Design and Analysis of an Artificial Intelligence-based Rotor Current Control Strategy for a Doubly Fed Induction Generator

Niguse Assefa Abebe

A Thesis Submitted to
The Department of Electrical Power and Control Engineering
School of Electrical Engineering and Computing

Presented in Partial Fulfillment of the Requirement for the Degree of Master's in
Electrical Power and Control Engineering (Control Engineering)

Office of Graduate Studies
Adama Science and Technology University

September 2022
Adama, Ethiopia

Design and Analysis of an Artificial Intelligence-based Rotor Current Control Strategy for a Doubly Fed Induction Generator

Niguse Assefa Abebe

Dr. Milkias Berhanu Tuka

A Thesis Proposal Submitted to

Department of Electrical Power and Control Engineering

School of Electrical Engineering and Computing

Presented in Partial Fulfillment of the Requirement for the Degree of Master's in Electrical Power and Control Engineering (Control Engineering)

Office of Graduate Studies

Adama Science and Technology University

September 2022

Adama, Ethiopia

DECLARATION

I hereby declare that this thesis entitled "Design and Analysis of Artificial Intelligence-based Controller for Rotor Current of Doubly fed Induction Generator" is my original work. That is, it has not been submitted for the award of any academic degree, diploma, or certificate in any other university. All sources of materials that are used for this thesis have been duly acknowledged through citation

Niguse Assefa	________________	________________
Name of student	Signature	Date

RECOMMENDATION

I, the major advisor of this thesis, hereby certify that I have read the revised version of the thesis entitled “Design and Analysis of Artificial Intelligence-based Controller for Rotor Current of Doubly fed Induction Generator” prepared under my guidance by Mr. Niguse Assefa submitted in partial fulfillment of the requirements for the degree of Master’s of Science in Control Engineering. Therefore, I recommend the submission of the revised version of the thesis to the department following the applicable procedures.

______________________ ______________ ___________________

Major Advisor Signature Date

APPROVAL PAGE

I hereby certify that the recommendation and suggestion given by the proposal review committee are appropriately incorporated into the final thesis proposal entitled "Design and Analysis of Artificial Intelligence-based Controller for Rotor Current of Doubly fed Induction Generator" by Mr. Niguse Assefa.

____________________	____________	_________________
Major Advisor	Signature	Date

We, the undersigned members of the Board of Examiners of the thesis open defense by Mr. Niguse Assefa have read and evaluated the thesis entitled "Design and Analysis of Artificial Intelligence-based Controller for Rotor Current of Doubly fed Induction Generator" and examined the candidate during the open defense.

___________________	___________	___________
Chairperson	Signature	Date
___________________	___________	___________
Internal Examiner 1	Signature	Date
___________________	___________	___________
Internal Examiner 2	Signature	Date
Dr. Tefera Kitaba	Tefera	October, 20/2022
External Examiner	Signature	Date

Finally, approval and acceptance of the thesis are contingent upon the submission of its final copy to the Office of Postgraduate Studies (OPGS) through the Department Graduate Council (DGC) and School Graduate Committee (SGC).

__________________	_________	___________
Department Head	Signature	Date
__________________	_________	___________
School Dean	Signature	Date
_________________	___________	___________
Office of Postgraduate Studies, Dean	Signature	Date

ACKNOWLEDGMENTS

Primarily, I would like to thank my Father in Heaven. Then, I would like to express my deep gratitude to my advisor Dr. Milkias Berhanu Tuka for his supervision, friendly attitude, and sharing of good ideas and materials. He is highly appreciated!

Great thanks to my whole family for their encouragement to work hard in my academics as well as for the sacrifices they have made. My extreme appreciation is for my two younger sisters for their unconditional encouragement, unlimited love, and infinite enthusiasm; I could not have gone through this journey without their help and support. The presence of my siblings has played a vital role to accomplish my work. Long live my siblings.

My sincere appreciation is extended to my best friend Mr. Moti Bekuma for his appreciation and reinvigoration. I am very grateful to my other friends for sharing their knowledge and materials with me. We had a wonderful moment together!

Finally, I would like to appreciate Haramaya University for offering me the chance to attend the Master’s Program. Further, my great thanks to Adama Science and Technology for their invitation and acceptance to the Master’s program.

Stay blessed!

Niguse Assefa Abebe

TABLE OF CONTENTS

CONTENTS **PAGE**

LIST OF FIGURES

LIST OF TABLES

ACRONYMS

ANFIS	Artificial Neuro-Fuzzy inference system
DFIG	Doubly fed Induction Generator
DFIM	Doubly fed Induction Machine
DC	Direct Current
DQ	Direct quadrature
FLC	Fuzzy Logic Controller
GSC	Grid Side Converter
IEEE	Institute of Electrical and Electronics Engineering
IGBT	Insulated Gate Bipolar Transistor
MATLAB	Matrix laboratory
MPPT	Maximum power point tracking
MW	Mega Watt
PI	Proportional Integral
PID	Proportional Integral derivative
PWM	Pulse Width Modulation
VSCF	Variable Speed Constant Frequency
RSC	Rotor Side Converter
TSR	Tip Speed Ratio
WECS	Wind energy conversion systems

LIST OF SYMBOLS

C_p	Aerodynamic power coefficient
T_{ae}	Aerodynamic torque (Nm)
ρ	Air density (Kg/m^3)
v_{ds} / v_{dr}	Direct axis stator/rotor voltage (V)
Ψ_{dr} / Ψ_{ds}	Direct axis rotor/stator flux (Wb)
i_{dr} / i_{ds}	Direct axis rotor/stator current (A)
Tem	Electromagnetic torque (Nm)
ω_r	Frequency of rotor voltages and currents (rad/s)
ω_s	Frequency of stator voltages and currents (rad/s)
N	Gearbox exchange ratio
V_g	Grid voltage (V)
J	Inertia of the rotor (Kgm^2)
L_m	Mutual inductance (H)
θ_{pitch}	Pitch angle (radian)
i_r	Rotor current (A)
Ψ_{qs} / Ψ_{qr}	Quadrature axis stator/rotor flux (Wb)
v_{qs} / v_{qr}	Quadrature axis stator/rotor voltage (V)
i_{qr} / i_{dr}	Quadrature axis rotor/stator current (A)
Qs	Reactive power (VAR)
Ω_t	Rotor angular speed (rev/min)
ω_m	Rotor electrical speed (rad/s)
R	Rotor radius (m)
i_s / i_r	Stator current/rotor (A)
L_s / L_r	Stator/rotor leakage inductance (H)
R_s / R_r	Stator/rotor resistance (Ω)
v_r / v_s	Supplied rotor/stator voltage (V)
Ps	Stator power (W)
λ	Tip speed ratio

ABSTRACT

A 2 MW doubly fed induction generator (DFIG) wind turbine is used in this thesis as the plant of the overall system, and the mathematical modeling of the wind speed, aerodynamics, mechanical component, and electrical component of the wind turbine along with the appropriate control mechanisms are incorporated. The dynamic features of DFIG mainly depend upon rotor current, stator current, and stator flux. These three parameters of DFIG increase the overall system complexity. One of the DFIG parameters is rotor current, which needs to be controlled because the output current of the wind turbine has greater overshoot, very large settling time, and large steady-state error, which reduce the output power efficiency and affect the system's stability. To reduce such problems the design of a reliable and high-performance controller is required. Hence, the design and analysis of rotor current control strategy have been applied to the rotor side converter of a doubly fed induction generator in this thesis work as a solution to the shortcomings of the DFIG parameter. The conventional control design, such as a proportional-integral (PI) controller has many shortcomings, such as difficulties in setting parameters and reduced robustness. Thus, the fuzzy logic, fuzzy-PI, and artificial neuro-fuzzy inference system (ANFIS) controllers are designed to regulate the rotor current loop of DFIG in the rotor side voltage source converter. By applying a vector control strategy, the rotor-side converter (RSC) is controlled to regulate the generator rotor speed, torque, and rotor currents, since the conventional controller could not satisfy the required system performance of the rotor currents. The designed systems are simulated in MATLAB/Simulink. The performance of the system subject to PI, fuzzy, fuzzy-PI and ANFIS are presented, and the result shows that the system used with artificial intelligent controllers show the best performance compared to the conventional PI controller in some performance index. When the objective of control is steady state error (SSE) of the system's rotor current, the SSE of the system used with; PI is 2.9084 A, fuzzy,0.8668 A, fuzzy-PI, 7.654A, ANFIS, 11.5472. The fuzzy controller based system is found to be the best candidate for SSE control of rotor current. The ANFIS controller based system has the best settling time for both quadrature axis and direct axis rotor currents, and the fuzzy-PI controller experiences the best performance from the perspective of steady-state error and settling time for torque control purposes.

