and fatigue. When there are uncertainties in blade stiffness, the adaptive controller is designed to both manage generator speed and alleviate component loads in a turbulent wind field. The suggested algorithm is being evaluated on the NREL offshore reference wind turbine, which has a capacity of 5 MW. The control's performance is compared to those of baselines such as gain-proportional integral control (GSPI) and disturbance adaption control (DAC). In comparison to baseline controllers, the performance of blade load mitigation is better under unknown blade stiffness reduction. Adaptive control outperforms the DAC when it comes to generator speed control. GSPI will be sensitive to model uncertainty due to the possibility of structural uncertainties such as the loss of wind turbine blade rigidity. Adaptive control can increase load reduction on GSPI and speed regulation on DAC, according to the findings. (Yuan and Tang 2017).

The purpose of the research reported in this paper is to look into the aerodynamic properties and performance of the NREL Phase II rotor, which is a horizontal axis wind turbine rotor. Four turbulence models were utilized and assessed in this study: Spalart-Allmaras (S-A), Standard, RNG, and SST. The turbulence modeling literature considers these models to be the workhorses. Fluent predicted the aerodynamic characteristics of a HAWT model. The pressure coefficient and torque of the horizontal axis wind turbine were predicted using CFD simulations of the HAWT blade and the evaluation of four turbulence models for two wind speeds. The Spalart-Allmaras model outperformed the standard turbulence model at lower wind speeds, while the standard turbulence model outperformed the experimental data at higher wind speeds. Surprisingly, the RNG and

SST models' outputs were insufficient to reliably forecast HAWT torque. (Sebit, Alpman and Yılmaz 2014).

A strong wind turbine may be chosen because some of its properties may be advantageous for siting near inhabited areas. On the other hand, some aerodynamic difficulties emerge. This type of wind turbine operates at low speeds where dynamic stall plays a very significant role. The objective of this thesis work is to complete the knowledge of the dynamic stall phenomenon on a vertical axis wind turbine in order to improve the existing numerical prediction models. This study is based on a combined analysis of numerical and experimental results. The effects of viscosity are introduced by corrections using a semi-empirical dynamic stall model. The experimental work focused on the vortex dynamics in the immediate vicinity of the rotor resulting from the dynamic stall. The dynamics of the vortex above the airfoil was first examined. It was shown that the assumption of vortex convection velocity proportional to the relative Ur velocity along the profile appears to be a good approximation of what is observed from a position above 20% of the chord. It is shown here that this additional delay does indeed allow the time of vortex onset to be calibrated and to reconstruct an evolution of the vortex dynamics comparable to the experiment. This has allowed us to establish that the intensity of the vortex-induced depression is maximal when it is located around 20% chord and that this intensity decreases the more the vortex appears at a late azimuthal position. This study illustrates that the Leishman-Beddoes model is well suited to reproduce the evolution of the leading edge vortex but still needs some corrections or adaptations to be used with confidence. (Beaudet 2014).

The intense environmental conditions to which wind turbine components are exposed, particularly those located offshore, cause significant fatigue. Wind turbine blades are subjected to considerable gravitational, inertial, and aerodynamic loads, causing fatigue and degradation over the turbine's lifetime. The current study looks into modeling this phenomenon utilizing blade root moment data from a sensor in a high-fidelity simulator of an industrial-scale wind turbine. In recent years, there has been a surge in interest in combining control with fatigue load minimization. The research reported in this paper investigates the use of a predictive control model (PCM) in conjunction with a fatigue-based prognostic strategy to reduce wind turbine component damage (blades). The wind turbine now has a mechanism to run securely and optimize the trade-off between component life and energy production respect to the implementation of a system health management module with the MPC control. The controller objective was changed by including a new criterion that considers cumulative damage. The findings reveal that there is a trade-off between maximal power and accumulating damage. (Sanchez *et al*. 2015).

This article has three entries in the form of free software projects. The goal of this article is to examine the differences between the three open source projects offered in order to identify which of the three is more appropriate in some situations than others. First and foremost, we have the Delft Research Controller, a community-driven basic wind turbine controller (DRC). The DRC is a DISCON controller interface application for high-fidelity simulation software. The controller's versatility of control adjustments and knowledge, as well as its ease of use and unparalleled fit for wind turbine models, set it apart. The controller is designed to serve as a criterion controller for evaluating the most recent control algorithms. A single parameter file shapes the controller, allowing for modular customisation and the easy replacement of existing control implementations, as well as easy comparison, reproducibility, and the development of new algorithms. Second, the environment for developing and compiling graphical SimulinkDRC controllers has been expanded in the aftermath of the DRC. To provide a turbine controller's graphical design utility and collection. Finally, the FASTTool was created for educational reasons, with a concentration on the graphical component of wind turbine design (controllers). A graphical user interface to the NREL's FAST aeroelastic simulation code is also included. Through complete presentations and analysis tools, the tool streamlines the connection to the powerful FAST simulation program. FASTTool helps beginners to the area comprehend the design process by offering information on it, displaying modifications in a three-dimensional visualization of the turbine, and adapting to the current design. The software includes features such as spontaneous integrity checks and the ability to generate FAST input files for high-fidelity simulations based on the turbine design. (Mulders *et al*. 2020).

For deeper waters, floating wind generators are a viable option. The usage of a floating platform, on the other hand, introduces extra movements that must be considered during the design phase. The need for a movement control system is increasing. Several controllers designed exclusively for floating wind turbines have been developed. Some controllers were designed to prevent structural resonance, while others were designed to manage rotor speed and platform pitch. The evolution of a periodic state controller is described, which exploits the pitch of individual blades to perfect power output and reduce platform motions in the region of wind speeds over the rated speed. To boost the platform's restoring moments, individual blade pitching provides asymmetric aerodynamic loads in addition to the symmetric stresses provided by collective blade pitching. For comparison, a proportional-integral programmed gain (GSPI) controller, a collective blade pitch state-space (CBP SS) controller, and an individual blade pitch (IBP) controller designed utilizing periodic state-space (SS) control theory were all implemented on a floating platform. The collective pitch of the blades is used by the first two types of controllers. The latter makes use of the blades' individual pitches. In terms of rotor speed control and platform pitch limiting, the IBP controller surpasses the CBP SS controller under certain conditions. In addition to the symmetric loads provided by the collective pitch of the blades, the IBP controller realizes asymmetric aerodynamic loads. It also reduces the collective blade pitch, allowing the rotor speed to be controlled. The individual pitch of the blades is used by the SS controller. In Region 3, this created controller has proven to be effective (above the nominal wind speed region). (Namik and Stol 2010).

This research demonstrates the benefits of employing the self-optimizing control (SOC) technique for wind turbine generator control. High set-up and maintenance costs are associated with the enhanced optimality of complex operational schemes, such as real-time optimisation systems. While wind speed is an important element for wind turbine management, using it as a direct input results in poor power control response. As a result, the goal of this thesis is to improve the wind turbine's performance while lowering its cost. It is proposed to limit the functioning of the wind turbine in the face of inconsistent wind speed by using a constrained self-optimizing control. This sparks the concept of optimizing the wind turbine's performance while maximizing its power output. The turbine must limit its power output at high wind speeds to a nominal rate. The FAST v8 5MW earth model was used in MATLAB/SIMULINK to create the structure. The self-optimising control is based on the structure of the best off-line solution, and it then applies the optimisation approach, regulating torque and pitch to maximize the energy received. The fact that SOC is not a deterministic control adds to its potential. Instead, it learns the structure of the ideal solution off-line and then optimizes the strategy, adjusting torque and pitch to control wind speed uncertainty. In addition, because the SOC is based on prior knowledge of the ideal result structure, the control laws used to limit pitch and torque take an unanticipated path, bringing the WT operation to a higher CP. Despite the indeterminacy of wind speed, the SOC structure exploits both degrees of freedom to accomplish optimum operation. (Iordanov 2016).

The impact of wind turbine systems on frequency and voltage stability is becoming increasingly relevant as wind power penetration in the electrical grid grows dramatically. The rotor of a wind turbine is susceptible to three types of loads: aerodynamic, gravitational, and centrifugal. The stresses produce fatigue and vibrations in the blades, resulting in rotor blade degradation. These stresses can be overcome, and the amount of power collected, by utilizing a good pitch controller (PC), which adjusts the angle of attack of a wind turbine rotor blade into or out of the wind. Different loads are applied to each blade. Because of the rotor blades' different wind speeds. As a result, in the future, individual electric drives could be utilized to adjust the pitch of the blades in a technique known as individual pitch control. These stresses can be overcome, and the amount of power collected, by utilizing a good pitch controller (PC), which adjusts the angle of attack of a wind turbine rotor blade into or out of the wind. Because the wind speed varies throughout the rotor blades, each blade is subjected to various loads. As a result, in the future, individual electric drives could be utilized to adjust the pitch of the blades in a technique known as individual pitch control. They created a mathematical model of the wind turbine pitch control system and used MATLAB/Simulink to simulate it with a PID controller to get the best response. The response is produced with less delay and rising time by the PID controller, however there are minor oscillations. (Silpa Baburajan and Ismail 2017).

2 MATERIALS AND METHODS

2.1 Introduction

A wind turbine is a machine that converts wind energy into electricity. Wind turbines are used by power companies to obtain wind energy. A 'wind farm' will usually have a large number of turbines. When the twist passes through a turbine's blades, they rotate, propelling a rotor within a generator and providing power. Every turbine is free to operate. The energy flows via the turbine's linkages, down the turbine tower, and into the nearby power grid or the national grid, where it is combined with the vitality from the other wind turbines in the wind farm.

2.2 Type of Wind Turbine

2.2.1 Vertical axis wind turbine

The main rotor shaft of vertical axis wind turbines, often known as VAWTs in Figure (2.1), is positioned vertically. The wind turbine should not be directed into the wind, according to this course of action's main point of view. This is advantageous at locations where the wind direction is very varied or where the wind is turbulent. With a vertical axis, the generator and other key components can be placed close to the ground, eliminating the need for the tower to support them. This also reduces the amount of support required. The most significant disadvantage of a VAWT is drag while rotating towards the wind. Vertical-axis turbines are difficult to put on towers, thus they are commonly introduced closer to the base on which they rest, such as the ground or the roof of a structure. At a lower elevation, the wind speed is slower, hence there is less wind vigor available for a given size turbine. Air flow near the ground and other objects can cause turbulent flow, which can cause vibration concerns such as noise and bearing wear, requiring more maintenance and shortening the service life. When a turbine is positioned on a rooftop, however, the building often redirects wind over the roof, which can double the turbine's wind speed. The height of the rooftop mounted turbine tower should be approximately half the height of the building for maximum wind energy and minimal wind turbulence(Ragheb 2011).

Figure 2.1 Vertical Axis Wind Turbine (Ragheb 2011).

2.2.2 Horizontal axis wind turbine

Horizontal axis wind turbines in Figure (2.2), often known as HAWTs, are the most common type of wind turbine that most people think of. A HAWT is similar to a windmill in that it features propeller-like blades that rotate on an even hub. The primary rotor shaft and electrical generator are located at the highest point of a tower in hub wind turbines, and they should be directed into the wind. A simple wind vane positioned square with the rotor (blades) points small turbines, whereas large turbines typically use a wind sensor combined with a servo engine to convert the turbine towards the wind. Most large wind turbines contain a gearbox that converts the rotor's moderate rotation into a faster revolution that can drive an electrical generator. The turbine is usually pointed upwind of the tower because it causes turbulence behind it. The blades of wind turbines are solid to prevent them from being forced against the tower by strong winds. Downwind machines have been used regardless of turbulence since they do not require an additional device to maintain them aligned with the wind. In addition, in high gusts, the blades can be allowed to curl, reducing their cleared zone and, as a result, their wind resistance. Because turbulence leads to tiredness and dissatisfaction, unwavering quality is the most important factor in HAWT(Kumara, Hettiarachchi and Jayathilake 2017).

Figure 2.2 Horizontal Axis Wind Turbine (Sebit, Alpman and Yılmaz 2014).

2.2.3 Parts of wind turbine

The wind turbine models are composing by three parts: the first part is the rotor, which include the blades who convert wind energy to low speed rotational energy (kinetic energy). The second part is the generator add the electrical generator, which contain all control circuit with gearbox that convert rotational low speed into electric power and the last we have the structure who hold all of the previous components that is the tower and the nacelle. Wind turbine are categorize in two main groups depending into their axe which the turbine rotate. We have the horizontal axis and the vertical axis (Vyas, Singh and Santoso 2011).

Power, torque, and thrust are three indicators that fluctuate with wind speed and represent a wind turbine's performance. The rotor's power is determined by the quantity of energy it captures, the gearbox's torque is determined by the size of the gearbox, and the rotor thrust has a significant impact on the tower's structural design.

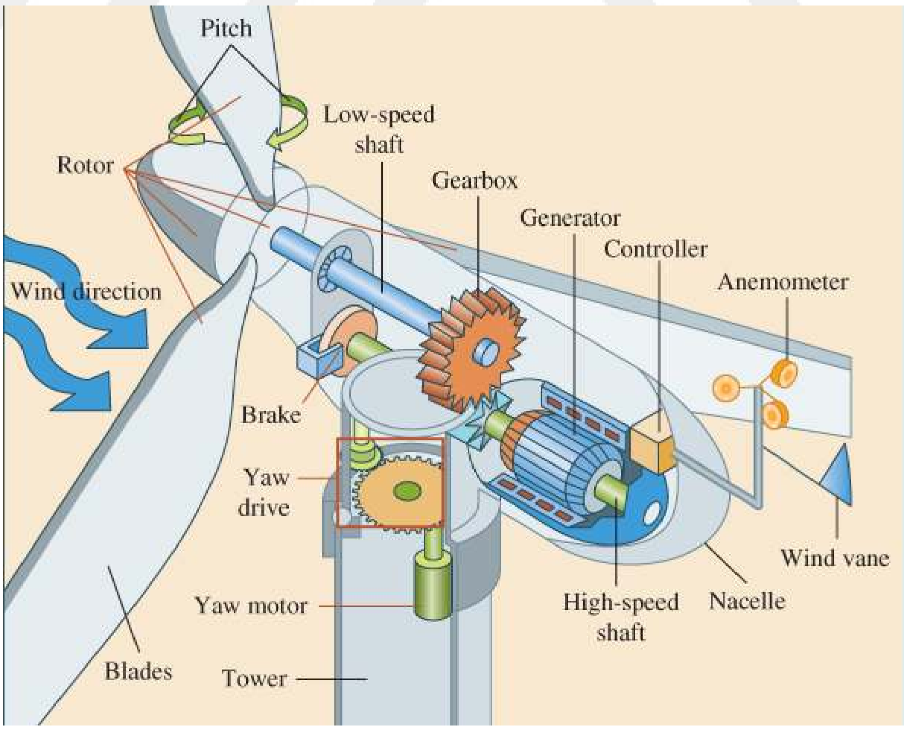


Figure 2.3 Part of Wind Turbine (Sebit, Alpman and Yılmaz 2014).

As illustrated in the diagram, a wind turbine is made up of the following subsystems Figure (2.3):

Rotor blade and hub;

Nacelle contains shafts, gearbox, couplings, brake, and generator;

Tower the hold the nacelle;

Electrical system such as switchgear, transformers, cables, and power converters in Figure (2.5).

- Tower: To reduce ground disturbance from the ground, the tower, which is normally a steel tube, should be as high as feasible. However, the amount of material utilized is costly, and the weight must be kept to a minimum. A compromise is to utilize a mast that is slightly larger than the wind generator's rotor diameter. The tour's construction does not stop at supporting the weight of the nacelle and rotor pales. She must support the enormous static charges generated by the changing energy of the wind, as well as the rotor's pales. The wind's ever-changing energy. In most cases, a tubulaire development in ciment or acier is used. Another option is to use a triangular tour structure. Depending on the consistency of the base soil, plates or tas foundations are used to ensure the stability of an éolienne.
- Nacelle: Slow and fast shafts, bearings, and a gearbox are all part of the nacelle. The wind rotor can be attached to the electric generator thanks to all of these mechanical elements. The disc brake, unlike the aerodynamic brake, stops the system in the event of an overload. The nacelle, which is usually a synchronous or asynchronous machine, and the hydraulic or electrical systems that control the blades' orientation (aerodynamic brake) (necessary to keep the surface swept by the wind generator perpendicular to the wind direction). In addition, an air or water cooling system, an anemometer, and the wind turbine's electronic management system are frequently utilized.

- ✓ Gearbox: The gearbox converts the rotor's 18-50 rpm movement into the 1.500 rpm required by the generator. As a result, the gearbox faces the challenge of coordinating the slow-moving rotor's turnspeeds with the fast-moving generator, and it often has a few stages to account for various wind conditions. If a specially designed multi-pole ring generator is used, the gearbox is no longer required.
- ✓ Generator: When it comes to high-power wind turbines, doubly-fed asynchronous generators are frequently used. Unlike when using ordinary asynchronous generators, the working turn speed can be altered fairly here. Synchronous generators are another option. Because of the fixed revolution conduct, synchronous generators can only be connected to the grid via transformers. The disadvantage of requiring 22 entangled control systems is offset by improved network similarity and overall competency.
- ✓ Couple: The coupling between the main shaft and the transmission is rigid due to the enormous torque. The type of brake is determined by the blades' control component.
- Rotor: is made up of blades that are assembled in a hub. The number of blades of wind turbines meant to generate power normally ranges from one to three, with the three-bladed rotor being by far the most popular because to its favourable balance of cost, vibratory behaviour, and noise. At this time, the rotor had three pales and a moyeu plat command. The rotor's pales are mostly made of plastic reinforced with glass or carbon fibers (GRP, CFRP). The profile of the pales is 21, similar to that of an airplane wing.