Keywords: *ANFIS, Artificial Intelligence-based controller (AI), DFIG-based Wind Turbine, FLC, Fuzzy-PI, PI controller, Rotor Current Control.*

CHAPTER ONE

1. INTRODUCTION

1.1 Background

The continuously growing populations and the rapidly developing industries are increasing the demand for energy that is highly required to satisfy the modern standards of human needs. Therefore, renewable energy sources have been incorporated into modern power systems to contribute to power production and compensate for the growing energy gap (Abo-Khalil 2013). Renewable energy assets have gained critical consideration from industry and scholars because of their ecological friendliness. Among the renewable energy sources, wind power is growing rapidly in the energy industry. The wind is a clean, free, and inexhaustible energy source (Gupta 2015). Electrical Energy production from wind turbines has been the focus of the past decade in power production. Wind power uses wind energy conversion systems to provide an eco-friendly power generation and helps to meet the national energy demand when there is a diminishing trend in terms of non-renewable resources (Naidu, Member, and Singh 2015).

Wind energy conversion system (WECS) includes wind turbines, generators, and the controlling mechanism. Wind turbines are classified into two types, fixed speed, and variable speed wind turbines from the perspective of their operational speed.

1.1.1. Fixed Speed Wind Generator: In the early years of wind power systems development, fixed speed wind turbines and induction generators were the most commonly used in wind farms. A fixed-speed wind generator normally consists of a squirrel cage induction generator that is directly connected to the grid with limited speed variations. In this type of generator, the power can only be controlled through pitch angle variations. Since the wind turbine efficiency mainly depends on the tip-speed ratio, the power of a fixed-speed wind generator is directly proportional to the wind speed. Since the generator is considered an induction machine that does not have reactive power control capabilities, power factor correction systems are usually utilized to compensate for the reactive power demand of the generator (Eloghene Okedu 2019).

1.1.2. Variable Speed Wind Generator: Unlike the fixed speed generators, the variable speed concepts ensure that the wind turbine operates at the optimum tip-speed ratio, and accordingly this will result in achieving the optimum power coefficient for a wide wind speed range. Variable speed generators are capable of supplying a higher amount of energy to the network while maintaining lower power fluctuations. The most popular variable speed wind generator is the doubly fed induction generator (DFIG) (Deshpande and Kulkarni 2020; K. Okedu 2020).

However, the dynamic features of such generators are mainly dependent on certain non-linear parameters, such as stator flux, stator current, and rotor current, which makes the whole system more complicated. Therefore, it is necessary to design a robust controller in this type of system to support the dynamic frequency of wind energy to maintain stability and good performance of the system (K. E. Okedu and Barghash 2021).

A conventional control systems, such as proportional-integral controller has many shortcomings, such as difficulty in setting parameters and reduced robustness. Thus, the fuzzy logic, fuzzy-PI, and artificial neuro-fuzzy inference systems (ANFIS) controllers are designed to regulate the rotor current loop of DFIG in the rotor side voltage source converter. By applying a vector control strategy, the rotor side converter (RSC) is controlled to regulate the generator rotor speed, torque, and rotor currents. The rotor speed and torque are controlled by the outer control loop, and the inner control loop is responsible for the rotor current components.

1.2 Statement of the Problem

Originally, the wind power generation industry relied on fixed-speed wind turbines that comprised an asynchronous squirrel-cage induction generator with a multistage gearbox, which was driven by the wind turbine. It has a simple structure and requires a simple manufacturing process but disadvantages of reactive power compensation, no support for grid fault stability imposing higher mechanical stresses on the wind turbine (Naik and Gupta 2018). Integration of the variable speed wind turbine generator to the grid affects power system operation and stability due to harmonics and flickers in the voltage, current, and power. A variable speed wind turbine generator, a doubly fed induction generator (DFIG) works in a wide wind speed variation range. The operation of DFIG is dependent upon wind speed. During lower wind speed, the rotor will draw power from the grid, while in the case of high wind speed; the rotor delivers power to the grid. DFIG is becoming popular because of its simplicity to implement advanced features for grid integration (Dida, Merahi, and Mekhilef 2020). The dynamic features of DFIG mainly depend upon rotor current, stator current, and stator flux. Since these three parameters of DFIG are non-linear; they increase the overall complexity of the system. One of the main parameters of DFIG is the rotor current, which needs to be controlled because the output current of the wind turbine has a greater overshoot, a very large settling time, and a large steady-state error. These transients reduce the output power efficiency and affect the system's stability (Qin and Sun 2018).

To mitigate the problems mentioned above, the design of a reliable and high-performance controller is required. Thus, different DFIG-based wind-turbine control strategies such as control of pitch angle, voltage, current, torque, and active and reactive power have been widely studied in the literature to find the solution for the problems aforementioned. The advantages of rotor connected back to back voltage source converter control, include reduced flicker, variable speed constant frequency operation, independent control capabilities for active and reactive powers, and relatively lower converter cost and power losses, have captured researchers' and manufacturers' attention all over the world (Marugán et al. 2018). Hence, the design and analysis of rotor current control strategy have been applied to the rotor side converter of a doubly fed induction generator in this thesis work as a solution to the problems stated above.

1.3 Objectives

1.3.1 General Objective

The main objective of this thesis is to design and analyze an artificial intelligence-based rotor current control strategy for a doubly fed induction generator.

1.3.2 Specific Objectives

The specific objectives of this thesis are to:

- ✓ Develop vector control strategy for the RSC.
- ✓ Develop the rotor current controller for torque and reactive power (Qs) control
- ✓ Design proportional integral and artificial intelligent controllers for the RSC in order to control the rotor current of the doubly fed induction generator.

1.4 Scope of the Study

This thesis work discusses the wind energy issues and challenges of maximum power extraction from wind turbines by comparing different control strategies based on DFIG rotor current control. The thesis presents the modeling and simulation of the system based on PI, fuzzy, fuzzy-PI, and artificial neuro-fuzzy inference system (ANFIS) control strategies applied to the DFIG's rotor current. These control strategies are applied to rotor side converter control so that the rotor side currents are regulated, and the work mainly focuses on the doubly fed induction generator where constant electromagnetic torque and wind speed are considered as input to the generator. Further, the grid side converter is replaced with a DC voltage source to have a concise view of the rotor side convertors' performance subject to the four different control strategies that are mentioned above. The comparative analysis of the performance of these controllers is presented in the result and discussion chapter of this thesis work.

1.5 Significance of the Study

As examined from the reviewed literature, reliable and high-performance controllers are required to be incorporated into wind turbine power generation systems to maintain stability and improve efficiency. Thus, this thesis considered the major drawback of the currently used conventional controller and replaced them with modern controllers. The main achievement of the thesis is the design and simulation of a DFIG-based wind turbine controller that satisfies the needed standards in MATLAB/Simulink. The comparative analysis of the designed controllers has been made with the widely used conventional PI regulator. The conventional PI controller is replaced by intelligent Controllers to enhance the DFIG rotor current components' performances from the perspective view of the system's overshoot, undershoot, steady-State error, rise time, and settling time.

1.6 Organization of the Thesis

This thesis is organized as follows:

Chapter 1: background of the thesis work, the problem statement, objectives of the study, scope of the study, significance of the study, and organization of the thesis.

Chapter 2: background of the study, and review of related work.

Chapter 3: introduction, materials used, methods employed, modeling of wind turbines, development of control systems, and simulation of the modeled system.

Chapter 4: presents the simulation results and discussions.

Finally, the thesis presents the conclusion of the work and recommendations for future work.

CHAPTER TWO

2. LITERATURE REVIEW

Wind energy is one of the most significant renewable energy sources. More than 420 GW of electrical power was produced by wind energy conversion systems (WECS) in 2016, and in the 2030s, this is anticipated to increase to more than 1000 GW. The number of requirements to ensure efficient energy production has increased since wind energy systems are undergoing a drastic change. A rise in WECS and their complexity, together with a need for new strategies and tactics to improve reliability, maintenance, and investments, have all contributed to increased competition in the energy market (Marugán et al. 2018).

Due to their variable-speed operation with decreased mechanical stress, high controllability, smoother grid connection, maximum power extraction, reactive power compensation using back-to-back power converters with ratings close to 25–30% of the generator capacity, as well as the controlling flexibility of reactive power, DFIG-based wind turbines have drawn increasing attention and are dominating the global market (Ayrir, Meriem, and Haddi 2017).

The rotor side converter control is one of the mechanisms employed in DFIG. To control the rotor side converter, a variety of different structures and control algorithms could be applied. Rotor current control is one of the most commonly used techniques (Abo-Khalil et al. 2019; H. Ahmad et al. 2021), thus this paper presents the rotor current control of DFIG, using intelligent control mechanisms instead of the conventional proportional-integral (PI) controller.

2.1 Wind Energy Conversion Systems

The first simple windmills were constructed approximately 200 BC, despite evidence suggesting that wind had been used for work since 5000 BC. The ancient windmills were used to convert wind energy into mechanical work, such as pumping water and driving vehicles. Although wind energy conversion was first developed in the late 19th century, it gained acceptance in the mid-1970s during the oil crisis (Gomroki, Vaez-Zadeh, and Jabbarnejad 2020).

These days, wind energy conversion systems come in a variety of sizes and technologies, from tiny, hand-held models used to charge smartphones and for other extremely low-power tasks to enormous, offshore industrial models. The existence of these varieties has led to the development of several criteria for classifying modern wind turbines (Chijioke and Kiray 2020).

2.1.1 Wind Turbine Classifications Based on Rotor Axis

Based on the orientation of the axis on which the rotor lies, wind turbines can be broadly divided into two categories.

I) Vertical axis wind turbines (VAWT)

Vertical axis wind turbines rotate on a vertical axis, thus the name “vertical axis”. As shown in Figure 2.1, the Darius, Savonius, and Helical types are the two popular VAWTs. These wind turbine types have the drawbacks of being less effective at extracting wind energy and more prone to vibrations. Despite these drawbacks, they are more affordable and simple since they do not require yawing and pitching, are easier to repair because their gears and generators are positioned closer to the ground, and are more easily scaled down to proportions suitable for use in densely populated areas (Rajendran and Jena 2014).

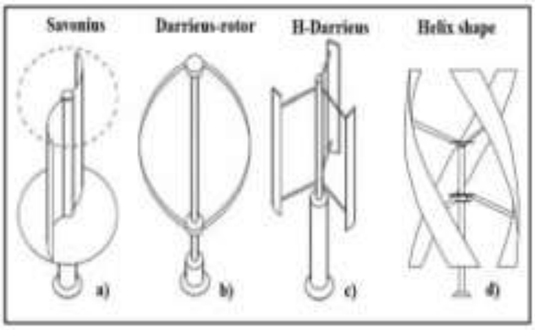


Figure 2. 1: Common vertical axis wind turbine

II) Horizontal axis wind turbines (HAWT)

Due to their high wind energy extraction efficiency, as depicted in Figure 2.2, these types of wind turbines are the more prevalent ones. They have a nacelle that contains the gear, generator, braking system, wind-vane, anemometer, and control system.

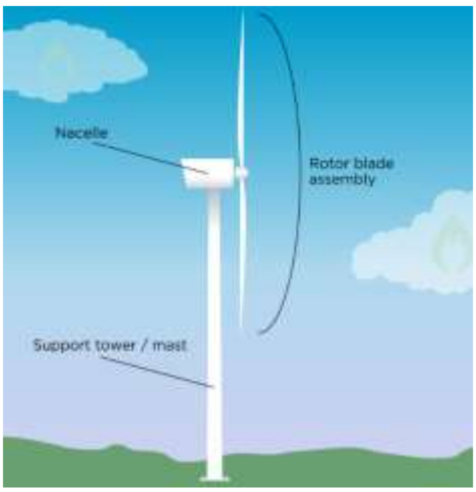


Figure 2. 2: Horizontal Axis Wind Turbine

In contrast to vertical axis wind turbines, where only one blade turns at a time, pulling the others with it, horizontal axis wind turbines are able to apply torque from all of their blades at once.

2.1.2 Wind Turbine Classification Based on Size

Based on their sizes, wind turbines can be categorized as follows:

I) Utility-scale wind turbines

These wind turbines, which have a capacity greater than 100 kW, are frequently used in wind farms, central stations, community use, and massive multi-turbine systems connected to the electricity grid (Cao, Xie, and T 2014). They could be onshore or offshore. They are referred to as distributed wind turbines when they are connected in close proximity to the customers to meet on-site energy demand (Gribkov and Chizhma 2021).

II) Large-scale wind turbines

These are primarily offshore wind farms built on vast bodies of water, generally near continental shelves. They are typically bigger than their onshore counter parts (Nayar et al. 2011).

2.1.3 Wind Turbine Classifications Based on Rotor Speed

I) Fixed speed

Fixed speed wind turbines are made to operate at their maximum capacity only at a single wind speed. Their speeds are governed by the required supply grid frequency, gear ratio, and generator design, independent of the wind speed. They are commonly used with squirrel cage or wound rotor induction generators (Soliman, Nesser, and Pytel 2019)**.**

II) Variable speed

Due to their ability to achieve aerodynamic efficiency over a wide range of wind speeds, these types of wind turbines have become the most widely used. They keep the generator's torque constant by using the proper control mechanisms, which allows them to accommodate changes in wind speed (Soliman, Nesser, and Pytel 2019; WINDExchange: Utility-Scale Wind Energy n.d.).

2.1.4 Wind Turbine Classifications Based on Generator Type

There are three primary types of wind turbine generators: AC synchronous generators, AC asynchronous generators, and DC generators. AC synchronous wind turbine generators operate on either permanent magnet synchronous generators (PMSG) or electrically excited synchronous generators (EESG). While the AC asynchronous wind turbine generators are based on induction. They are utilized as either doubly fed induction generators (DFIGs) with wound rotors or squirrel cage induction generators (SCIGs) (Gomroki, Vaez-Zadeh, and Jabbarnejad 2020).

The DFIGs have been widely used in wind energy conversion systems due to their relative simplicity in controlling active and reactive power as well as their higher operational energy efficiency. Due to the fact that only 30% of the power in doubly fed induction generators is handled by electronic components, the DFIG architecture is more economical than alternative directly linked full-power rated topologies (K. E. Okedu 2019).

Table 1 summarizes further justifications for choosing DFIG over alternative generator types. The wind turbine's rotor position (upwind tower or downwind tower), drive type (direct drive or geared drive), and yaw control (active or free) are additional classification standards. Power control (stalling, pitching, yawing, or use of aerodynamic surfaces), hub (rigid, teetering, gimbaled, or hinged blades), and finally, dependent on whether they are physically rigid or flexible (Gomroki, Vaez-Zadeh, and Jabbarnejad 2020).

Table 2. 1: Comparison of Wind Turbine Generators

Indicator	**DC Generator**	**Induction Generator**			**Synchronous Generator**		
		FSIG	**DFIG**	**Electro-Magnet**	**PM**	**Reluctance**	**HTS**
Reliability	Poor	Medium	High	High	High	Very High	High
Fault Response	Slow	Slow	High	High	High	High	Very High
Efficiency	Low	Low	High	High	Very High	Medium	Extremely High
Controllability	Poor	Poor	Good	Good	Good	Good	Very Good
Grid-Support Capability	Low	Low	High	Medium	Very High	Medium	High
Voltage fluctuation	High	High	Low	Low	Low	Medium	Very Low

DC generator wind turbine system constitutes wind turbine, DC generator, Insulator Gate Bipolar Transistor (IGBT) inverter, controller, and transformer components. Except for heating applications, low-power requirements, and situations where the loads are physically close to the wind turbines, DC generators are rarely used (Gomroki, Vaez-Zadeh, and Jabbarnejad 2020).