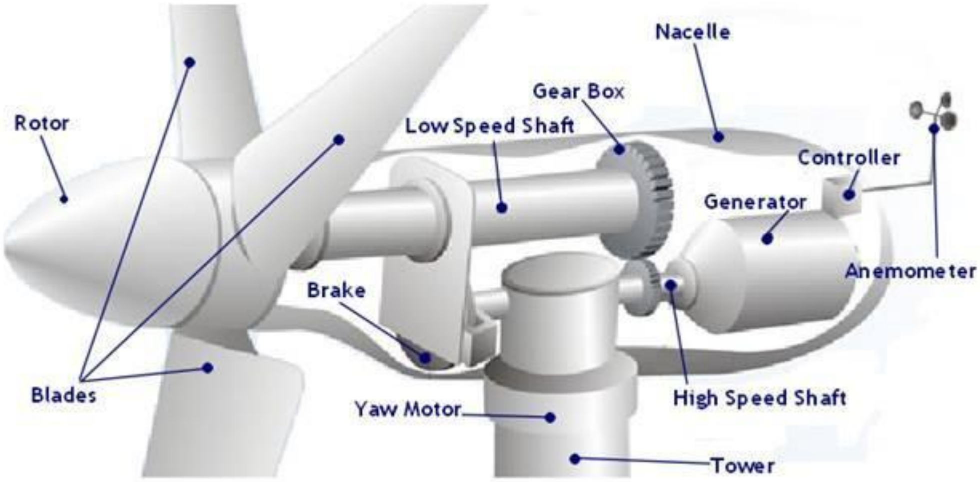


Figure 2.4 Component of wind turbine (Ahmad and Abdul-Hussain 2017).

It capture energy from the air density (moving air to energy). The following Equation (2.1) shows how air density, power coefficient, air density, and turbine swept area effect the captured energy:

$$P = 0.5 \times \rho \times Cp \times V3 \times A \quad (2.1)$$

Where:

P = Mechanical power in the moving air (Watt).

ρ = Air density (kg/m3).

A = Area swept by the rotor blades (m2).

V = Velocity of the air (m/s).

Cp = Power coefficient.

According to scientist Betz, the maximum power that can be obtained from an ideal turbine rotor with infinite blades from wind under perfect conditions is 59.26 percent

(0.5926 times) of the power available in the wind. Due to structural and economic concerns, wind turbines are designed with two or three blades, and the quantity of power they can generate is closer to 50 percent (0.5 times) of the available power.

The aerodynamic torque occurring at the turbine is therefore a function of this power is therefore a function of this power:

$$C\_aero = \frac{P\_aero}{\Omega\_t} = \frac{Cp(\lambda,\beta).1}{(2\Omega\_t\,)}.(\rho.v)^3.\pi.R^2 \tag{2.2}$$

The gearbox is the link between the turbine and the generator. It is used to match the fastest speed of the generator to the slowest speed of the turbine and is often modelled by the following expressions:

$$\Omega_t = \frac{\Omega\_mec}{G} \tag{2.3}$$

$$C_g = \frac{C_{aero}}{G} \tag{2.4}$$

The generator shaft is modelled by the following expressions:

$$\frac{J\,(d\Omega_{mec})}{dt} = \sum C = C_T - C_{vis} \tag{2.5}$$

J is total inertie:

$$J = \frac{J_{turb}}{G^2} + J_{generator} \tag{2.6}$$

C_{vis} viscous frictional torque:

$$C_{vis} = f\Omega_{mec} \tag{2.7}$$

C_T total torque which equals the superposition of the generator and electromagnetic torques. The expression of the mechanical shaft then becomes:

$$J\frac{d\Omega_{mec}}{dt} + f\Omega_{mec} = C_T = C_g - C_{em} \tag{2.8}$$

Tip speed ratio (TSR) of wind turbine is defined by:

$$\lambda = \frac{\Omega \times R}{V} \tag{2.9}$$

Where:

Ω = Mechanical speed at the rotor shaft of the wind turbine (rad/s).

R = Radios of the blade (m).

V = Velocity of the air (m/s).

The rotor power coefficient, represented by Cp, is calculated using the TSR and blade pitch angle. The rotor power coefficient can be determined using the following formula:

$$Cp = \frac{Extracted\ power}{(power\ in\ wind)} \tag{2.10}$$

$$\mathrm{Cp} = \frac{\mathrm{P_{rotor}}}{\mathrm{P_{wind}}} \tag{2.11}$$

Variable-speed winds turbines are arm with the pitch-change mechanism (Pitch angle control) to modifies the blade pitch angle and get a best power coefficient profile because it controls its rotation speed.

2.2.4 Pitch angle control

Cp is the power coefficient of the wind turbine, β is the blade pitch angle and λ is the tip speed ratio (Ahmad and Abdul-Hussain 2017). The value of Cp is very nonlinear, varying with wind speed, turbine rotational speed, and turbine blade parameters such as pitch angle. In this study, the value of Cp (β, λ) is calculated as follows:

$$Cp = c1\left(\frac{c2}{\lambda} - c3\beta - c4\right)e^{\frac{c5}{\lambda_i}} + c6\lambda \tag{2.12}$$

$$Cp = 0.5176\left(\frac{116}{\lambda} - 0.4\beta - 5\right)e^{\frac{-21}{\lambda_i}} + 0.0068\lambda \tag{2.13}$$

Where:

$$\frac{1}{\lambda} = \frac{1}{\lambda + 0.08\beta} - \frac{0.035}{\beta^3 + 1} \tag{2.14}$$

As a result, any change in rotor or wind speed causes a change in the tip speed ratio, resulting in a shift in the power coefficient. Figure (2.5) displays the mechanical power of a 5 MW wind rotor as a function of rotor speed at various wind speeds.

As the wind speed increases, the turbine's power output increases. The generating power of the turbine reaches its rated power at rated wind speed. Because the output power will increase as the wind speed rises, the control system must keep the power constant at the design limit. For safety reasons, the turbine is shut down when the wind speed exceeds the cut-out wind speed.

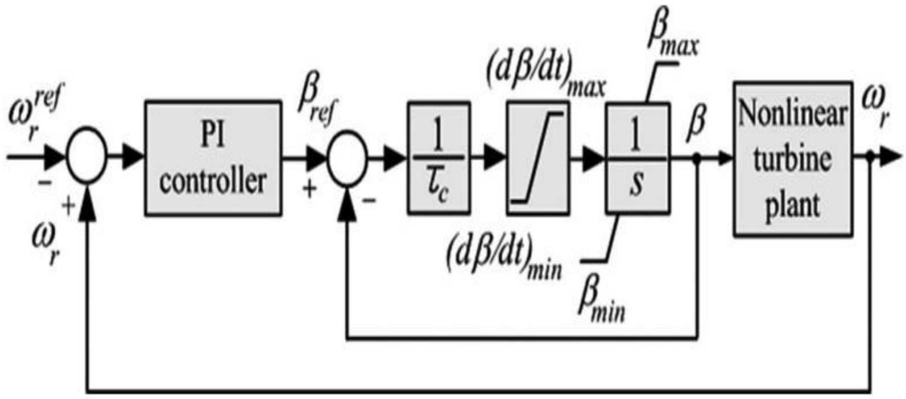


Figure 2.5 Pitch angle control (Bossanyi 2005)

2.3 Wind Turbine With Fast System

In this model we can see that the component is connected to the FAST block to work properly in Figure (2.6). FAST helps to move each element with dynamic multibody formulas. It does this by stimulating the dynamic multibody. The FAST method helps to calculate the fatigue and extreme load on the wind turbine. Therefore, FAST takes the characteristics of the wind turbines to perform a concrete simulation. As shown in the figure 2.6 below, our goal will be to control the blades of the wind turbine individually using torque.

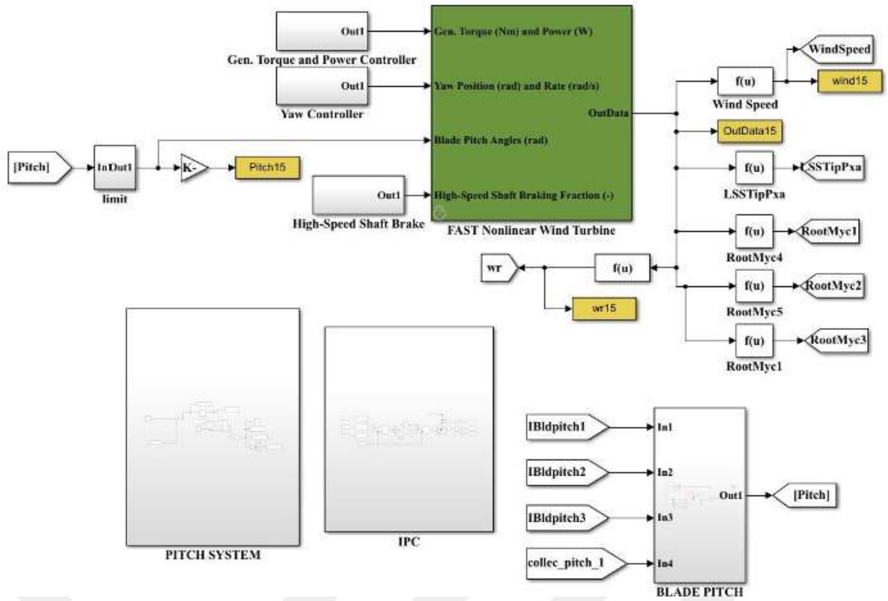


Figure 2.6 Fast modelling.

2.3.1 Fast system

A mathematical model encompassing the complicated, non-linear dynamic effects of the WT is required for the design of a control system. The National Renewable Energy Laboratory (NREL) developed the FAST (Fatigue, Aerodynamics, Structures, and Turbulence) model as part of a US Department of Energy-sponsored software development project for the wind power engineering community. FAST is a time-domain non-linear aeroservo-elastic modeling tool that combines aerodynamic, control, electrical, and structural dynamic models. The architecture of the FAST v8 model is shown in Figure (2.7).

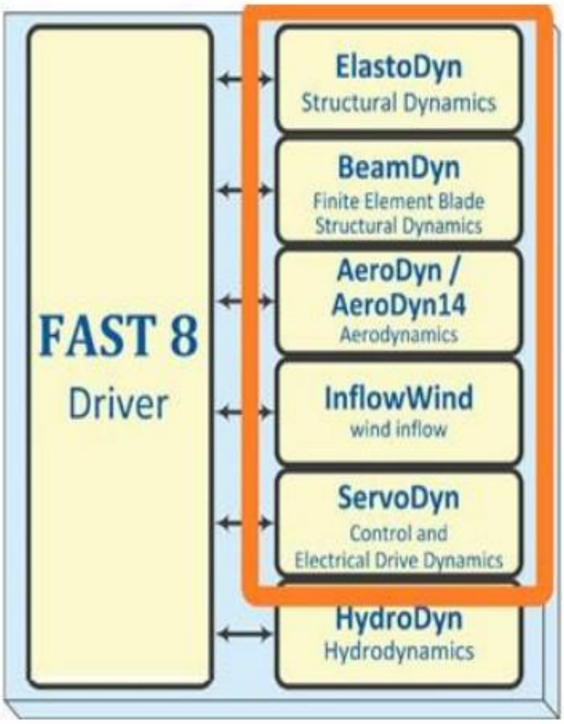


Figure 2.7 FAST method (Jonkman and Buhl Jr 2005).

The new FAST v8 is made up of a modularization framework that is detailed in Refs. (Jonkman 2013, Jonkman and Buhl 2005). Aerodynamics (AeroDyn); hydrodynamics (HydroDyn); control and electrical drive dynamics (ServoDyn); rotor, drivetrain, nacelle, tower, and platform structural dynamics (ElastoDyn); multimember substructure structural dynamics (SubDyn); mooring statics and dynamics (MAP); and ice loads are among the modules included in the framework (IceFloe). BeamDyn, a nonlinear finite-element structural dynamics module for blades, FEAMooring, and IceDyn, a finite-element module for advanced ice loads, are also in the works. The architecture differences between FAST v7 and FAST v8 are shown in Figure 2.8.

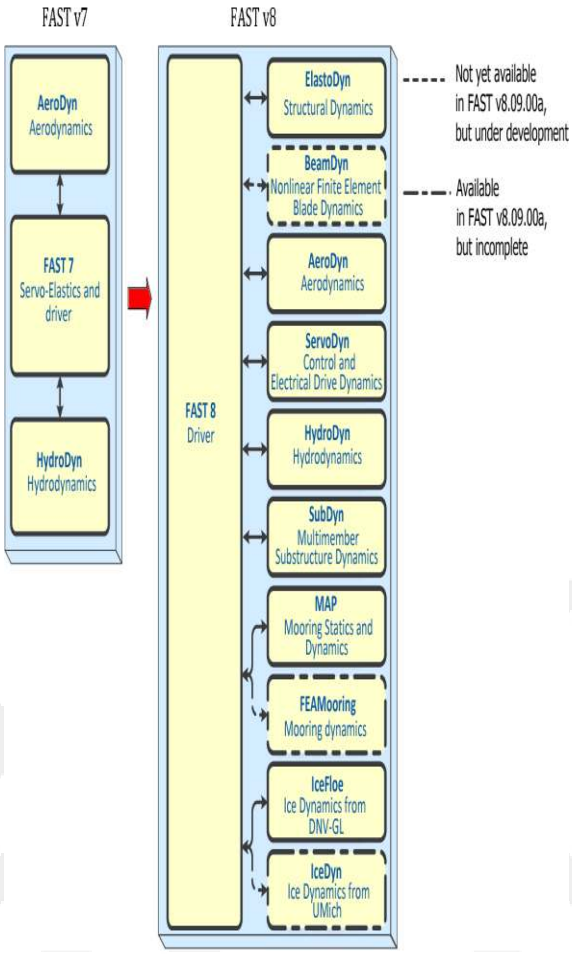


Figure 2.8 FAST evolution (Jonkman and Buhl Jr 2005).

For the wind turbine we use the FAST model in MATLAB/SIMULINK developed by American Renewable Energy Laboratory (NREL). This FAST model allows for both collective and individual control of the blades of the wind turbine in Figure (2.8). For this purpose, the "Test 11.fst" is used to simulate an autonomous wind turbine. The reading of the system by MATLAB/SIMULINK consists in using a file such as 'test11.fst' then launching 'run' to add the data of the file in the simulation the file. (Jonkman *et al*. 2009). The Simulink implementation of "Test 11" is discussed in Appendix [1] and described in (Civelek 2017).

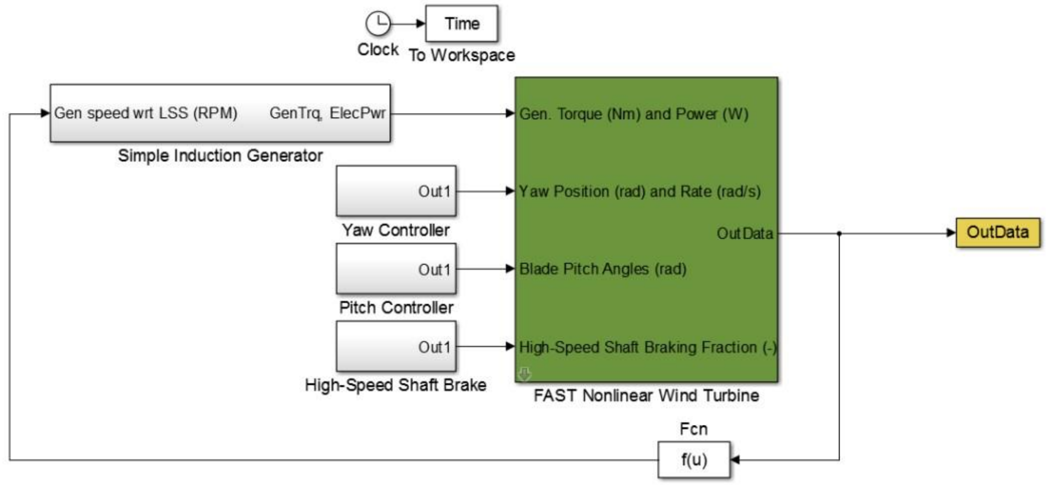


Figure 2.9 Fast Matlab/Sımulınk.

The Matlab/Simulink environment's S function has been modified from the FORTRAN language's wind turbine rapid simulator in Figure (2.9). As a result, before starting the Matlab Fast simulator, the S function's route must be specified. Fast offers user's ready-to-use models of wind turbines and lets users change numerous wind turbine variables in more complex studies. Table 2.1 displays the Fast models that are ready for use.

Test01	AWT-27CR2	2	27	175	Flexible, fixed yaw error, steady wind
Test02	AWT-27CR2	2	27	175	Flexible, steady wind, startup, high-speed shaft brake shutdown
Test03	AWT-27CR2	2	27	175	Flexible, free yaw, steady wind
Test04	AWT-27CR2	2	27	175	Flexible, free yaw, turbulence
Test05	AWT-27CR2	2	27	175	Flexible, steady wind, startup, shutdown
Test06	AOC-15/50	3	15	50	Flexible, steady wind
Test07	AOC-15/50	3	15	50	Flexible, free yaw, turbulence
Test08	AOC-15/50	3	15	50	Flexible, fixed yaw error, steady wind
Test09	UAE VI downwind	2	10	20	Flexible, yaw ramp, steady wind
Test10	UAE VI upwind	2	10	20	Rigid, power curve, ramp wind
Test11	WP 1.5 MW	3	70	1500	Flexible, variable speed & pitch control, pitch failure, turbulence
Test12	WP 1.5 MW	3	70	1500	Flexible, variable speed & pitch control, ECD event
Test13	WP 1.5 MW	3	70	1500	Flexible, variable speed & pitch control, turbulence
Test14	WP 1.5 MW	3	70	1500	Flexible, stationary, linearization in vacuum
Test15	SWRT	3	5.8	10	Flexible, variable speed control, free yaw, EOG01 event
Test16	SWRT	3	5.8	10	Flexible, variable speed control, free yaw, EDC01 event
Test17	SWRT	3	5.8	10	Flexible, variable speed control, free yaw, turbulence
Test18	NREL 5 MW - Land-based	3	126	5000	Flexible, DLL control, tower potential flow and drag, turbulence
Test19	NREL 5 MW - OC3 Monopile	3	126	5000	Flexible, DLL control, tower potential flow, turbulence, irregular waves
Test20	NREL 5 MW - OC3 Tripod	3	126	5000	Flexible, DLL control, tower potential flow, steady wind, regular waves with 0 phase
Test21	NREL 5 MW - OC4 Jacket	3	126	5000	Flexible, DLL control, tower potential flow, turbulence, irregular waves, marine growth
Test22	NREL 5 MW - ITI Barge	3	126	5000	Flexible, DLL control, irregular waves, turbulence
Test23	NREL 5 MW - MIT-NREL TLP	3	126	5000	Flexible, DLL control, turbulence, irregular & multidirectional waves
Test24	NREL 5 MW - OC3 Hywind	3	126	5000	Flexible, DLL control, turbulence, irregular waves
Test25	NREL 5 MW - OC4-DeepCwind Semi-Submersible	3	126	5000	OC4 Load Case (LC) 3.7: steady wind, white noise second-order waves
Test26	NREL 5 MW - Land-based	3	126	5000	Flexible, DLL control, tower potential flow and drag, turbulence - using BeamDyn

Figure 2.10 CertTest Fast (Jonkman and Buhl Jr 2005).