2.2 Wind Turbine Topologies

Wind turbines convert the kinetic energy from the wind into mechanical energy by producing torque. The wind turbine's blades face the wind flow to capture the wind's kinetic energy in order to generate the torque. The torque produced by the wind then turns the wind turbine's rotor, creating mechanical energy. The magnitude of energy contained in the wind depends on the air density and the wind velocity since the form of energy that the wind constitutes is kinetic energy (Syahputra et al. 2017).

The mechanical energy produced by wind turbines that harnesses the kinetic energy of moving air is converted into electricity by a generator. The generator is the component that converts the rotational energy from the high-speed shaft in the gearbox into an electrical current (How Do Wind Turbines Work? | Department of Energy n.d.; Wind Energy | Alternative Energy Tutorials n.d.).

The principle of electromagnetic induction indicates that while a magnet is moving past a coil of wire, an electric current is induced in the wire. There are two main components inside the generator, the rotor, and the stator. The rotor is the rotating part while the stator is stationary. While some systems utilize rotating magnets against static coils of wire and others use rotating coils of wire against static magnets, both approaches produce an electric current in the coils as their end result (How electricity is generated - U.S. Energy Information Administration (EIA) n.d.). The basic idea is that wind turbines use rotor blades to collect wind's potential and kinetic energy. The wind turns the blades, which spin a rotor that is connected to a generator to create electric energy. A modern wind turbine consists of different parts, which can be divided into three main components as illustrated in Figure 2.2 (The Parts of a Wind Turbine: Major Components Explained - Energy Follower n.d.).

A. Support tower or mast keeps the generation equipment and rotor blades far above the ground.
B. Nacelle houses the turbine generator, gearbox assembly, mechanical braking system, aerodynamic braking system, and electrical power transmission systems as shown in Figure 2.3.
C. Rotor Blades are the three (typically three) lengthy, thin blades that are mounted to the nacelle's hub. The blades combined with the hub make the rotor. The blades are built so that they can capture the kinetic energy of the passing wind and convert it into rotational energy (Different Types and Parts of a Horizontal Axis Wind Turbines - DolceraWiki n.d.).

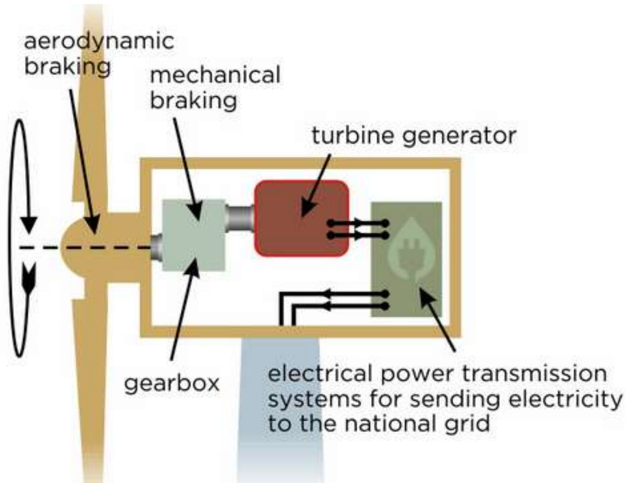


Figure 2. 3: Major components of wind turbine nacelle

2.3 Doubly Fed Induction Generator

As mentioned in the preceding section, DFIG is the most widely used of all the WECS technologies because it combines the benefits of lower converter ratings for power conversion with effective power capture as a result of variable speed operation. Unlike a standard, singly-fed induction generator, the electrical power of a doubly fed induction machine is independent of speed. Therefore, it is feasible to create a wind generator with variable speed that allows mechanical speed adjustment in response to wind speed and, as a result, operates the turbine at the aerodynamically optimal point for a certain wind speed range (Badreldien et al. 2014).

The majority of the top wind turbine producers in the world make use of doubly fed induction generator technologies. This is because just a small portion (20–30%) of the overall power, or the slip power, must be handled by the power electronic converter. This means that if the speed is within a 30 percent range of the synchronous speed, the converter has a rating of 30 percent of the rated turbine power, which reduces losses in the power electronic converter as compared to a system where the converter must handle the entire amount of power. Previously, the AC to AC converter that is connected to the rotor consisted of a thyristor bridges based rectifier and inverter. Nowadays, AC-AC converters are designed with bidirectional IGBT's, linking the rotor of the doubly fed induction generator to the electrical grid (Syahputra et al. 2017).

The basic configuration of the typical doubly fed induction generator is shown in Figure 2. The wind turbine is connected to the induction generator through a mechanical shaft system that consists of a low-speed shaft connected to the wind turbine, a high-speed shaft connected to DFIG, and a gearbox in between (Badreldien et al. 2014).

DFIG is an induction generator with wound rotor winding which consists of the stationary (stator) and rotating (rotor) parts (A Ashglaf 2019). Its rotor terminals are connected to the grid via a partial-load Variable Frequency, AC/DC/AC Converter, but the stator terminals are grid-connected. A back-to-back (i.e.AC/DC/AC) converter, which comprises two bidirectional converters and a dc link, serves as an optimal operation point-tracking interface between the generator and grid using modern power electronic techniques (Zou, Elbuluk, and Sozer 2013). The two bidirectional converters, i,e, RSC and GSC, are independently controlled. RSC controls active and reactive power by controlling the rotor current, whereas GSC regulates the intermediate circuit voltage and the reactive power exchange with the grid. The DC-link capacitor placed between the two power electronic converters is used as energy storage to maintain a constant DC bus voltage (Lamnadi et al. 2016; Noussi et al. 2019).

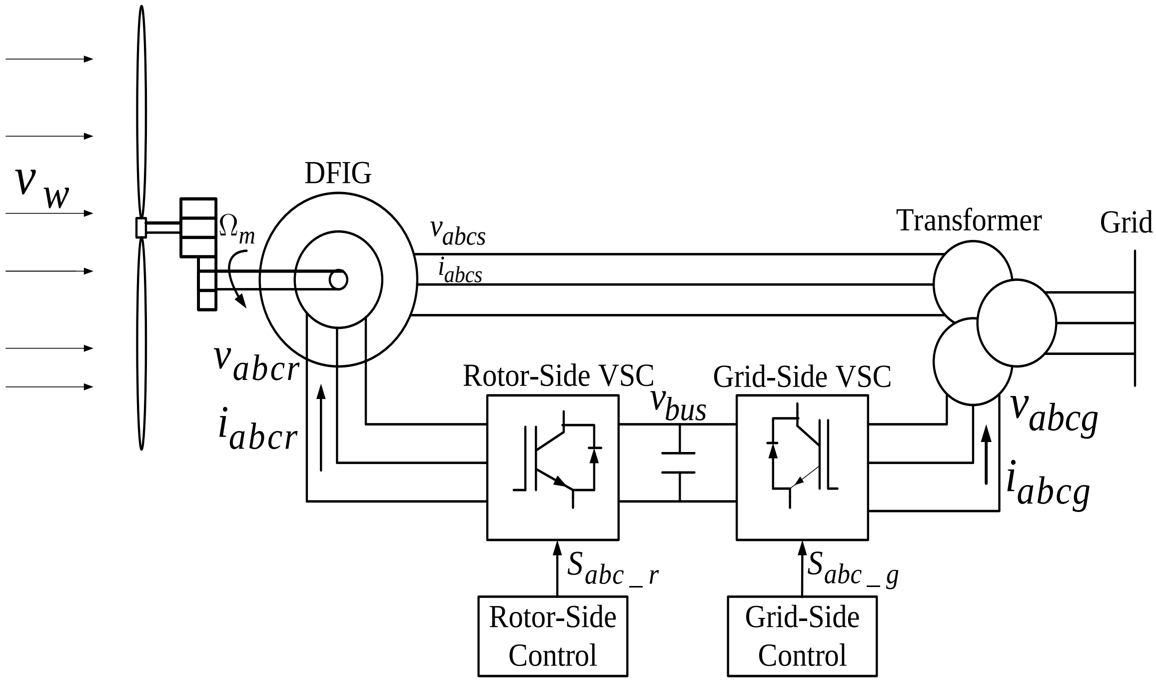


Figure 2. 4: General DFIG system configuration

2.3.1 Operating principles of DFIG

In the DFIG, both the rotor and the stator are able to supply active power, but the direction of this power flow in the rotor circuit depends on the wind speed and correspondingly the generator speed. Below the synchronous speed, active power flows from the grid to the rotor side and the GSC acts as a rectifier while RSC acts as the voltage source inverter. Above the synchronous speed, the rotor delivers power to the grid and RSC acts as the rectifier while GSC acts as the inverter. The converter manages only around 25% of the machine's rated power, and the variation range of the speed is 30% of the synchronous speed. According to the operation mode of DFIG, the stator side always supplies active power to the grid, but the active power is fed in or out of the rotor circuit (Badreldien et al. 2014).

2.3.2 Intelligent Control Techniques

A control method that can be carried out utilizing the "intelligence" of a human being who is competent in the specific control domain may be referred to as "intelligent control" in a broad sense. Human information processing speeds, sensory and actuation limitations, and other restrictions are not taken into account in this description. It follows that a system would be a strong candidate for intelligent control if a human using it can effectively operate a plant. Intelligent control is thought to benefit from information abstraction and knowledge-based decision-making that integrates abstracted information. Contrary to conventional control methods, intelligent control approaches are capable of dealing with uncertain or unanticipated conditions as well as incomplete knowledge of the plant and its surroundings. Expert control, fuzzy logic, neural networks, and sophisticated search algorithms are some of the artificial intelligence methods that can serve as the foundation for the creation of intelligent systems (Liu 2018).

2.4 Related Work

Doubly fed induction generators (DFIGs) are frequently employed in energy conversion systems. Designing high-performance control systems is crucial due to the dynamic properties of DFIGs. . Because these generators' dynamic properties depend on nonlinear factors like stator flux, stator current, and rotor current, the complexity of the entire system increases. Thus, robust controllers that can sustain the dynamic frequency of wind energy must be incorporated to ensure system stability. Conventional vector control designs that employ proportional-integral controllers have several disadvantages, such as troublesome parameter tuning, average dynamic performance, and decreased robustness (Abo-Khalil et al. 2019).

(Zhang and Jiao 2018), proposed a model predictive current control method based on the mathematical model of DFIG to directly regulate rotor current without requiring any tuning work. The reference rotor voltage is calculated and synthesized by space vector modulation (SVM) after getting the analytical differentiation of the rotor currents. The result demonstrates that all the three-phase rotor currents under the proposed method are sinusoidal and balanced when DFIG runs at an unbalanced network. However, the initial rotor current overshoot/undershoot and the steady-state error of the system response are not discussed.

(Hamane and Doumbia 2014), compared the performances (reference tracking and robustness) of classical Proportional-Integral (PI), Fuzzy-PI, and Sliding Mode (SM) controllers concerning the wind speed fluctuation and their impact on the produced energy quality based on doubly fed induction generator (DFIG). Further, direct vector control of the DFIG with stator flux orientation is employed to control the active and reactive powers between the stator and the grid. Their finding demonstrates that the transient responses provided by the Fuzzy-PI are best compared to conventional PI and Sliding Mode controllers (Hamane and Doumbia 2014). Nevertheless, their proposed method focuses on controlling the active and reactive power between the stator and the grid rather than the rotor side especially the rotor current of the DFIGs.

(Hamane et al. 2012), synthesized two types of controllers employing a classical PI and Fuzzy-PI to perform powers reference tracking and efficient disturbance rejection by using a decoupled control method of active and reactive powers for DFIG. The results indicate that when compared to the standard PI Controller, the Fuzzy-PI controller significantly reduces settling time, limits peak overshoot value, and dampens oscillations more quickly**.** But rather than controlling the rotor current of the DFIG, their proposed decoupled regulation method focuses on controlling the active and reactive power.

(Medjber et al. 2016), proposed a method that uses neural networks and fuzzy logic controllers to control the power transfer between the machine and the grid utilizing indirect vector control and reactive power control approaches. They controlled the RSC and grid side converter with a filter in between the two converters to ensure their work. Their results reveal that the response time obtained by the neural network controller is considerably reduced, and the peak values are exceeded relative to those of the fuzzy logic controller. However, the quadrature axis current does not match the reference value as can be seen from their results.

(Tang et al. 2014), presented a neural network-based controller for the reactive power control of wind farms with doubly fed induction generators to dampen the oscillation of the wind farm systems after the ground fault of the grid. The primary focus of the paper was on the online learning and control method based on adaptive dynamic programming for the grid integration and control of wind farms. The result indicates that the sag and overshoot of the active power and voltage of the wind farm could be reduced significantly after the disturbance is removed or the fault is recovered. Nevertheless, the obtained result looks good; the comparison was with no (open loop) controller.

CHAPTER THREE

3. MATERIALS AND METHODS

3.1 Chapter Overview

This chapter deals with materials used, mathematical modeling of the wind speed, aerodynamics, mechanical and electrical components of the wind turbine as well as the design of the control strategies for the rotor current. The designed control strategies are; PI, fuzzy logic, fuzzy PI and artificial neuro-fuzzy inference system. Finally, the system is simulated using MATLAB software in both Simulink and script format.

3.2 Materials

In this research work, as MATLAB, Microsoft office, Visio, and MathType softwares are used. MATLAB is the computing software that consists of technical toolboxes and Simulink. Microsoft office is used for editing the thesis documentation, and Microsoft Visio is used to draw figures. MathType is used for writing mathematical formulas and equations integrated with Microsoft office word and power point presentation tool.

3.3 Methods

To accomplish the general and specific objectives of the study stated in the first chapter, the methods listed below had been followed.

- Reading the most recent journals and magazines to discover the hot area of study that got researchers' attention, thus the title is determined.
- Gathering and reviewing of closely related papers and literatures from different books, related journals and google scholars.
- Modelling of DFIG based wind turbine and develop the currently used conventional control mechanism for 2 MW DFIG based wind turbine.
- Replace the conventional control mechanism by intelligent control technique so that the specified system requirements are achieved.
- Then, the design is simulated in MATLAB/Simulink and the results are analyzed.
- In case the design does not meet the requirements the system is revised
- Finally, the overall thesis work is documented and the result is presented.