If the user wishes to make additional alterations in addition to these pre-made templates in Fast, he or she must modify some files in the Fast folder in Figure (2.10). It is necessary to open the "CerTest" folder in the "FAST" folder for this reason. In the folder "CerTest" there are text files related to the wind turbine models in Table 2.1. The files related to the turbine to be analyzed will start with the name and test number of the turbine. For example, if you wish to operate with a 1.5 MW turbine with adjustable pitch control and variable speed, you should select test 11 (up to 14 possible) (up to 14 possible). Test11 AD.ipt, Test11 ElastoDyn.dat, and Test11 ServoDyn.dat are the titles of the three files in "CerTest" for this turbine. All of the structure's parameters can be found in these files. The Fast user handbook contains comprehensive information.

2.3.2 Individual pitch angle controller

Changing the pitch angle of the blades is a possible means of limiting the turbine's effectiveness in heavy winds while also allowing individual control. Electric or hydraulic pitch actuators are utilized to set the blades in the required position. The information signals used to determine the pitch setpoint can be used to classify blade control systems. the signals that were used to create the pitch setpoint.

- The pitch point reference can be obtained in an ideal environment by bending the pitch angle vs wind speed. This control approach is ineffective because the tremendous wind speed cannot be precisely measured.
- The pitch angel is adjusted based on the difference between the needed and output power.
- The pitch angle can be adjusted by a classical control method or use modern methods to obtain relevant results.

The third method is employed to alter the pitch angle in this thesis. The control approach adopted in this thesis is based on the wind turbine's individual blade control method.

The Park transformation is used to effectively control the individual blades.

Figure (2.11) illustrates the control scheme and demonstrates how the three blade root out-of-plane bending moment signals, which are typically derived from flapwise and edgewise signals resolved through pitch angle, are transformed into two orthogonal d- and q-axes (which can be thought of as the horizontal and vertical axes) by means of a special transformation. The two d- and q-axis pitch demands are translated by the reverse transformation to create pitch demand increments for each blade. A controller for each axis generates a pitch demand for that axis. To determine the total pitch requirement for each blade, these are added together with the collective pitch demand (which regulates torque and thrust and, as a result, rotational speed, tower vibration, etc.).

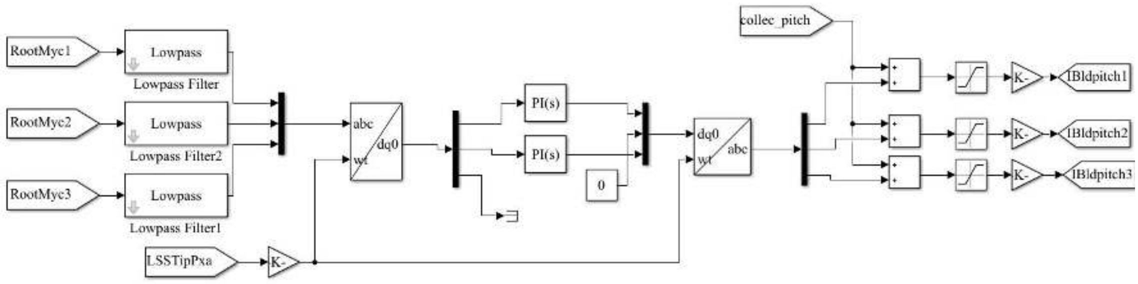


Figure 2.11 Individual pitch angle control.

3 METHODS OF CONTROL

3.1 Introduction

In this chapter, we'll look at three different control strategies that can be used to manage each pitch angle individually. To maximize the amount of energy received, a wind turbine's pitch angle must be modified using an effective control approach. As a result, Nichols-Ziegler, Genetic-Algorithm, and Particule Swarm Optimization controllers will be used to change the pitch angles.

3.2 Nichols-Ziegler

The L and T values used in computing the necessary coefficients for the PI and PID controllers were found using the output power curve data in the Matlab environment, as shown in Figure 3.1. The wind turbine output power curve is broken into four zones in the Matlab environment by looking at the system without applying it to a controller(Pai, Chang and Huang 2010). The L value is in the I region, which is called Cut-in. The T value can be calculated in region II. The clipped region is called in region IV, and the K value can be determined. The T value was obtained in this study by utilising the Nichols-Ziegler method to draw the tangent line in region II of the output power curve. The x-axis in the cutout part between regions I and II is indicated by the L value. The L and T values are always different when using a PI or PID controller. The wind system values calculated or stated in this section must have a pitch angle of zero. Following that, using the methods in Table 1, Kp, Ki, and Kd were easily derived (Sukmono 2019).

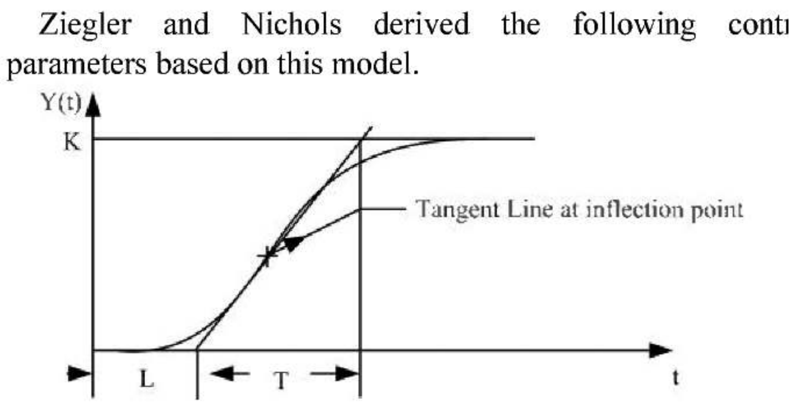


Figure 3.1 L and T curve (Civelek 2017)

To determine Kp, Ki, and Kd, the PI and PID coefficients were derived using the open loop system according to Table 1. Looking at the Matlab simulation yields the L and T values. The coefficients Kp, Ki, and Kd were determined after identifying the L and T values of the open-loop system according to Figure (3.1) by looking at the output power curve in Matlab in simulation. This approach is not used in this study because the system does not contain an open loop. The closed loop approach is shown below. The PI and PID controller's closed loop approach has been determined as shown in Table 1: The coefficients Kp, Ki, and Kd were derived after identifying the L and T values of the open loop system according to Figure 3.1 by looking at the output power curve in Matlab in simulation. This approach is not used in this study because the system does not contain an open loop. The closed loop approach is shown below. The PI and PID controller's closed-loop technique is determined as shown in Table (3.1).

Table 3.1 Calculation of PI and PID closed loop coefficients (Civelek 2017).

Ziegler-Nichols methods giving K' values (loop times considered to be constant and equal to dT)			
Control type	Kp	Ki'	Kd'
P	$0.50K_c$	0	0
PI	$0.45\ K_c$	$1.2\ K_p dT/P_c$	0
PID	$0.60\ K_c$	$2\ K_p dT/\ P_c$	$K_p\ P_c/(8dT)$

- PI design controller

Depicts the PI regulator's transfer function. The controller input signal is generated by an intrinsic portion of the controller. The PI controller reduces the system's steady state error to some extent.

The PI controller's transfer function is described in Equation (3.15). The Ziegler Nichols method is used in this study to optimize the output power of the wind turbine and keep it at the setpoint. Only Kp and Ki are present in the PI controller.

$$G_c(s) = K_p + \frac{K_I}{s} \tag{3.15}$$

- PID design controller

Figure 8 depicts the transfer function of the 'PID controller,' which includes Kp, Ki, and Kd. The integral part of the PID controller generates a signal proportional to the controller input signal's time integral.

The PID controller's transfer function is described in Equation (3.16). In this case, the Nichols Ziegler approach optimizes the wind turbine output power set point more effectively than the PI controller. Kp, Ki, and Kd are the three components of the PID controller.

$$G_c(s) = K_p + K_d + \frac{K_I}{s} \tag{3.16}$$

3.3 Genetic Algorithm

The genetic algorithm is a type of random search method for solving nonlinear formula and optimizing complex tasks in Figure (3.2). GA maintains a population of potential solutions called individuals or chromosomes that evolve iteratively and uses probabilistic transition rules instead of deterministic principles. A generation is the name given to each iteration of the algorithm. A weighted combination of objective functions, namely the integral square error (ISE), the integral absolute error (IAE), and the integrated time absolute error (ITAE), has been proposed to adjust the optimised PID parameters in a continuous stirred tank reactor (CSTR) process using a genetic algorithm (GA)-based proportional integral derivative (PID) controller (ITAE). The chemical and biochemical sectors are focused on optimizing PID controller parameters. The operational range of processes with dynamic non-linearity has been reduced thanks to PID controls. A random population is commonly used to start the genetic process described in Figure 3.2. A chromosome is a real-valued number or a binary string that represents a population (mating pool). The objective function, which assigns each individual a corresponding number called fitness, measures and evaluates the individual's performance (Jayachitra and Vinodha 2014).

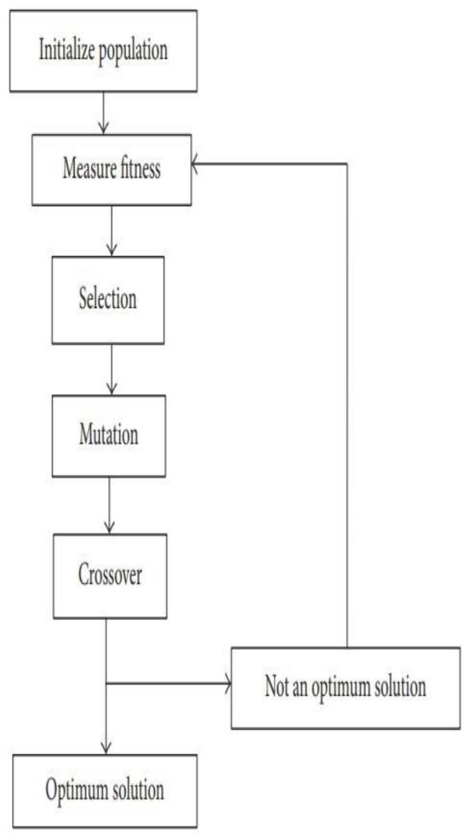


Figure 3.2 Algorithm Genetic-algorithm (GA) (Jayachitra and Vinodha 2014).

Each chromosome's fitness is measured, and a survival-of-the-fittest approach is used. The error value is utilized to assess each chromosome's fitness in this study. A genetic algorithm has three major operations: breeding, crossover, and mutation. Figure depicts the series of operations involved in GA. Steps of the genetic algorithm

Steps 1: Initialise the parameter with a population of random solutions, such as crossover rate, mutation rate, number of clusters and number of generations. Determine the coding mode.

Step 2: Calculate and evaluate the value of the fitness function.

Step 3: Carry out the crossover and mutation operation and form the new cluster.

Step 4: Repeat step 2, until the best value is obtained

Encoding and decoding are two different things. Instead of looking at the parameters directly, genetic algorithms work with a population of strings or chromosomes.

Fitness: A chromosome's fitness is assessed using a fitness metric. A chromosome with a greater fitness value has a higher chance of contributing to one or more offspring in the future generation, according to the survival of the fittest principle. The performance criterion is linked to the fitness function via a genetic algorithm, and the ideal PID parameters are determined by minimizing an objective that incorporates a weighted mixture of IAE, ISE, and ITAE.

Selection: The genetic algorithm's basic operator is breeding. It is run on the principle of "survival of the fittest." The present population's chromosome is reproduced or copied into the next generation in each generation, according to the reproduction probability Pri, which is defined in

$$P_{ri} = \frac{(F_i(?))}{(\sum_{i=1}^{P_i} F_i(?))} \tag{3.17}$$

Pl stands for the population size.

Crossover: The genetic algorithm is directed to the best individuals by breeding. The crossover operation is used to share information between any two chromosomes in the mating pool via a probabilistic decision and to give a mechanism for mixing chromosomes with the splice point.

Mutation: However, in genetic algorithms, the gene pool tends to become increasingly homogeneous as a better gene emerges after several generations, resulting in the convergence of a non-optimal solution too soon. To avoid this unwanted convergence, a probability-based mutation of the third genetic operator is included into the genetic algorithm. The mutation is a once-in-a-while change in the gene from zero to one or one to zero, with the mutation location chosen at random.

A genetic algorithm is used to control PID. The following is a summary. GA creates a random population to begin with, which is then implemented with a modest population size to allow the controller to be optimized and converge quickly. The PID parameters Kc, TI, and Td are encoded in binary strings called chromosomes to define the initial population. Each chromosome's fitness is determined by translating its binary text into real-world values that represent the PID parameters. The PID controller receives each set of PID parameters. Individual cost functions such as ISE, IAE, and ITAE, as well as a weighted mixture of these three cost functions, are used to determine the complete system response for each PID parameter value and its initial fitness value. Steps 2 and 3 will be repeated till the process is complete. Generations that have acquired the highest level of fitness illustrate in Figure (3.3). The final purpose of GA is to find global PID values (Kc, TI, and Td) with a minimal fitness value that will allow the CSTR plant to run through the entire range. The code of the G-A optimization method is put in the appendix to show the coding used in the MATLAB/SIMULINK shown below in the Appendix [3].

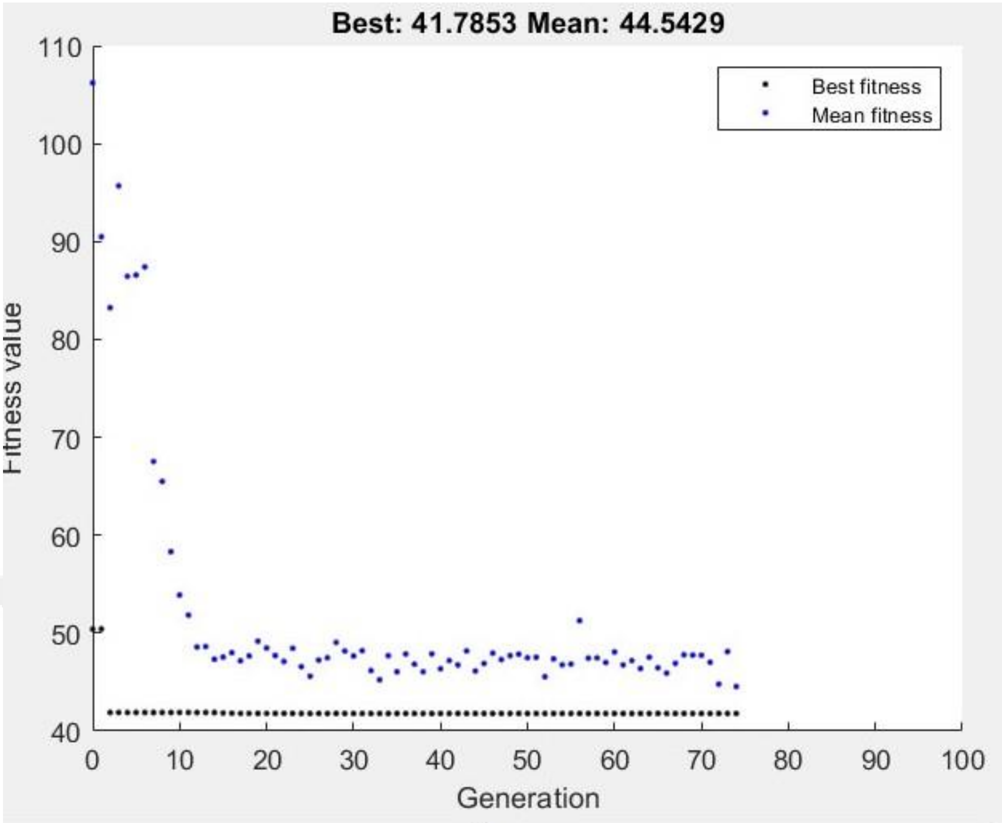


Figure 3.3 Best fitness GA

3.4 Particle Swarm Optimization

Because of their functional simplicity and performance, PI (proportional-integral) and PID (proportional-integral-derivative) controllers are used to control quantities such as voltage, power, and frequency in most industrial systems (robust and reliable) in Figure (3.4). To tune the PI/PID controller settings in the loop of electrical systems, several ways have been presented. According to the literature, the system's performance is influenced by both the controller and the objective function used to set the controller gains. The frequency control method has incorporated a great variety of nature-inspired algorithms and artificial intelligence techniques in recent years. Particle Swarm Optimization is a population based stochastic optimization technique developed by Dr. Eberhart and Dr. Kennedy in 1995 inspired by the social behavior of birds or schools of fish. The PSO algorithm optimization technique is utilized for the most optimum values of Kp and Ki in real-time operation to reduce transient response, eliminate time overshoot, and obtain low steady state error owing to load changes. When compared to other methods, PSO has been shown to produce superior results in a faster and less expensive manner. It's also possible to parallelize it. Furthermore, it does not take into account the gradient of the problem to be solved. To put it another way, unlike classic optimization approaches, PSO does not require a differentiable issue. We haven't yet discussed the inertia, cognitive, or social coefficients, as you may have noted. The levels of exploration and exploitation are controlled by these coefficients. Particles' ability to target the best solutions found so far is known as exploitation. Particles' ability to evaluate the complete research space is referred to as exploration. The remainder of the article's challenge will be to determine the impact of these coefficients in order to strike a suitable balance between exploration and exploitation. This assumption of a balance between exploration and exploitation makes little sense unless both are quantifiable, and such a balance is neither required nor sufficient from the standpoint of efficiency. However, I shall use these terminology in this post for the sake of clarity (GirirajKumar, Jayaraj and Kishan 2010).

$$\mathrm{P_{ri}} = \frac{(\mathrm{F_i}\ (\square))}{(\sum_{i=1}^{P_i} \mathrm{F_i}\ (\square)} \tag{3.18}$$

$$P_i^t = [x_0^t, i, x_1^t, i, x_2^t, i, x_3^t, i, \dots \dots, x_n^t,] \tag{3.19}$$

In a research space, a group of particles (possible solutions) of the global minimum. In this search space, there is only one global minimum. None of the particles knows where the global minimum is, but they all have fitness values that need to be optimized using the fitness function.

$$V_i^t = [v_0^t, i, v_1^t, i, v_2^t, i, v_3^t, i, \dots \dots, v_n^t\ ] \tag{3.20}$$

Each of these particles has a velocity that allows them to update their position over time in order to discover the global minimum.

$$P_i^{t+1} = \ P_i^t + V_i^{t+1} \tag{3.21}$$

$$V_i^{t+1} = \omega V_i^t + c_1 r_1 \left(P_{best(i)}^t - P_i^t\right) + c_2 r_2 \left(P_{bestglobal}^t - P_i^t\right) \quad (3.22)$$

Each particle's speed is stochastically accelerated towards its prior best position (personal best) and the group's best solution across iterations in the search space (global best).

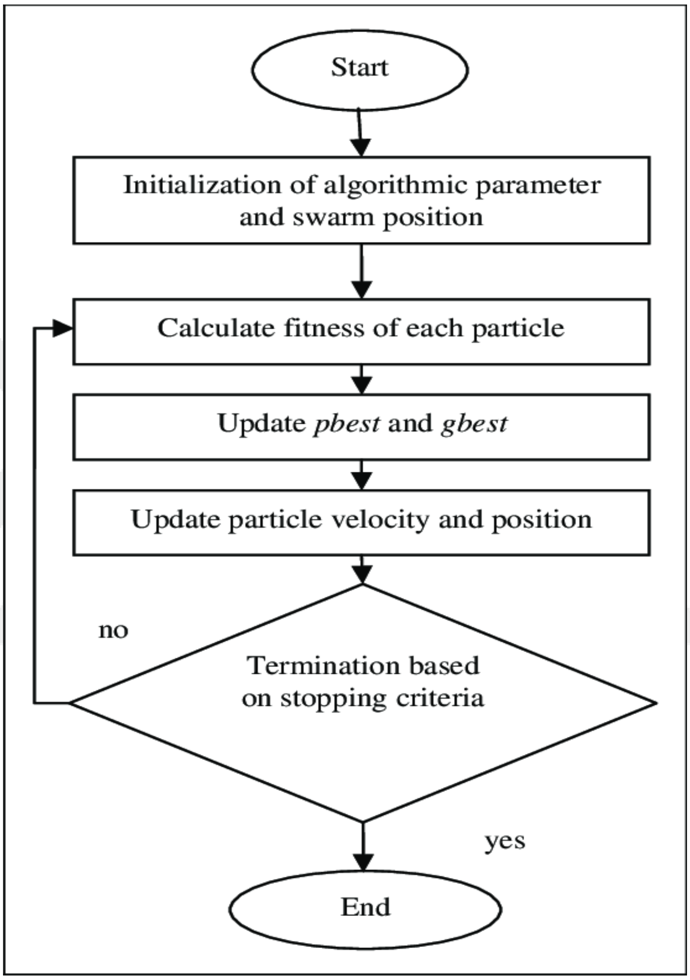


Figure 3.4 Algorithm Particule Swarm Optimization (PSO) (GirirajKumar, Jayaraj and Kishan 2010).

To put it another way, each particle's velocity is modified at each iteration. This velocity is governed by the two best values found so far and is subject to inertia. The first value is the particle's greatest personal solution to date. The second is the best global answer the swarm of particles has discovered thus far. As a result, each particle has its best personal answer as well as the best global solution stored in its memory. The code of the

PSO optimization method is put in the appendix to show the coding used in the MATLAB/SIMULINK shown below in the Appendix [4].

4 RESULTS AND DISCUSSION

4.1 Introduction

The control technique stated in the other chapters is taken into consideration in this section. A wind turbine's pitch angle needs to be managed in relation to time. A suitable control technique is required to alter a turbine's pitch angle in order to maximize the energy received because the wind speed is always changing. The pitch angles in this thesis are changed using PI, GA-PI, and PSO-PI controllers.

4.2 Comparison of Control Methods

The performance of the PI, GA-PI, and PSO-PI strategies should be compared in order to identify the most suitable strategy for controlling the individual blades with the FAST simulation.

To test a control on the wind turbine's three blades, we use a real wind condition with turbulence, as shown in figure (4.1).

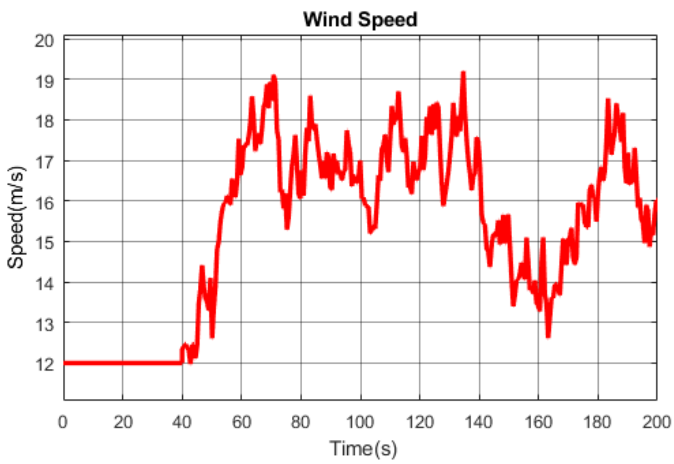


Figure 4.1 Wind Speed.

Figure (4.2) shows the rotor speed at the end of the simulation with each optimisation method, and Figure (4.3), (4.4), and (4.5) shows the values of the individual pitch angles modified by the PI, GA, and PSO controllers. It can be seen in figure 4.2 that the PSO curve reaches the optimum rotor speed faster than PI and GA, which take 40 seconds to reach the optimum speed of 45 rpm. The GA method has less odulation than the other two.

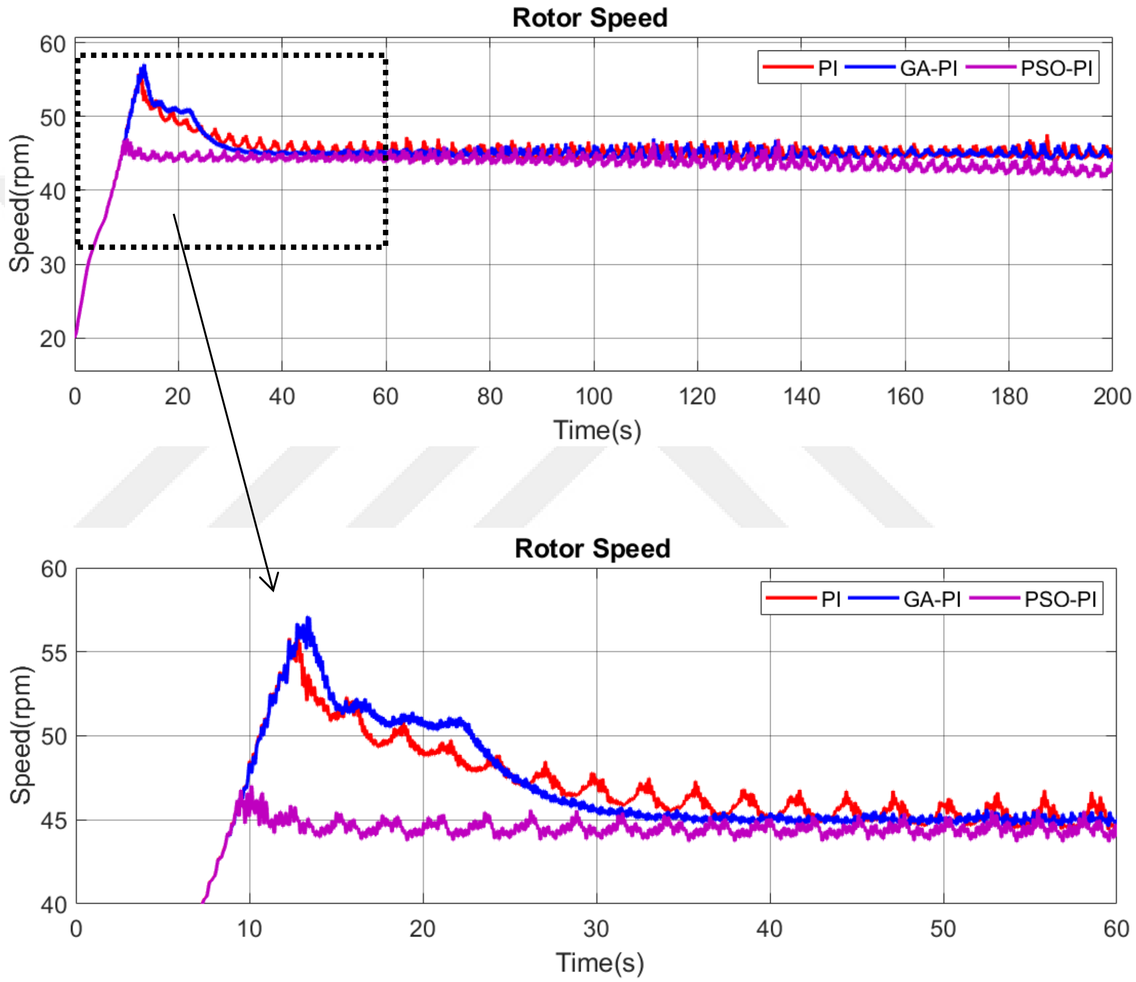


Figure 4.2 Rotor Speed.

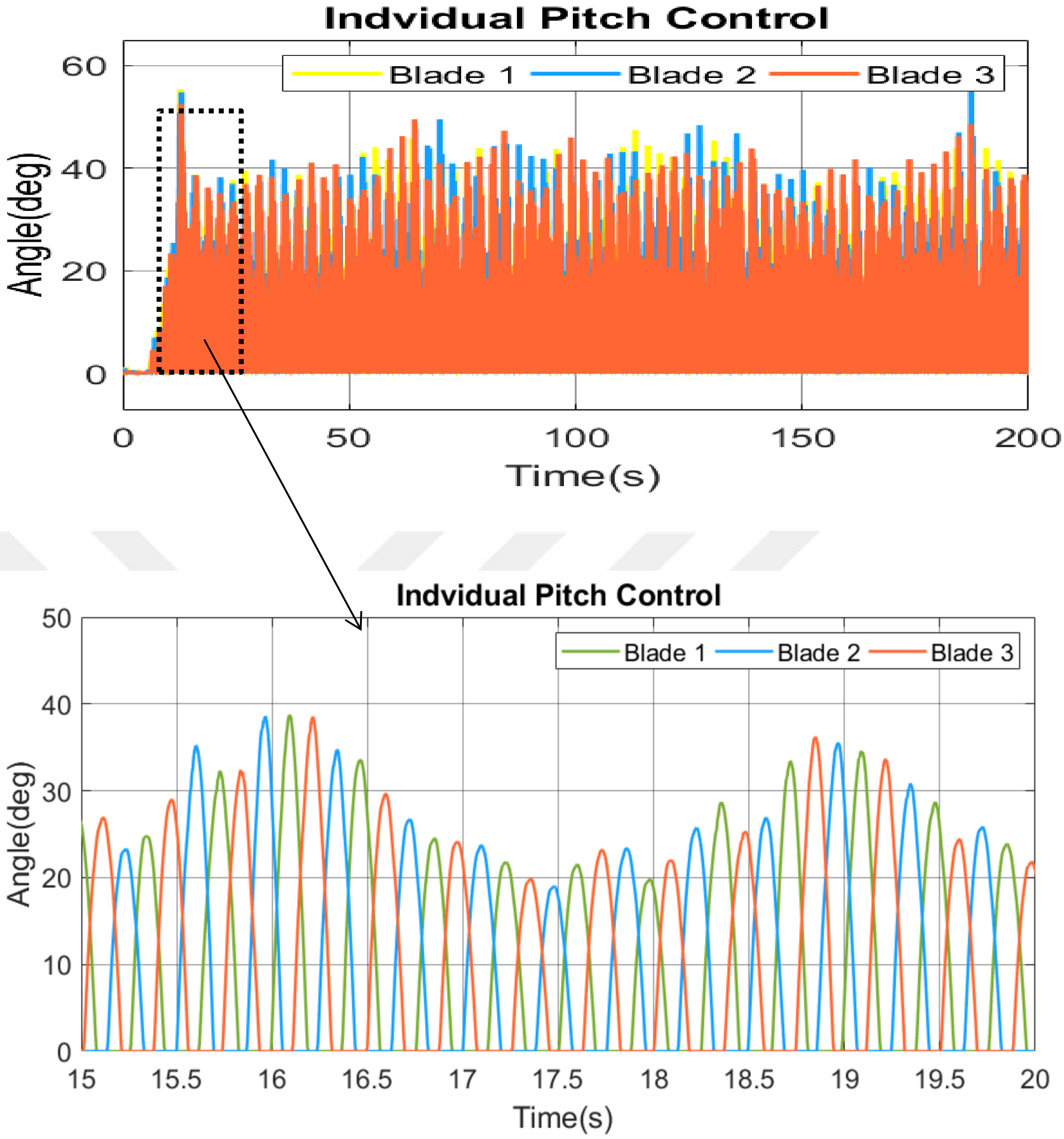


Figure 4.3 Individual Pitch Control PI.

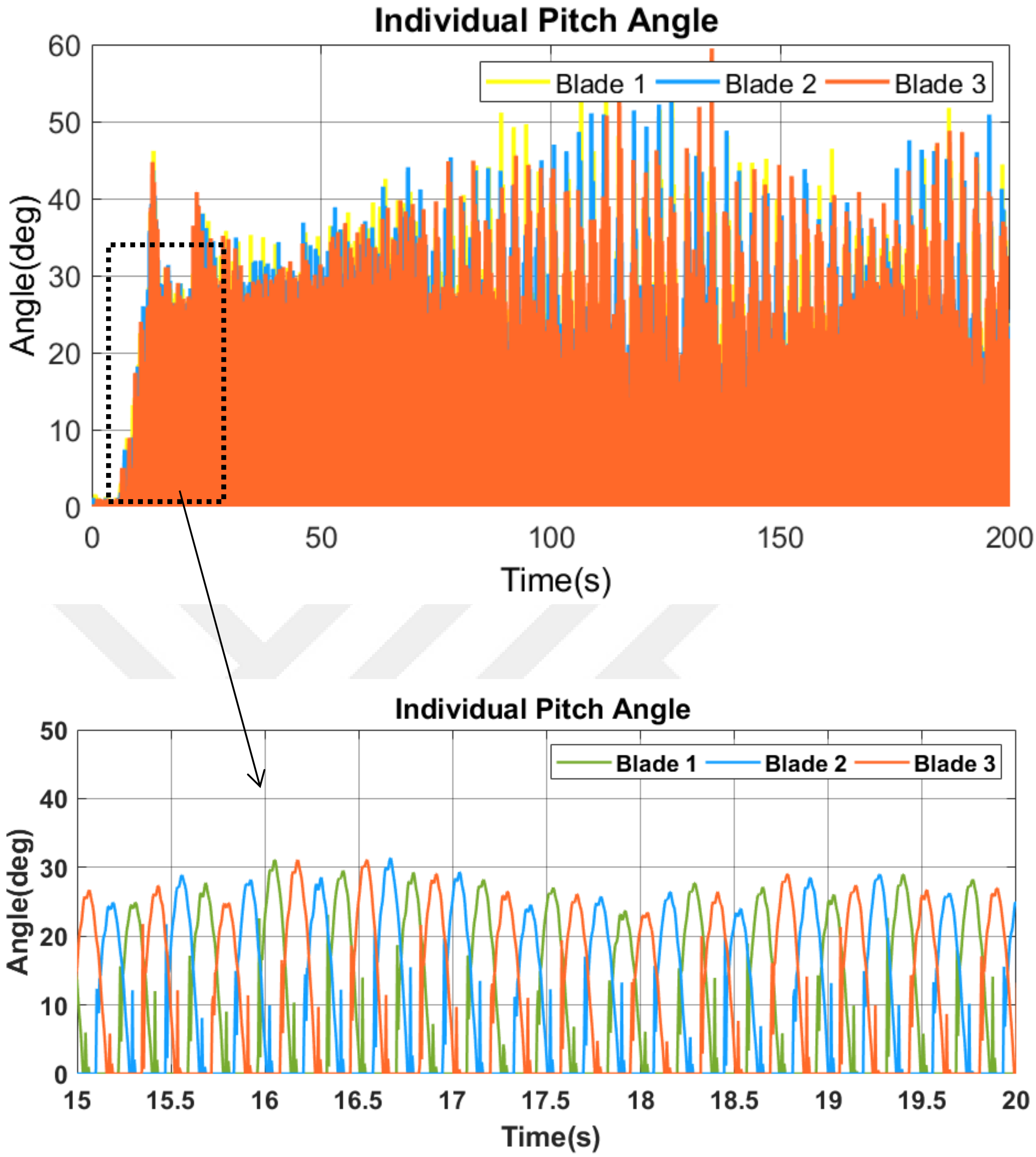


Figure 4.4 Individual Pitch Control GA.