The method used are summarized graphically as shown in Figure 3.1

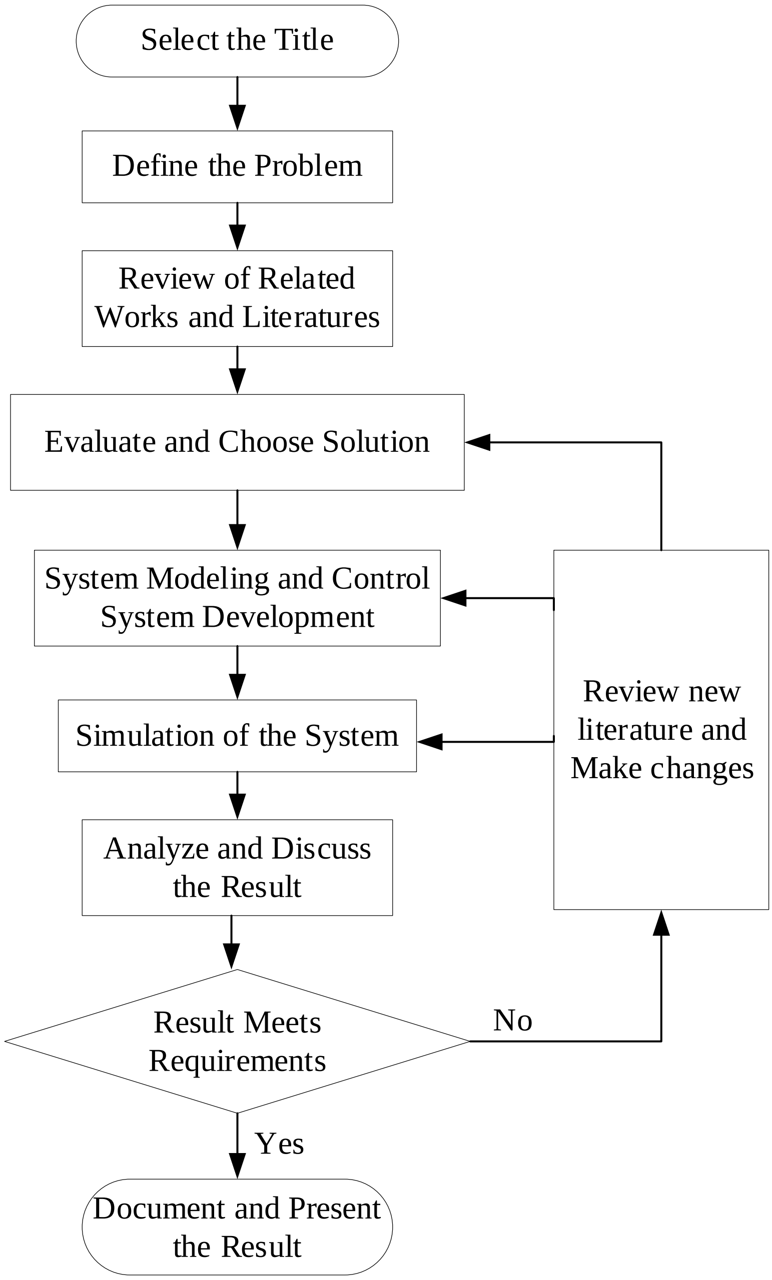


Figure 3. 1: Research Methodology

3.4 Modeling of Wind Turbines

The model under study consists of a single two-mass wind turbine connected directly to the power system. Assuming that the wind turbine has a general model consisting of aerodynamic, mechanical, electrical, and control systems, the model is present for all such components. Figure 3.2 shows the structure of the wind turbine design (Hansen et al. 2012; Kaloi, Wang, and Baloch 2016a). To determine the comparable wind speed fed to the aerodynamic model, the wind speed model is included. To account for the tower shadow, rotational turbulence, and fluctuations in wind speed over the rotor disk, the mechanical system model supplies the wind model with information about the position of the turbine rotor. The pitch angle from the control system and the speed of the turbine rotor from the mechanical model are supplied into the aerodynamic model along with the comparable wind speed. On the other hand, the mechanical model's generator speed is determined by the electrical model, and the aerodynamic model computes the aerodynamic torque provided to the mechanical model (Hansen et al. 2012). The mechanical model output is the mechanical turbine power that is an input to the electrical model. The electrical model is responsible for delivering the output currents to the power system through the wind turbine terminals while receiving input voltages from the power system. The input signals to the control system are 1) the hub wind speed from the wind model, 2) the generator speed from the electrical model, and, 3) the active and reactive powers, which represent the measured voltages and currents of the control system from the electrical model. The control systems output signals are 1) the pitch angle to the aerodynamic model, 2) the soft starter angle to the electrical model, and, 3) the capacitor bank switch signals to the electrical model (Hansen et al. 2012).

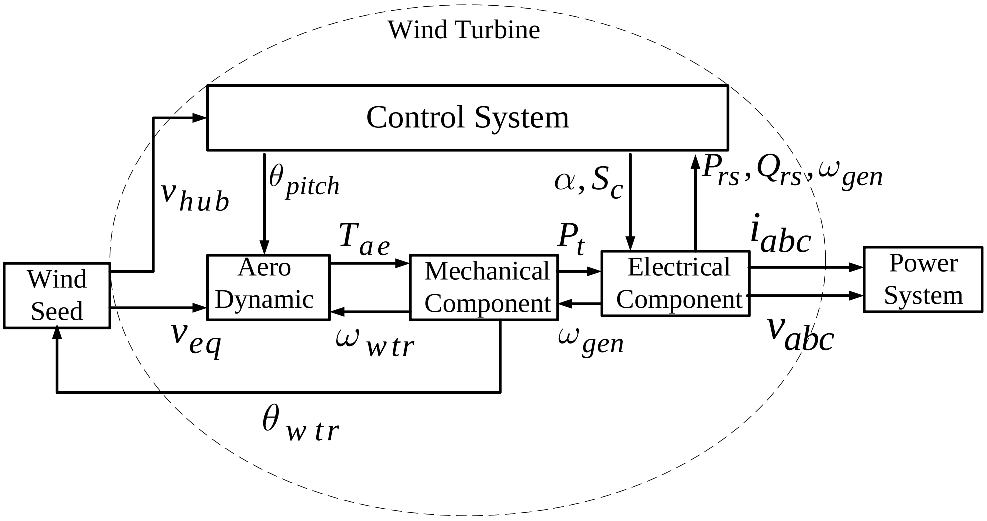


Figure 3. 2: General structure of the wind turbine model

3.4.1 Wind Speed Model

Wind speed varies from time to time and from one location to another. This variation highly affects the angular speed and torque of the wind turbine rotor and hence it has a direct relation to the amount of power captured by the wind turbine (Ibrahim and Watson 2015). Therefore, to simulate the WTGS dynamics properly, it is important to study the wind speed model. The approach of wind speed modeling in this work depends on summing up all four components of the wind speed such as 1) constant component representing the average wind speed $v_{wa}(t)$ 2) ramp component $v_{wr}(t)$ 3) gust component $v_{wg}(t)$ and 4) turbulence component $v_{wt}(t)$ as shown in the equation below (Hu, J. Nian 2016; Zhang and Jiao 2018):

$$v_w(t) = v_{wa}(t) + v_{wg}(t) + v_{wr}(t) + v_{wt}(t) \quad (3.1)$$

The wind speeds ramp component can be calculated by the following equation:

$$v_{wr}(t) = \begin{cases} 0, & t \leq T_{r1} \\ A_r\left(\dfrac{t - T_{r1}}{T_{r2} - T_{r1}}\right), & T_{r1} \leq t \leq T_{r2} \\ A_r, & t > T_{r2,} \end{cases} \quad (3.2)$$

Where T_{r1} and T_{r2} are the times representing the starts and ends of the ramp respectively, and A_r is the ramp maximum amplitude. The gust component simulates any abnormal temporary increase of the wind speed and it is represented by:

$$v_{wg}(t) = \begin{cases} 0, & t < T_{g1} \\ A_g\left((1 - \cos\left[2\pi(\dfrac{t - T_{g1}}{T_{g2} - T_{g1}})\right]\right) & ,T_{g1} \leq t \leq T_{g2} \\ 0, & t > T_{g2} \end{cases} \quad (3.3)$$

Where T_{g1} and T_{g2} are the times representing the starts and ends of the gust respectively, and A_g is the ramp maximum amplitude. Finally, the turbulence component is represented by a signal that has a power spectral density of the form (Noussi et al. 2019):

$$P_{Dt}(f) = \frac{lw_{wa}\left[\ln(\dfrac{h}{z_0})\right]^{-2}}{\left[1 + 1.5\dfrac{fl}{v_{wa}}\right]^{5/3}} \quad (3.4)$$

Where h is the wind wheel height, l is the turbulence scale, which is greater than h by a factor of 20 and has a maximum value of 300 meters, and z_o is a roughness length parameter.