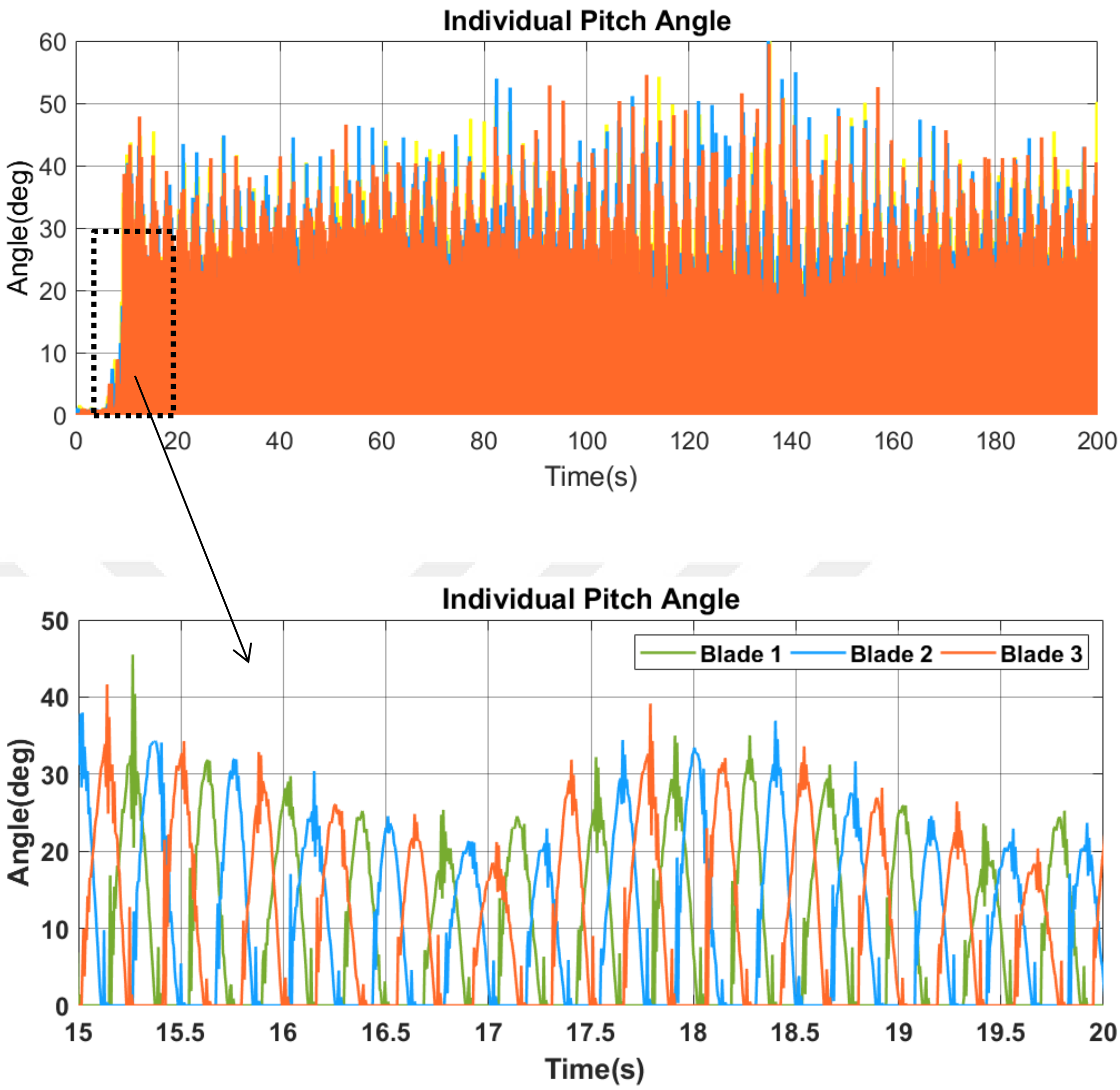


Figure 4.5 Individual Pitch Control PSO.

The individual pitch angle values in Figures (4.6), (4.7), and (4.8) each have three values for each blade of the wind turbine and have a phase difference of 120° between the individual pitch angles. To choose the best of the three control strategies, it is essential to understand how each blade differs from the others. The magnitude of the oscillation is proportional to the coefficient by which the RootMyc moment is multiplied. The optimal adjustment of this coefficient is a separate research topic. For each blade pitch angle, the variation in angles follows the change in rotor speed, so the PSO variation seems more optimal than the other two. But, as the GA has less ripple, that's why these simulations seem to be the most stable.

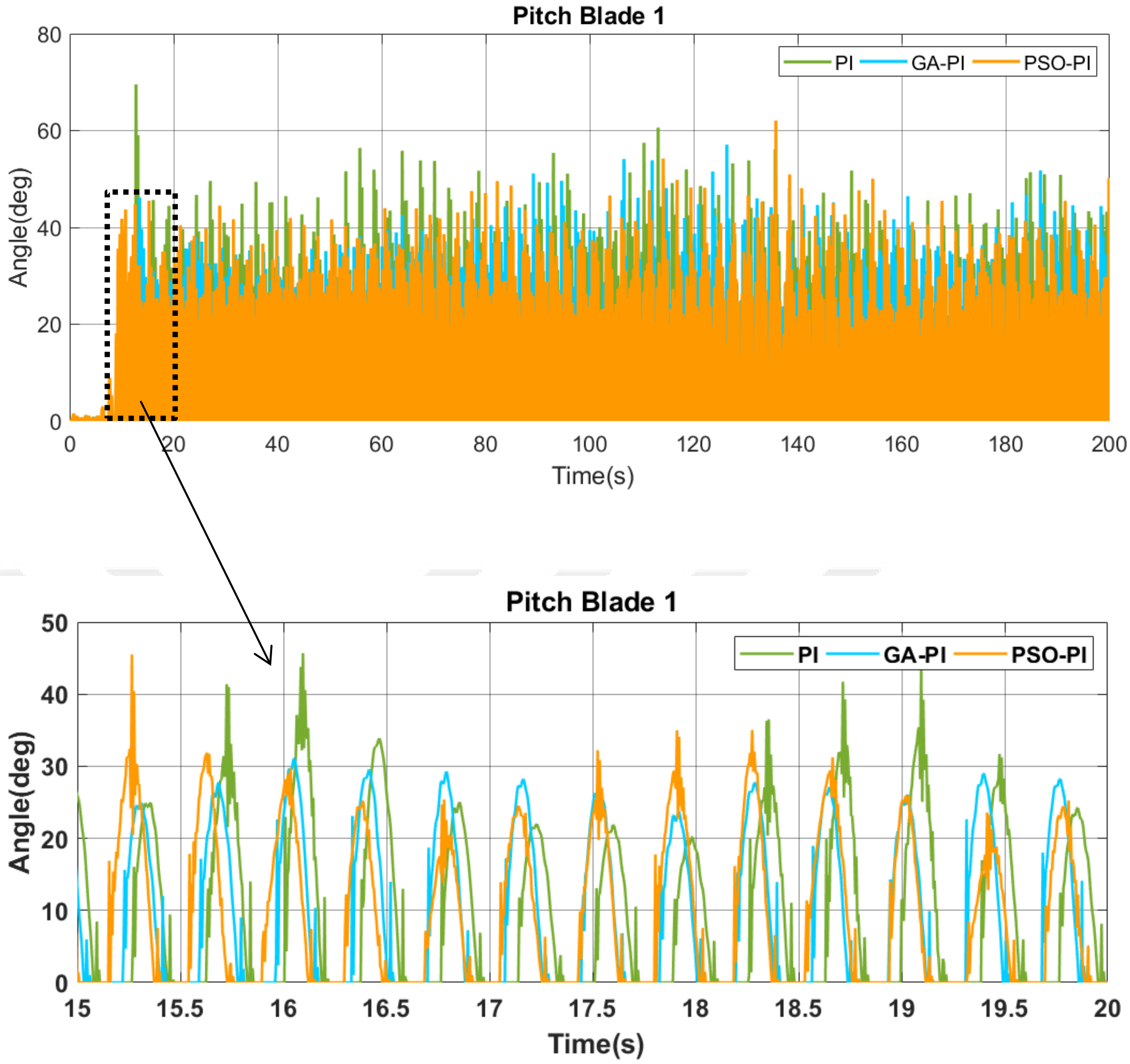


Figure 4.6 Pitch blade 1

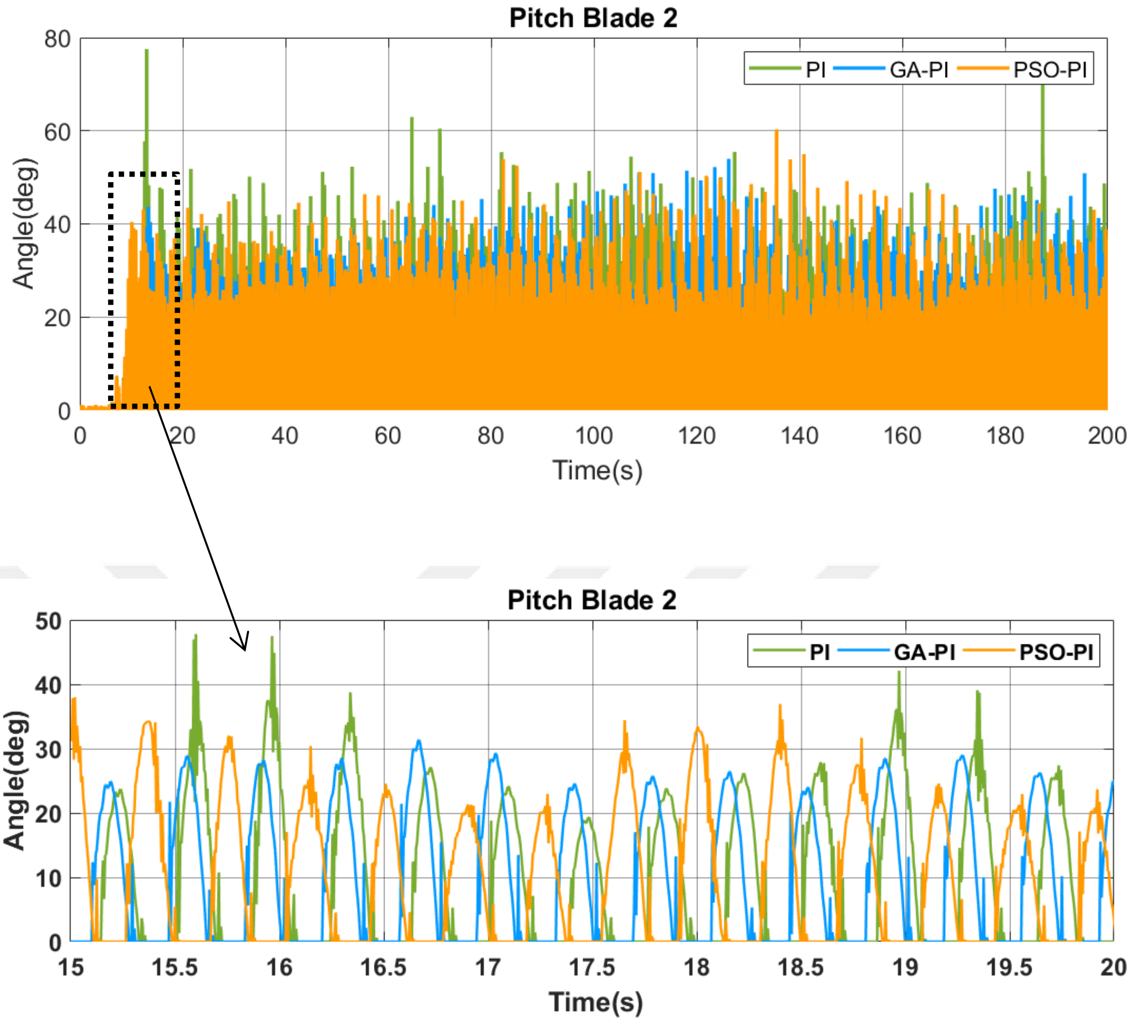


Figure 4.7 Pitch blade 2.

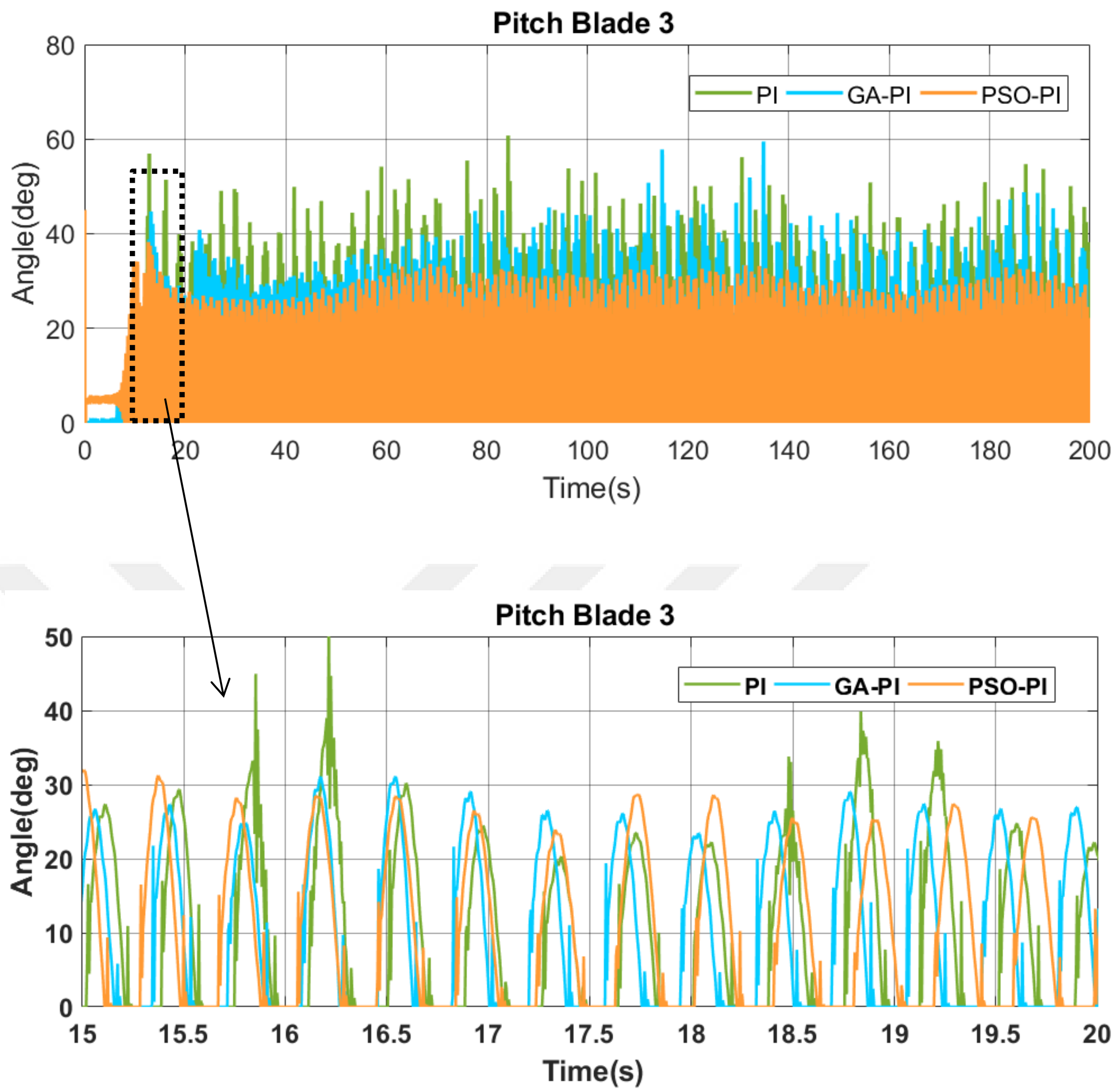


Figure 4.8 Pitch blade 3

For these figures (4.9), (4.10), (4.11), (4.12), and (4.13), the data from plane 1 of the wind turbine is used as an example. Regarding the pitch angle, we can see that for these figures, up to 9 seconds, all three methods take the same value, then the PSO takes a peak at around 10 seconds and then takes a signal that becomes stable as time goes by. We can see that the separation of the different control methods arrives at the moment when the rotor speed stops the peak it used to make before, while for the GA, it takes a long time to start and then takes a huge peak that varies according to the figure. Similarly, for the PI method, which is identical to the GA method, which, like the latter, stabilises over time.

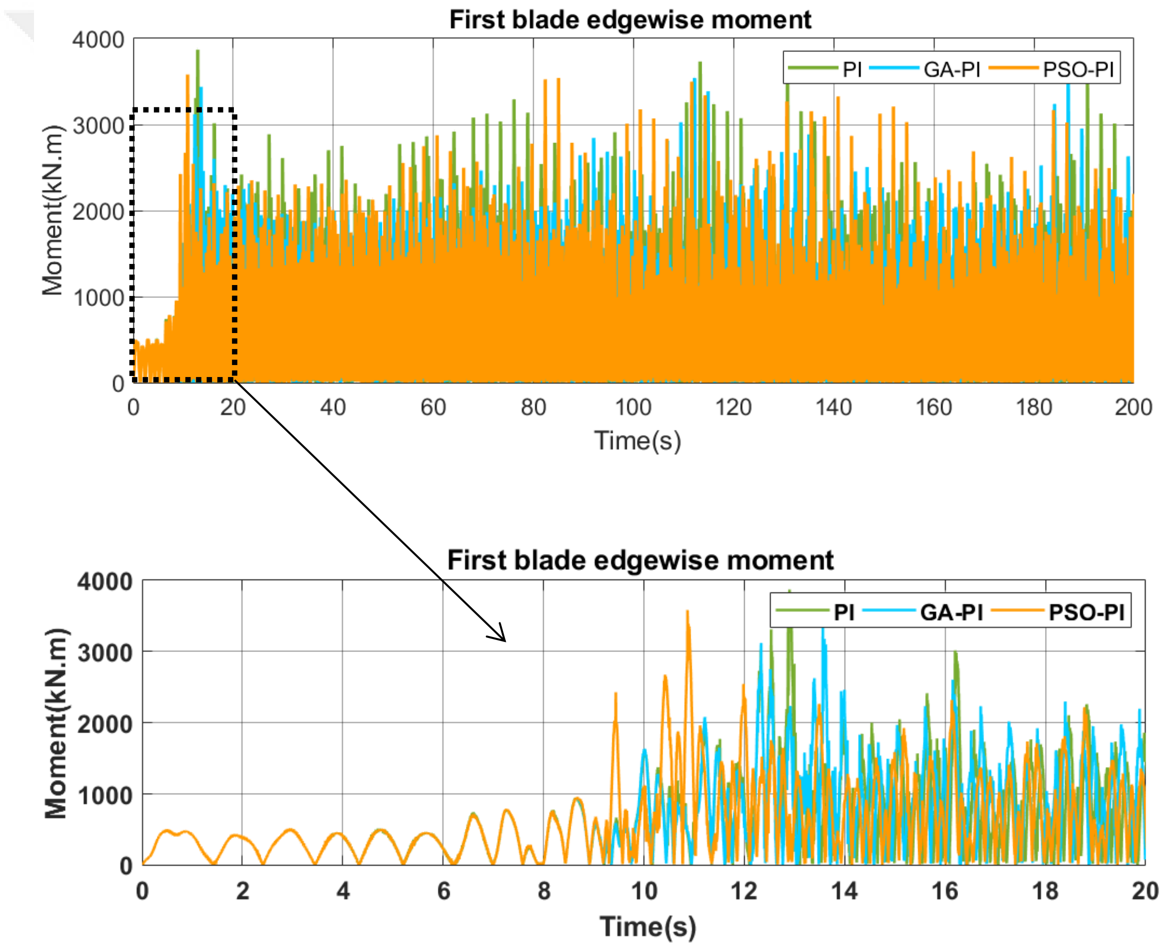


Figure 4.9 First blade edgewise moment.

In figure (4.10), we before the flapwise on the same results as in the previous figure. Flapwise bending moments (RootMyb) were calculated for a teetering rotor blade using a reasonably rapid theoretical method in which airloads obtained from wind-tunnel tests were employed.

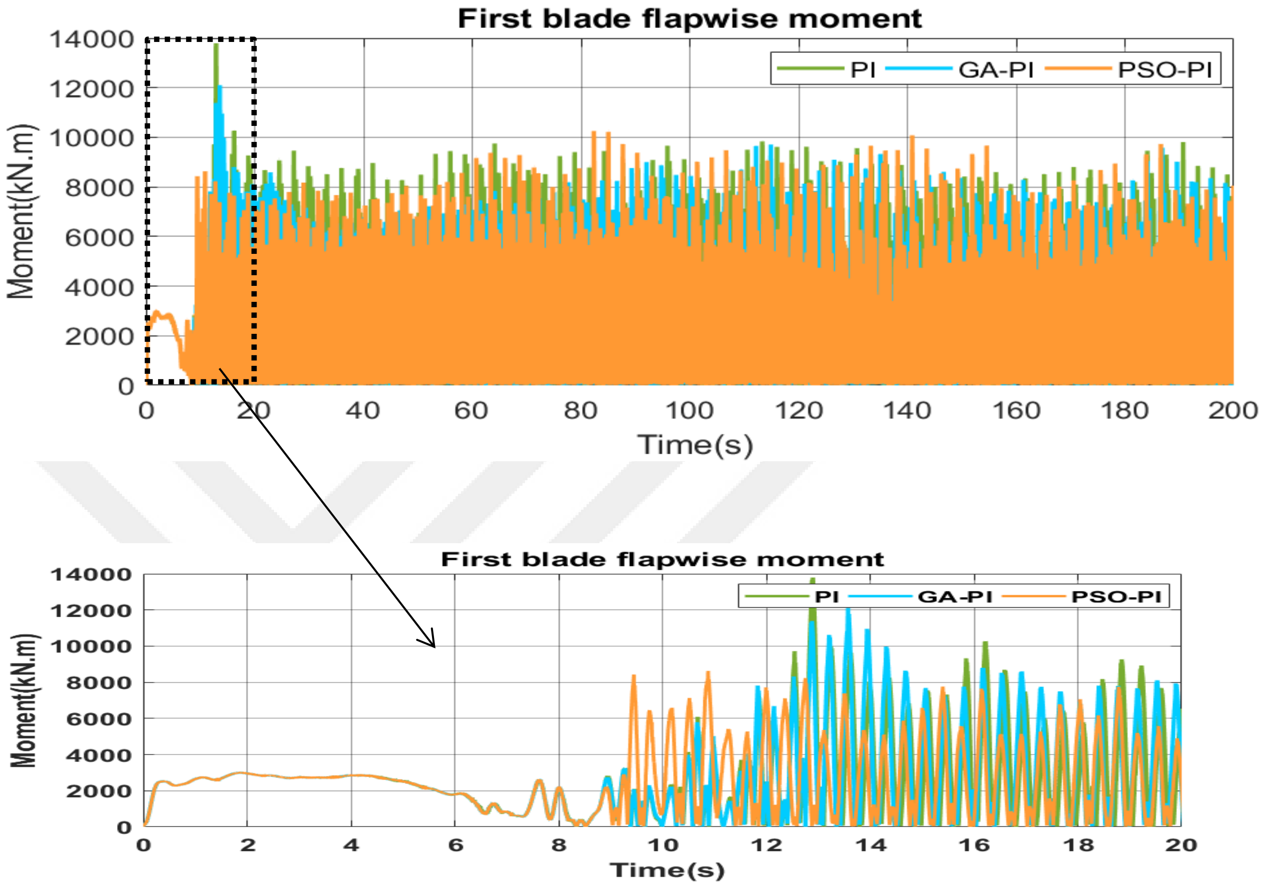


Figure 4.10 First blade flapwise moment.

Figure (4.11) shows the pitching moment before the similar output of the previous figure. In aerodynamics, the pitching moment (RootMzb) on an airfoil is the moment produced by the aerodynamic force on the airfoil if that aerodynamic force is considered to be applied not at the center of pressure but at the aerodynamic center of the airfoil.

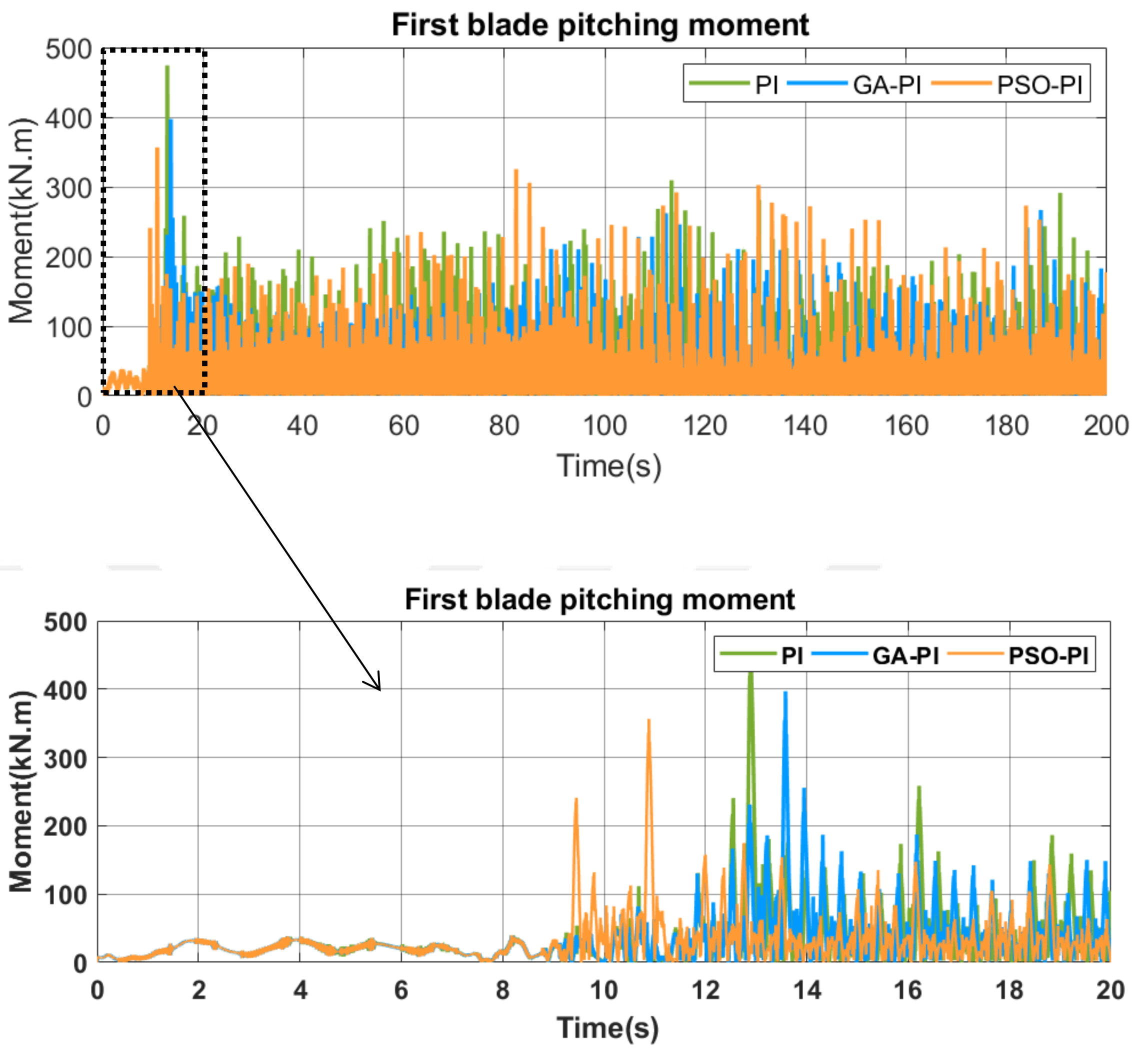


Figure 4.11 First blade pitching moment.

Figure (4.12) shows the out-of-plane moment, which has the same output as in the preceding fig. Out of plane bending moments (RootMyc) are those which are caused by out of plane forces, such as a building slab.

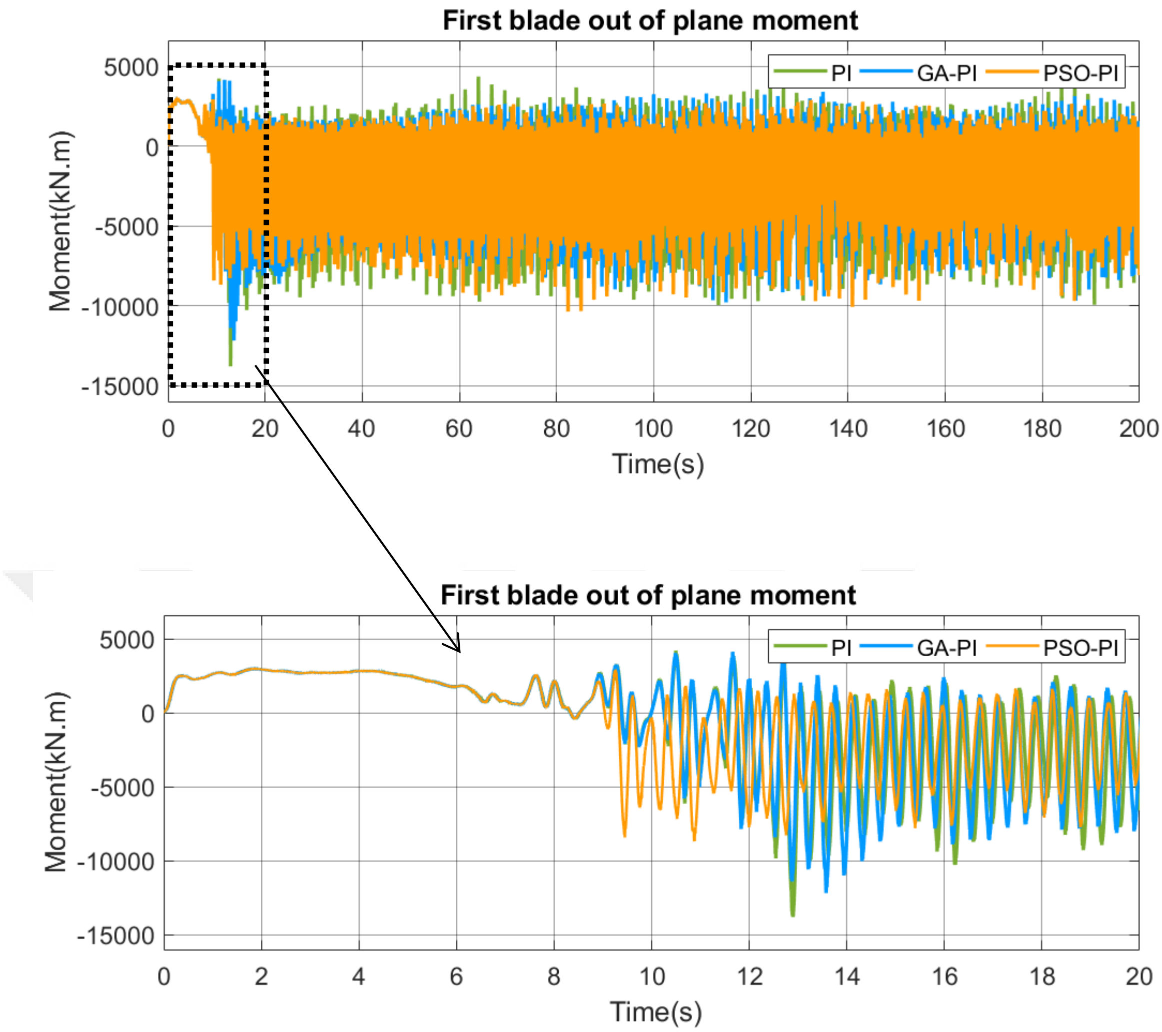


Figure 4.12 First blade out of plane moment.

Figure (4.13) shows the in-of-plane moment, which has the same output as in the preceding fig. In of plane bending moments (RootMxc) are those which are caused by in of plane forces, such as a building slab.

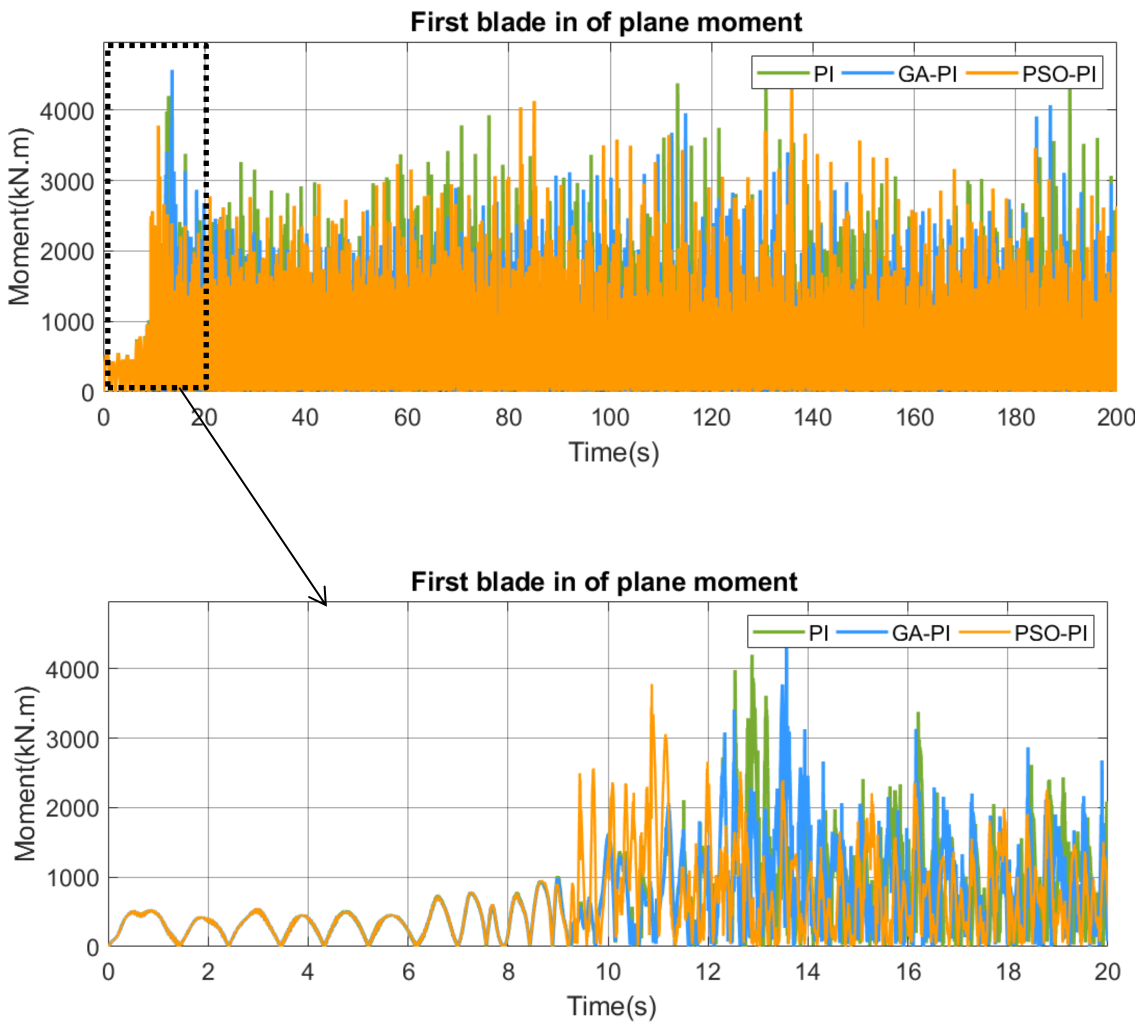


Figure 4.13 First blade in plane moment.