A shaping filter that uses additive white Gaussian noise (AWGN), a flat-spectrum noise signal, is created to produce the turbulence signal. As a result, the response of a first-order filter is extremely close to the power spectral densityP_{Dt}. The transfer function of the designed filter is:

$$H(s) = \frac{kK}{s + p} \quad (3.5)$$

Where $p = \dfrac{2\pi((K_1{}^2)^{3/5} - 1)}{K_2\sqrt{K_1{}^2 - 1}}$, (3.6)

$$K = K_1 p, \quad (3.7)$$

Where K_1 and K_2 are:

$$K_1 = lv_{wa}\left[\ln(\frac{h}{z_0})\right]^{-2}, \tag{3.8}$$

$$K_2 = 1.5\frac{1}{v_{wa}}, \tag{3.9}$$

To obtain the power spectral density of the designed filter, the laplace transform of equation 3.5 is found as follow,

$$L^{-1}\left\{H(s) = \frac{kK}{s+p}\right\} = H(t) = kKe^{-pt}u(t)$$

$$P_{filter=F}\ H(t)\ = \int_{-\infty}^{\infty} kKe^{-pt}e^{-j2\pi ft}u(t)d(t)$$

$$= kK\int_{-\infty}^{0} e^{pt}e^{-j2\pi ft}d(t) + \int_{0}^{\infty} e^{-pt}e^{-j2\pi ft}d(t)$$

$$= kK\left[\frac{1}{p-j2\pi f} + \frac{1}{p+j2\pi f}\right]$$

Therefore, generated signal has a power spectral density of the form:

$$P_{filter} = \frac{2kK/p}{1+4\pi^2 f^2/p^2} \tag{3.10}$$

3.4.2 Aerodynamic Model

The blade element method is the most used way for building an aerodynamic model of a wind turbine (Kaloi, Wang, & Baloch, 2016). This method involves extensive computation, hence it is deemed inadequate for simulating a wind farm with several wind turbines.

The following equation is used to compute the aerodynamic torque (T_{ae}) based on the rotor radius (R), wind speed (v_w), air density (ρ), and power coefficient (C_t) (Abad et al. 2011).

$$T_{ae} = \frac{\pi}{2\lambda} pR^3 v_w{}^2 c_p(\theta_{pitch}, \lambda), \tag{3.11}$$

$$c_p = \lambda c_{t,} \tag{3.12}$$

$$\lambda = R\Omega_t / v_w \tag{3.13}$$

Where Ω_t represents DFIG rotor angular speed. In this work, a 2 MW wind turbine of type Mitsubishi MWT 92 is modeled. The presented aerodynamic model is based on data taken from the manufacturer's brochures (Abad et al. 2011). For the case of two-pole generators and 50 Hz grid, the selected ratio of the gearbox is 100, and the power coefficient and the tip blade maximum speed ratio are represented by the following equations, respectively:

$$c_p = 0.773\left(\frac{151}{\lambda_i} - 0.58\beta - 0.002\beta^{2.14} - 13.2\right) e^{-18.4/\lambda_i} \quad , \tag{3.14}$$

$$\lambda_i = \frac{1}{\lambda + 0.02\beta} - \frac{0.003}{\beta^3 + 1}, \tag{3.15}$$

3.4.3 Mechanical Model

Certain elements of the wind turbine's dynamic structure that directly influence how it interacts with the power system make up the mechanical or drive train model (Noussi et al. 2019). Since the drive train has the greatest impact on power fluctuations, it is the only component taken into consideration. The wind wheel, the turbine shaft (low speed shaft), the gearbox, and the generator's rotor shaft make up the drive train model (high speed shaft). The drive train transforms the aerodynamic torque on the turbine rotor into the torque on the low speed shaft (Hamane et al. 2012; Hamane and Doumbia 2014).

The gearbox reduces the torque applied to the high-speed shaft from the low speed shaft. The gearbox ratio between the low-speed and high-speed shafts typically ranges from 50 to 150, and the turbine rotor's inertia typically accounts for 90% of the system's inertia. (Kaloi et al., 2016). Typically, the drive train model is viewed as a collection of masses connected by an elastic coupling with linear stiffness, a damping ratio, and a multiplication ratio. In this work, a two mass model presented in Figure 3.3 is assumed where the wind wheel and the turbine rotor are considered as one inertia J_t and the gearbox and the generator's rotor as another inertia J_m that are connected, with a k angular stiffness coefficient and a c angular damping coefficient, through the elastic turbine shaft. The low speed shaft's inertia, Jt, and the damping coefficient, c, are considered to be 127 kg.m2 and 0.001, respectively, with a gearbox exchange ratio of N=100. The dynamics of the system are shown as:

$$\begin{pmatrix} \dot{\omega}_m \\ \dot{\omega}_t \\ \omega_m \\ \omega_t \end{pmatrix} = \begin{pmatrix} \frac{-v^2 c}{J_m} & \frac{vc}{J_m} & \frac{-v^2 k}{J_m} & \frac{vk}{J_m} \\ \frac{vc}{J_t} & \frac{-c}{J_t} & \frac{vk}{J_t} & \frac{-k}{J_t} \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix} \begin{pmatrix} \omega_m \\ \omega_t \\ \theta_m \\ \theta_t \end{pmatrix} + \begin{pmatrix} \frac{1}{J_m} & 0 \\ 0 & \frac{1}{J_m} \\ 0 & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} T_{em} \\ T_{ae} \end{pmatrix}, \tag{3.16}$$

where θ_t and θ_m represent the angles of the wind wheel and the generator shaft, ω_t and ω_m are the angular speed of the wind wheel and the generator, τ_{ae} is the torque applied to the turbine axis by the wind wheel and τ_{em} is the generator torque

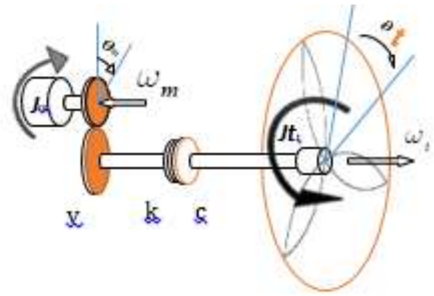


Figure 3. 3: Two mass drive train

3.4.4 Electrical Model

In this work, a doubly fed induction generator with the typical supply configuration of Figure 2.1 is used. The back-to-back three-phase converter together with the appropriate control strategy is in charge of imposing the required rotor AC voltages and frequencies to control the overall DFIG operating conditions to perform the power exchange between rotor and grid. In DFIG the stator supplies three-phase voltages which are controlled by rotor and grid side converters directly to the grid at constant amplitude and frequency (Abu-Rub 2014). However, the power exchange between the rotor and the grid depends on the operating modes of the generator shown in Figure 2.1. The rotor delivers power to the grid in hyper-synchronous mode of operation, and consumes power from the grid in the sub synchronous mode. Moreover, the electromagnetic torque is conventionally negative in generator mode of operation.

3.4.4.1 Steady-State Equivalent Circuit

The equation that relates the different frequencies of the machine, such as ω_s (frequency of stator voltages and currents), ω_r (frequency of rotor voltages and currents) and ω_m (rotor electrical speed) is as shown in equation (3.17).

$$\omega_s = \omega_r + \omega_m \tag{3.17}$$

$$\omega_m = p\Omega_m \tag{3.18}$$

The slip *s* of the machine is defined as depicted in equation (3.19):

$$s = \frac{\omega_s - \omega_m}{\omega_s} = \frac{\omega_r}{\omega_s} \tag{3.19}$$

In most applications, the stator windings are directly connected to the grid thus ω_s is constant. However, ω_r depends on the rotor's electrical speed ω_m, which leads to three operating modes of the machine dependent on the speed:

$\omega_m < \omega_s \Rightarrow \omega_r > 0 \Rightarrow s > 0 \Rightarrow$ Sub-synchronous operation

$\omega_m > \omega_s \Rightarrow \omega_r < 0 \Rightarrow s < 0 \Rightarrow$ Hyper-synchronous operation

$\omega_m = \omega_s \Rightarrow \omega_r = 0 \Rightarrow s = 0 \Rightarrow$ Synchronous operation