This phase difference is due to the phase difference of the blade moments. Figure (4.1) shows the applied wind speed. As the nominal wind speed is assumed to be 12 m/s, the average wind speed is chosen above the nominal speed to show the effect of the controller. The wind speed is applied along the x-axis, perpendicular to the turbine, since it is applied for balanced loads. Figure (4.13) shows the in-of-plane moment of the first wing at the wing root. As can be seen in Figure (4.13), the individual control of the pitch angle has reduced the mechanical loads. Table (4.1) shows the values and reduction rates of the mechanical load between the PI, PSO, and GA controls. For example, if we look at the first row of Table (4.1), we can see that the average value of the flanking moment at the root of the 1st wing, denoted by RootMxb1, is 133.95356 kNm for the PI controller, 193.876739 for the PSO controller and 135.780444 kNm for the GA controller.

Table 4.1 Moment results of individual pitch angles with different controls are compared.

Moment Measurement	Individual pitch angle (average moment)		
	PI	PSO	GA
RootMxb1	133.95356	193.876739	135.780444
RootMyb1	-2032.05638	-2259.43321	-2104.71947
RootMzb1	6.69044711	7.30673	6.45872691
RootMxc1	14.1493801	13.6005646	16.2038511
RootMyc1	-1973.34693	-2211.40841	-2047.1397
RootMzc1	6.69044711	7.30673	6.45872691
RootMxc2	16.4892063	14.6235413	11.8308178
RootMyc2	-1976.70951	-2213.04534	-2037.80413
RootMzc2	6.78244133	7.38748383	6.37130644
RootMxb2	135.746938	193.815903	132.804979
RootMyb2	-2036.76938	-2260.81537	-2095.49509
RootMzb2	6.78244133	7.38748383	6.37130644
RootMxc3	12.0307229	15.4466927	14.756165
RootMyc3	-1968.25904	-2215.90503	-2046.55032
RootMzc3	6.59856044	7.4552228	6.44961312
RootMxb3	130.559511	193.660961	134.702561
RootMyb3	-2027.03279	-2262.47148	-2105.20852
RootMzb3	6.59856044	7.4552228	6.44961312

Table 4.2 Moment descriptions.

Moment Measurements	Description
RootMxb1	1. lateral-edgewise moment at the wing root of the wing
RootMyb1	1. pitch-flapwise moment of the wing at the wing root
RootMzb1	1. Pitch moment of the wing at the wing root
RootMxc1	1. plane moment in plane at the wing root of the wing
RootMyc1	1.out-of-plane of plane at the wing root of the wing moment
RootMxc2	2. plane moment of the wing at the wing root
RootMyc2	1.out-of-plane of plane at the wing root of the wing

	moment
RootMzc2	2. Pitch moment of the 2nd wing at the wing root
RootMxb2	1.laterally-edgewise at the wing root of the wing moment
RootMyb2	2. pitch-flapwise at the wing root of the wing moment
RootMxc3	3. plane moment of the wing at the wing root
RootMyc3	3. out-of-plane of plane at the wing root of the wing moment
RootMzc3	3. Pitch moment of the 2nd wing at the wing root
RootMxb3	3. laterally-edgewise at the wing root of the wing moment
RootMyb3	3. pitch-flapwise at the wing root of the wing moment

5 CONCLUSIONS AND RECOMMENDATION

The analysis of wind turbines in this thesis was followed by a discussion of the various types of wind turbines and their parts. The horizontal axis wind turbine and the vertical axis wind turbine are the two different types of wind turbines, as was discussed in the second section. both wind turbines with horizontal axes and those with vertical axes. The tower, foundation, rotor, and blades are only a handful of the parts that make up a wind turbine. Next, we discussed the FAST system's overall purpose, as this sophisticated technology was developed by NREL to run simulations involving wind turbines and other things. The many forms of control are discussed and explained one by one in the third section. The final section examined the FAST wind turbines to test the viability of their potential and effectiveness. To provide a realistic environment for the turbine to be exposed to, the wind speed used in the experiment is the same as that used by FAST in turbulence. The MATLAB/SIMULINK simulations allowed us to clearly distinguish between the three control methods used to optimise the FAST wind turbine; the PI N-Z method, the PSO-PI method and the GA-PI method produced results that were closer to each other. The PSO-PI method, which proved to be beneficial for the optimisation of all three blades of the turbine, is the most effective approach, in my opinion. Controlling the PI and GA methods as a function of rotor speed gave better results, although the PSO method quickly peaked at speed, but unfortunately did not stabilise the blade angle. The GA-PI approach was one of the best as it stabilised the rotor speed as well as synchronised the tilt angles with the FAST system in the simulation.Making the simulation realistic and demonstrating consistency between the results of the various control techniques applied to each specific pitch angle are the goals here. To lessen some of the mechanical demands on the wind turbine, individual adjustment of pitch angle can be suggested as a solution. The study's analysis of mechanical loads found that they are largely balanced. The winds that blow perpendicular to the turbine blades on the x-axis provide these loads. The pitch angles of the turbine blades are individually regulated along with the mechanical loads on the turbine, and the low frequency components of these loads are decreased. As a result, it is possible to reduce mechanical wind turbine part fatigue. According to the study in Table 1, the mechanical load for all three blades was reduced by an average of 44%

when compared to the PI and PSO methods and by 1% when compared to the PI and GA methods. The importance of the system's individual pitch control becomes apparent at this speed. As a result, the study's significance is also made clear. Future research will also use various control methods, which will boost sensitivity. The simulation study will also simulate the features of an actual wind turbine.

These works are recommended for future works in light of the research done for this thesis:Try different control methods to see if you can get better optimization, example: Grey Wolf Optimization, Bacteria Foraging Algorithm and etc. Use these control methods with other types of ‘Test’ of the FAST system. To compare the individual and collective angles to see if the differences were so obvious.

REFERENCES

Ahmad, R., and Abdul-Hussain, M. 2017. Modeling and simulation of wind turbine generator using Matlab-Simulink. *Journal of Al-Rafidain University College For Sciences Print ISSN: 1681-6870, Online ISSN: 2790-2293*, 2, 282-300.

Beaudet, L. 2014. Etude expérimentale et numérique du décrochage dynamique sur une éolienne à axe vertical de forte solidité. PhD Thesis, Poitiers University, 313 pages, France.

Beltran, B., Ahmed-Ali, T. and Benbouzid, M. E. H. 2008. High-order sliding-mode control of variable-speed wind turbines. *IEEE Transactions on Industrial electronics*, *56*(9), 3314-3321.

Bossanyi, E. A. 2005. Further load reductions with individual pitch control. *Wind Energy: An International Journal for Progress and Applications in Wind Power Conversion Technology*, *8*(4), 481-485.

Camblong, H., Nourdine, S., Vechiu, I., and Tapia, G. 2012. Comparison of an island wind turbine collective and individual pitch LQG controllers designed to alleviate fatigue loads. *IET Renewable Power Generation*,*6*(4), 267-275.

Civelek, Z. 2017. Rüzgâr türbinlerinde bireysel hatve açısının PID, bulanık PID ve genetik bulanık denetleyiciler ile kontrol edilerek elektriksel güç çıkışı ve momentin iyileştirilmesi PhD Thesis, Kırıkkale Üniversitesi, 163 pages, Turquie.

Dubé, F. 2014. Conception et comparaison des performances de stratégies de commande PI et DAC appliquées au calage variable d'une éolienne de 10 kw. PhD.Thesis, Québec University, 281 pages, Québec.

Ehlers, J. 2009. Le choix des capteurs et l'estimation des états pour la commande individuelle du calage des pales. PhD Thesis, Québec University, 135 pages, Rimouski.

El-Baklish, S. K., El-Badawy, A. A., Frison, G., and Diehl, M. 2020. Nonlinear model predictive pitch control of aero-elastic wind turbine blades. *Renewable Energy*, *161*, 777-791.

Galvani, P. A., Sun, F., and Turkoglu, K. 2016. Aerodynamic modeling of NREL 5-MW wind Turbine for nonlinear control system design: A case study based on real-time nonlinear receding horizon control.*Aerospace*,*3*(3), 27.

GirirajKumar, S. M., Jayaraj, D., and Kishan, A. R. 2010. PSO based tuning of a PID controller for a high performance drilling machine. *International Journal of Computer Applications*, *1* 19, 12-18.

Han, Y., and Leithead, W. E. 2015. Comparison of Individual Pitch Control and Individual Blade Control for Wind Turbine Load Reduction.

Hau, E. 2013. Wind turbines: fundamentals, technologies, application, economics. Springer Science & Business Media, 871 page, London.

Hemeida, A. M., Farag, W. A., and Mahgoub, O. A. 2011. Modeling and control of direct driven PMSG for ultra large wind turbines. *International Journal of Energy and Power Engineering*,*5*(11), 1269-1275.

Hwas, A., and Katebi, R. 2012. Wind turbine control using PI pitch angle controller. *ifac proceedings volumes*,*45*(3), 241-246.

Iordanov, S. G. 2016. Self-Optimizing control for wind turbine generators. MSc. Thesis, Cranfield University, 82 pages, UK.

Jayachitra, A., and Vinodha, R. 2014. Genetic Algorithm Based PID Controller Tuning Approach for Continuous Stirred Tank Reactor. *Advances in Artificial Intelligence 16877470*.

Jonkman, J. 2013. The new modularization framework for the FAST wind turbine CAE tool. In 51st AIAA aerospace sciences meeting including the new horizons forum and aerospace exposition p. 202.

Jonkman, J. M., and Buhl, M. L. 2005. *FAST user's guide* (Vol. 365, p. 366). Golden, CO, USA: National Renewable Energy Laboratory.

Jonkman, J., Butterfield, S., Musial, W., and Scott, G. 2009. Definition of a 5-MW reference wind turbine for offshore system development No. NREL/TP-500-38060. National Renewable Energy Lab.(NREL), Golden, CO United States.

Kumara, E. A. D., Hettiarachchi, N., and Jayathilake, R. 2017. Overview of the vertical axis wind turbines. *Int. J. Sci. Res. Innov. Technol*, *4*, 56-67.

Magar, K. T., Balas, M. J., and Frost, S. 2015. Direct adaptive control for individual blade pitch control of wind turbines for load reduction. *Journal of Intelligent Material Systems and Structures*,*26*(12), 1564-1572.

Manyonge, A. W., Ochieng, R. M., Onyango, F. N., and Shichikha, J. M. 2012. Mathematical modelling of wind turbine in a wind energy conversion system: Power coefficient analysis. *Applied Mathematical Sciences.*

Mulders, S. P., Zaaijer, M. B., Bos, R., and Van Wingerden, J. W. 2020. Wind turbine control: Open-source software for control education, standardization and compilation. In *Journal of Physics: Conference Series* Vol. 1452, No. 1, p. 012010. IOP Publishing.

Namik, H., and Stol, K. 2010. Individual blade pitch control of floating offshore wind turbines. *Wind Energy: An International Journal for Progress and Applications in Wind Power Conversion Technology*, *13* 1, 74-85.

Namik, H., Stol, K., and Jonkman, J. 2008,. State-space control of tower motion for deepwater floating offshore wind turbines. In *46th AIAA Aerospace Sciences Meeting and Exhibit*(p. 1307).

Omer, A. M. 2008. On the wind energy resources of Sudan. *Renewable and Sustainable Energy Reviews*, *12*(8), 2117-2139.

Pai, N. S., Chang, S. C., and Huang, C. T. 2010. Tuning PI/PID controllers for integrating processes with deadtime and inverse response by simple calculations. *Journal of Process Control*, *20* 6, 726-733.

Pettas, V., Salari, M., Schlipf, D., and Cheng, P. W. 2018. Investigation on the potential of individual blade control for lifetime extension. In *Journal of Physics: Conference Series* (Vol. 1037, No. 3, p. 032006). IOP Publishing.

Piasecka, I., Tomporowski, A., Flizikowski, J., Kruszelnicka, W., Kasner, R., and Mroziński, A. 2019. Life cycle analysis of ecological impacts of an offshore and a land-based wind power plant. *Applied Sciences*, *9*(2), 231.

Ragheb, M. 2011. Vertical axis wind turbines. *University of Illinois at Urbana-Champaign*, *1*, 40.

Sanchez, H., Escobet, T., Puig, V., and Odgaard, P. F. 2015. Health-aware model predictive control of wind turbines using fatigue prognosis. *IFAC-PapersOnLine*, *48* 21, 1363-1368.

Sarkar, M. R., Julai, S., Tong, C. W., Uddin, M., Romlie, M. F., and Shafiullah, G. M. 2020. Hybrid pitch angle controller approaches for stable wind turbine power under variable wind speed. *Energies*, *13*(14), 3622.

Sarkar, S., Fitzgerald, B., and Basu, B. 2020. Nonlinear model predictive control to reduce pitch actuation of floating offshore wind turbines. *IFAC-PapersOnLine*,*53*(2), 12783-12788.

Sebit, K., Alpman, E., and Yılmaz, B. 2014. Computational studies of horizontal axis wind turbines using advanced turbulence models. *Marmara Fen Bilimleri Dergisi*, *26* 2, 30-40.

Sharpe, T. 2010. The Role of Aesthetics, Visual and Physical Integration in Building Mounted Wind Turbines An Alternative Approach, In: Paths to Sustainable Energy, Nathwani, J. and Ng, A. (eds), Intech, pp. 279-300, Croatia.

Silpa Baburajan, D., and Ismail, A. 2017. Design and Control of the Pitch of Wind Turbine through PID. International Research Journal of Engineering and Technology, 2395-0056.

Skaare, B., Hörnsten, B., and Nielsen, F. G. 2013. Modeling, simulation and control of a wind turbine with a hydraulic transmission system. *Wind Energy*, *16*(8), 1259-1276.

Sukmono, F, 2019. Optimization control system in units 3 and 4, ulubelu geothermal field. Report, United nations University, 22 pages, Indonesia.

Sylla, A. M. 2013. Modélisation d'un émulateur éolien à base de machine asynchrone à double alimentation. PhD Thesis, Québec University, 137 pages, Québec.

Tran, T. T., and Kim, D. H. 2015. The coupled dynamic response computation for a semi-submersible platform of floating offshore wind turbine. *Journal of wind engineering and industrial aerodynamics*, *147*, 104-119.

Vyas, M., Singh, M., and Santoso, S. 2011. Handbook of Wind Power Systems, ch. Grid Integration of Wind Power Systems: Modeling of Wind Power Plants.

Yuan, Y., and Tang, J. 2017. Adaptive pitch control of wind turbine for load mitigation under structural uncertainties. *Renewable Energy*,*105*, 483-494.

APPENDICES

APPENDIX 1. The MatLab simulations of the NREL 1.5MW wind turbine used the FAST v8 input files.

APPENDIX 2. Moment force in FAST system

APPENDIX 3. Wind turbine and generator parameters

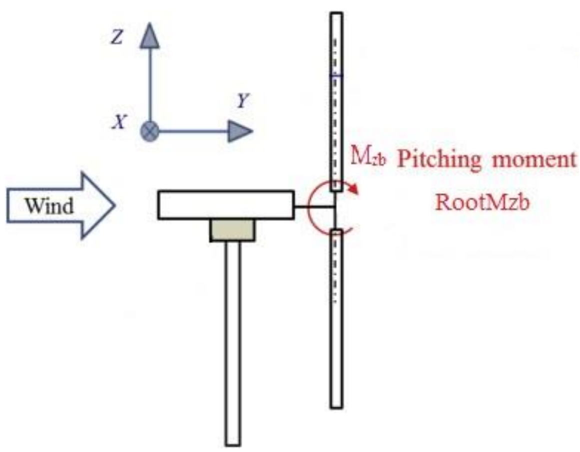
APPENDIX 4. GA Parameter

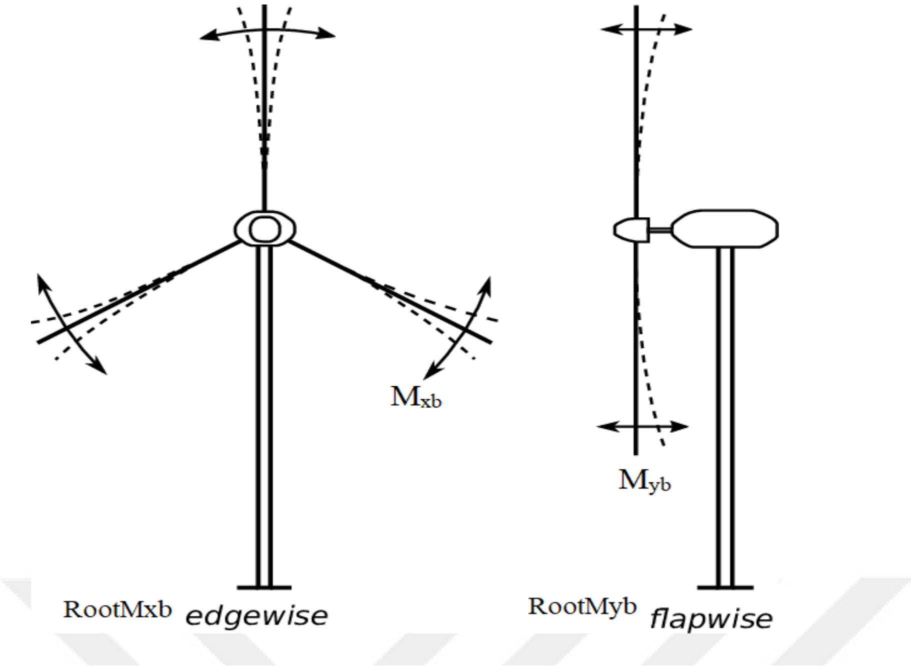
APPENDIX 5. PSO Parameter

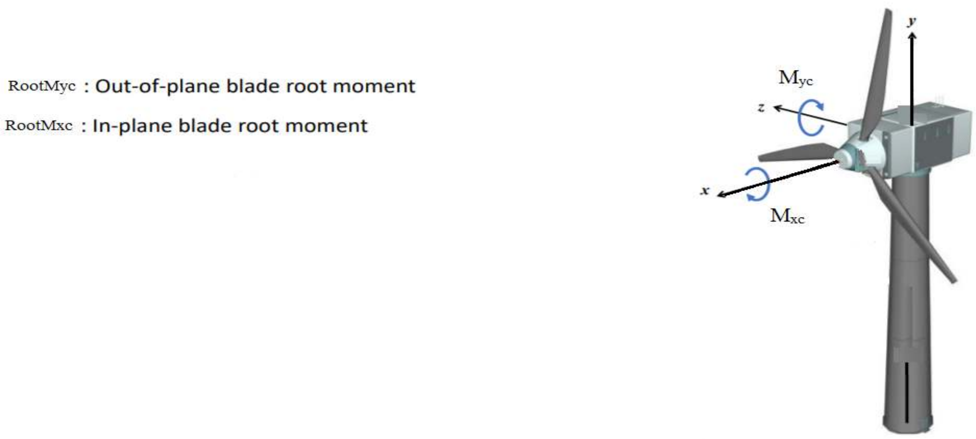
APPENDIX 1. The MatLab simulations of the NREL 1.5MW wind turbine used the FAST v8 input files

```
------- FAST v8.10.* INPUT FILE ------------------------------------------------
FAST Certification Test #11: WindPACT 1.5 MW Baseline with many DOFs undergoing a pitch failure.
--------------------- SIMULATION CONTROL ------------------------------------
False         Echo            - Echo input data to <RootName>.ech (flag)
"FATAL"       AbortLevel      - Error level when simulation should abort (string) {"WARNING", "SEVERE", "FATAL"}
         50   TMax            - Total run time (s)
      0.005   DT              - Recommended module time step (s)
          2   InterpOrder     - Interpolation order for input/output time history (-) {1=linear, 2=quadratic}
          0   NumCrctn        - Number of correction iterations (-) {0=explicit calculation, i.e., no corrections}
      99999   DT_UJac         - Time between calls to get Jacobians (s)
      1E+06   UJacSclFact     - Scaling factor used in Jacobians (-)
--------------------- FEATURE SWITCHES AND FLAGS ----------------------------
          1   CompElast       - Compute structural dynamics (switch) {1=ElastoDyn; 2=ElastoDyn + BeamDyn for blades}
          1   CompAero        - Compute aerodynamic loads (switch) {0=None; 1=AeroDyn}
          1   CompServo       - Compute control and electrical-drive dynamics (switch) {0=None; 1=ServoDyn}
          0   CompHydro       - Compute hydrodynamic loads (switch) {0=None; 1=HydroDyn}
          0   CompSub         - Compute sub-structural dynamics (switch) {0=None; 1=SubDyn}
          0   CompMooring     - Compute mooring system (switch) {0=None; 1=MAP++; 2=FEAMooring; 3=MoorDyn}
          0   CompIce         - Compute ice loads (switch) {0=None; 1=IceFloe; 2=IceDyn}
False         CompUserPtfmLd  - Compute additional platform loading (flag) {false: none; true: user-defined from routine UserPtfmLd}
False         CompUserTwrLd   - Compute additional tower loading (flag) {false: none; true: user-defined from routine UserTwrLd}
--------------------- INPUT FILES ---------------------------------------------
"WP_Baseline/Test11_ElastoDyn.dat"    EDFile          - Name of file containing ElastoDyn input parameters (quoted string)
"unused"      BDBldFile(1)    - Name of file containing BeamDyn input parameters for blade 1 (quoted string)
"unused"      BDBldFile(2)    - Name of file containing BeamDyn input parameters for blade 2 (quoted string)
"unused"      BDBldFile(3)    - Name of file containing BeamDyn input parameters for blade 3 (quoted string)
"WP_Baseline/Test11_AD.ipt"    AeroFile        - Name of file containing aerodynamic input parameters (quoted string)
"WP_Baseline/Test11_ServoDyn.dat"    ServoFile       - Name of file containing control and electrical-drive input parameters (quoted string)
"unused"      HydroFile       - Name of file containing hydrodynamic input parameters (quoted string)
"unused"      SubFile         - Name of file containing sub-structural input parameters (quoted string)
"unused"      MooringFile     - Name of file containing mooring system input parameters (quoted string)
"unused"      IceFile         - Name of file containing ice input parameters (quoted string)
--------------------- OUTPUT --------------------------------------------------
True          SumPrint        - Print summary data to "<RootName>.sum" (flag)
          1   SttsTime        - Amount of time between screen status messages (s)
      0.010   DT_Out          - Time step for tabular output (s)
          0   TStart          - Time to begin tabular output (s)
          1   OutFileFmt      - Format for tabular (time-marching) output file (switch) {1: text file [<RootName>.out], 2: binary file [<RootName>.outb], 3: both}
True          TabDelim        - Use tab delimiters in text tabular output file? (flag) {uses spaces if false}
"ES10.3E2"    OutFmt          - Format used for text tabular output, excluding the time channel.  Resulting field should be 10 characters. (quoted string)
```

APPENDIX 2. Moment force in FAST system





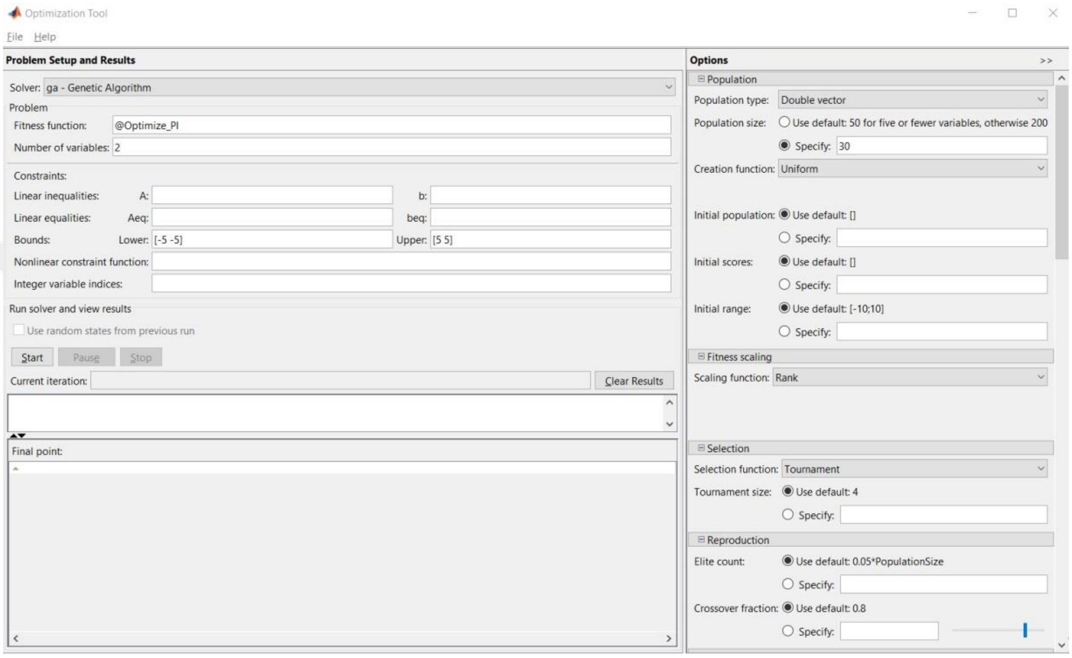


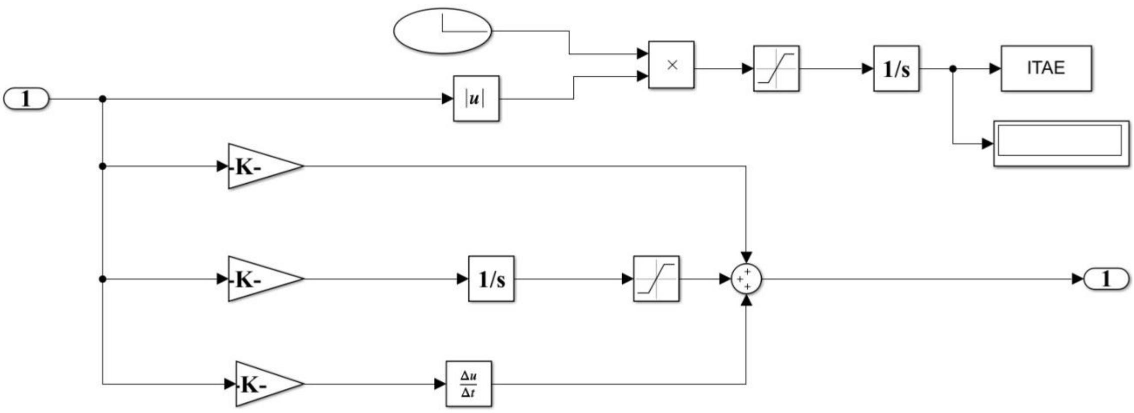
APPENDIX 3. Wind turbine parameters

Rated power of wind turbine	1.5	MW
Rotor diameter	70	m
Number of blade	3	
Hub height	84	m
Max/min pitch angle	90/0	Degree
Max pitch rate	9	Degree/s
Rate rotor speed	45	rpm
Gearbox ratio	1/40	
Cut-in/cut-out wind speed	3/25	m/s

APPENDIX 4. GA Parameter

```
function cost = optimize_PID(k)
assignin('base','k',k);
sim('formahdiGA.slx');
cost = ITAE(length(ITAE));
end
```





APPENDIX 5. PSO Parameter

```
%Swarm initialization:
n = 20;                      %Number of particles
dim =2;                      %Dimensions
```

```
steps=120;                  %Number of steps
c1=2.05;                    %Cognitive factor
c2=2.05;                    %Exploration factor
fi=c1+c2;
X=2/abs((2-fi-((fi^2)-(4*fi))^(1/2)));
%Constriction factor
rng('shuffle');                                        %Seed
initialization
Evap=1.5;
%Evaporation factor
velocity_clamping = 10;                                %Velocity
clamping
diversity_limit = 1;                                   %Diversity
limit
psi1=rand(2,1);
psi2=rand(2,1);                                        %Initial
random coefficients
swarmdir(1)=1;
swarmdir(2)=1;                                         %Initial
directions
position=(50*rand(dim,n));                             %Position
initialization
speed=10*rand(dim,n);                                  %Velocity
initialization
change1=35;                                            %Step of
first inertia change
change2=80;                                           %Step of
second inertia change
%%
%Initial values
Pbest=position;                                        %Matrix of
personal best positions
ObjFun=0*rand(n,1);                                    %Matrix of
objective function values
step=1;
%Calculation of initial Kp and Ki and ObjFun values:
for i=1:n
    Kp=Pbest(1,i);
    Ki=Pbest(2,i);
    ObjFun(i)=optimize1;                          %Optimize
functions is in another file
end
bestlocalObjFun=ObjFun;                                %Matrix of
best local objective function values
```

```
[globalObjFun,g]=min(bestlocalObjFun);       %Best
global objective function value
Gbest = Pbest(:,g) ;                         %Best
global position
speed(:,i)=X*(speed(:,i)+(fi/2)*swarmdir'.*(psi1.*(Pbe
st(:,i)-
position(:,i)))+(fi/2)*swarmdir'.*(psi2.*(Gbest-
position(:,i)))); %tablica predkosci
position(:,i)=position(:,i)+speed(:,i);
%
%%
%Initialization done
%Main loop:
for step=1:steps
 fprintf('Step %i \n', step);
 fprintf('Found optimum: \n Kp= %0.3f \n Ki= %0.3f
\n', Gbest(1,1),Gbest(2,1));

    %Inertia change conditions
    if step==change1
        J=J*5;
    end
    if step==change2
        J=J*0.2;
    end

    %Asynchronous update rule
    for i=1:n

         %Kp and Ki assignment and simulation
         Kp=position(1,i);
         Ki=position(2,i);
         ObjFun(i)=optimize(Kp,Ki);

        %Evaporation conditions
        if 0.01+ObjFun(i)>=bestlocalObjFun(i)*Evap

bestlocalObjFun(i)=Evap*bestlocalObjFun(i);
         else
           bestlocalObjFun(i)=ObjFun(i)+0.01;
           Pbest(:,i)=position(:,i);
        end

        %Gbest calculation
        [globalObjFun,g]=min(bestlocalObjFun);
        Gbest=Pbest(:,g);
```

```
        %Swarm diversity calculation
        swarm_div(1) = 0.5*(max(position(1,:))-
min(position(1,:)));
        swarm_div(2) = 0.5*(max(position(2,:))-
min(position(2,:)));

        %Direction change in case of too high
diversity
        for d=1:2,
                if swarm_div(d) < diversity_limit,
                    swarmdir(d) = -1;
                else
                    swarmdir(d) = 1;
                end
        end

        %Position and speed calculation
        psi1=rand(2,1);
        psi2=rand(2,1);

speed(:,i)=X*(speed(:,i)+(fi/2)*swarmdir'.*(psi1.*(Pbe
st(:,i)-
position(:,i)))+(fi/2)*swarmdir'.*(psi2.*(Gbest-
position(:,i))));
        speed(1,i)= min(max(-
velocity_clamping,speed(1,i)),velocity_clamping);
        speed(2,i)= min(max(-
velocity_clamping,speed(2,i)),velocity_clamping);
        position(:,i)=position(:,i)+speed(:,i);

        %Walls on Kp=0, Ki=0, Kpmax and Kimax
        if position(1,i)<0
         position(1,i) =  abs(position(1,i));
         speed(1,i)=abs(speed(1,i));
        end
        if position(2,i)<0
           position(2,i) = abs(position(2,i));
           speed(2,i) = abs(speed(2,i));
        end
        if position(1,i)>Kpmax
         position(1,i) = position(1,i)-
(2*(position(1,i)-Kpmax));
         speed(1,i)=-speed(1,i);
        end
```

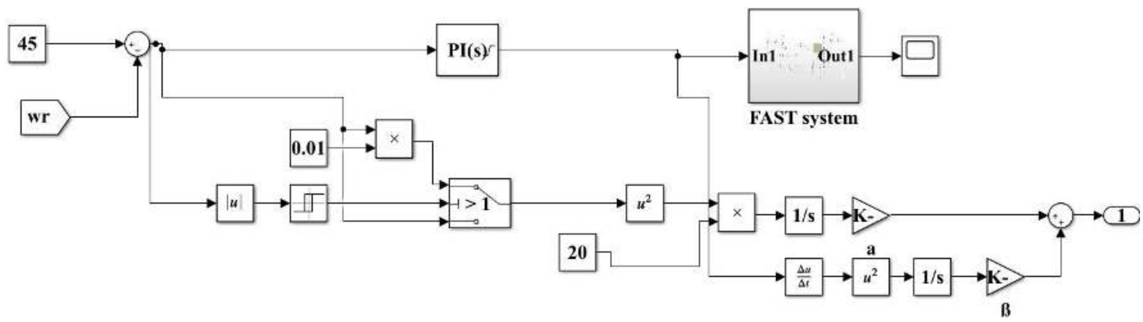
```
        if position(2,i)>Kimax
         position(2,i) =  position(2,i)-
(2*(position(2,i)-Kimax));
         speed(2,i)=-speed(2,i);
        end

    %History logging for plotting purposes
    History(1,step,:)=position(1,:);
    History(2,step,:)=position(2,:);
    speedHistory(1,step,:)=speed(1,:);
    speedHistory(2,step,:)=speed(2,:);
    GbestHistory(step,:)=Gbest;
    Gbestvalue(step,:)=globalObjFun;
    %figure(1);
    %for i=1:n
    %  subplot(2,1,1);
    %
scatter(log10(position(2,i)),log10(position(1,i)));
    %    title('Pozycje czastek');
    %    xlabel('log10(Ki)'); ylabel('log10(Kp)');
    %    axis([0 3 0 3]);
    %    hold on
    %end
  %  hold off;
    end
end

 Kp=Gbest(1,1);
 Ki=Gbest(2,1);
fprintf('Optimum found: \n Kp= %0.3f \n Ki= %0.3f \n',
Gbest(1,1),Gbest(2,1));
figure();
subplot(3,1,1);
scatter(1:steps, GbestHistory(:,1));
hold;
plot([change1-1, change1-1], [1,
max(GbestHistory(:,1))], 'LineWidth', 1, 'Color',
'blue');
plot([change2-1, change2-1], [1,
max(GbestHistory(:,1))], 'LineWidth', 1, 'Color',
'blue');
xlabel('iteration');
ylabel('Kp');
grid;
subplot(3,1,2);
```

```
scatter(1:steps, GbestHistory(:,2));
hold;
plot([change1-1, change1-1], [1,
max(GbestHistory(:,2))], 'LineWidth', 1, 'Color',
'blue');
plot([change2-1, change2-1], [1,
max(GbestHistory(:,2))], 'LineWidth', 1, 'Color',
'blue');
xlabel('iteration');
ylabel('Ki');
grid;
subplot(3,1,3);
scatter(1:steps, Gbestvalue);
hold;
plot([change1-1, change1-1], [1, max(Gbestvalue)],
'LineWidth', 1, 'Color', 'blue');
plot([change2-1, change2-1], [1, max(Gbestvalue)],
'LineWidth', 1, 'Color', 'blue');
xlabel('iteration');
ylabel('Gbest Value');
grid;


function OptimalPI = optimize1
data=sim('formahdipso.slx');
datan=max(get(data,'yout'));
%Objective function calculation - partially calculated
inside of the model
OptimalPI=datan(1)+((abs(datan(2))-1)^2)*30000;
end
```



CURRICULUM VITAE

Personal Information

Name and Surname : Mahdi OSMAN ABDI

Education

MSc	Çankırı Karatekin University Graduate School of Natural and Applied Sciences Department of Electrical and Electronics Engineering	2019-Present
Undergraduate	Djibouti University Faculty of Engineering Department of Mechanical Engineering	2014-2018

Work Experience

Year	Institution	Position
02-2017-03-2017	Electricity of Djibouti, Department of Mechanic	Traineer.
04-2018-07-2018	Marill Group, Department of Automotive	Agent field Ass.