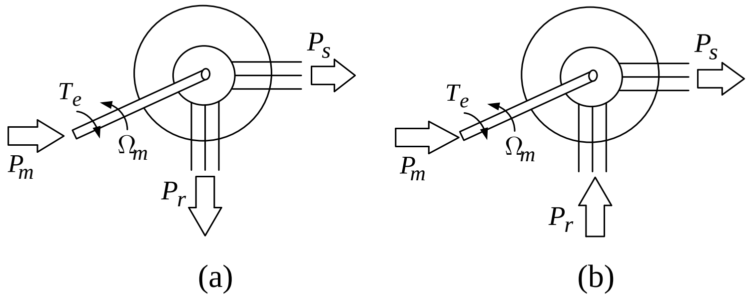


Figure 3. 4: (a) Hyper synchronous (b) Sub-Synchronous modes of operation of DFIG

3.4.4.2 Dynamic Modeling

Dynamic modeling is required to study the actual behavior of DFIG and to obtain a more reasonable perception about the machine. To perform dynamic modeling the system models should be represented in space vector model by using Clark and Park transformation. Figure 3.5 illustrates the three different rotating reference frames typically used to develop space vector-based models of the DFIG. The stator reference frame ($\alpha\beta$) is a stationary reference frame, the rotor reference frame (*DQ*) rotates at ω_m and the synchronous reference frame (*dq*) rotates at ω_s .The space vector can be represented in any of these frames by utilizing direct and inverse rotational transformation**.** Thus, by using space vector theory the voltage in the three coils of the stator and rotor can be represented separately by two stationary $\alpha\beta$ coils for the stator as shown in equation (3.20) and two rotating coils *DQ* for the rotor as represented in equations (3.21) (Abad et al. 2011; Abu-Rub 2014).

$$\vec{V}_s = R_s \vec{i}_s + \frac{d\vec{\Psi}_s}{dt} \tag{3.20}$$

$$\vec{V}_r = R_r \vec{i}_r + \frac{d\vec{\Psi}_r}{dt} \tag{3.21}$$

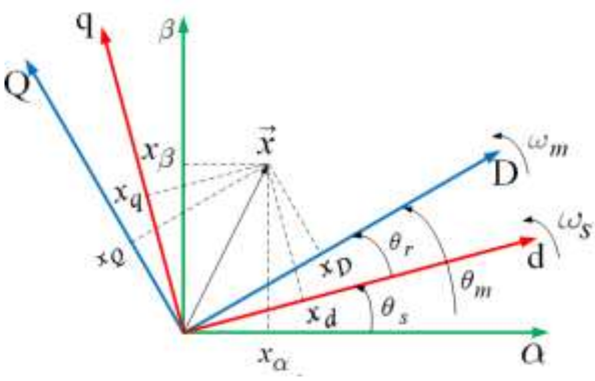


Figure 3. 5: Different reference frames to represent space vectors of the DFIM

i). The αβ model

To represent both voltage equations in stationary reference frame, $\alpha\beta$, then the rotor equation must be multiplied by $e^{j\theta_m}$ which yields the following set of equations:

$$\vec{V}_s = R_s \vec{i}_s + \frac{d\vec{\Psi}_s}{dt} \Rightarrow \begin{cases} V_{\alpha s} = R_s i_{\alpha s} + \dfrac{d\Psi_{\alpha s}}{dt} \\ V_{\beta s} = R_s i_{\beta s} + \dfrac{d\Psi_{\beta s}}{dt} \end{cases} \tag{3.22}$$

$$\vec{V}_r = R_r \vec{i}_r + \frac{d\vec{\Psi}_r}{dt} - j\omega_m \vec{\Psi}_r \Rightarrow \begin{cases} V_{\alpha r} = R_r i_{\alpha r} + \dfrac{d\Psi_{\alpha r}}{dt} + \omega_m \Psi_{\beta r} \\ V_{\beta r} = R_r i_{\beta r} + \dfrac{d\Psi_{\beta r}}{dt} - \omega_m \Psi_{\alpha r} \end{cases} \tag{3.23}$$

In a similar way, the stator and rotor flux expressions in space vector form in a stationary reference frame as shown in equation (3.24) and (3.25).

$$\vec{\Psi}_s = L_s \vec{i}_s + L_m \vec{i}_r \Rightarrow \begin{cases} \Psi_{\alpha s} = L_s i_{\alpha s} + L_m i_{\alpha s} \\ \Psi_{\beta s} = L_s i_{\beta s} + L_m i_{\beta s} \end{cases} \tag{3.24}$$

$$\vec{\Psi}_r = L_r \vec{i}_r + L_m \vec{i}_s \Rightarrow \begin{cases} \Psi_{\alpha r} = L_r i_{\alpha r} + L_m i_{\alpha r} \\ \Psi_{\alpha r} = L_r i_{\alpha r} + L_m i_{\beta r} \end{cases} \tag{3.25}$$

Then, from the set of equations (3.22) through (3.25), the $\alpha\beta$ equivalent circuit represented in Figure 3.6 is developed. All the voltage, current and flux magnitudes in the equivalent circuit are sinusoidal with a frequency ω_s . The active and reactive powers of the stator and rotor sides can be calculated by equations (3.26):

$$\begin{aligned} P_s &= \frac{3}{2}(v_{\alpha s} i_{\alpha s} + v_{\beta s} i_{\beta s}) \qquad P_r = \frac{3}{2}(v_{\alpha r} i_{\alpha r} + v_{\beta r} i_{\beta r}) \\ Q_s &= \frac{3}{2}(v_{\beta s} i_{\alpha s} - v_{\alpha s} i_{\beta s}) \qquad Q_r = \frac{3}{2}(v_{\beta r} i_{\alpha r} - v_{\alpha r} i_{\beta r}) \end{aligned} \tag{3.26}$$

The electromagnetic torque, T_{em} produced in the DFIG can be calculated by the following equivalent expressions:

$$T_{em} = \frac{3}{2} p \operatorname{Im}\ \vec{\Psi}_r \vec{i}_r^{*}\ = \frac{3}{2} p(\Psi_{\beta r} i_{\alpha r} - \Psi_{\alpha r} i_{\beta r}) \tag{3.27}$$

Where p is the number of the generator poles, which is equal to two in this work. The equation that describes the rotor speed behavior can be found by adding the mechanical motion:

$$T_{em} - T_L = J\frac{d\Omega_m}{dt} + D\Omega_m \tag{3.28}$$

Where *J* is the inertia of the rotor, and T_L is the load torque applied to the shaft.

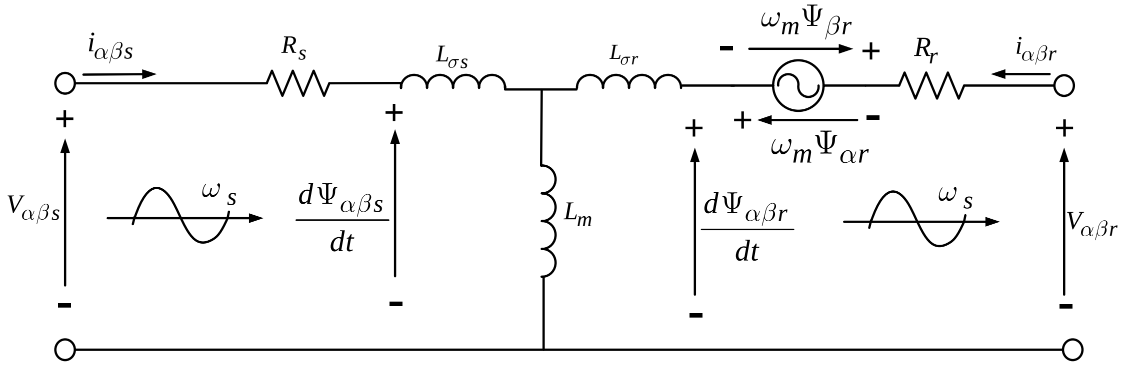


Figure 3. 6: Model of the DFIG in the $\alpha\beta$ reference frame

ii). The dq model

The space vector model representation of the DFIG in a synchronously rotating frame can be obtained by multiplying the voltage expressions (3.20) by $e^{-j\theta_s}$, and expression (3.21) by $e^{-j\theta_r}$. Equations (3.29) and (3.30) thus illustrates the voltage representation in *dq* frame.

$$\vec{v}_s = R_s\vec{i}_s + \frac{d\vec{\Psi}_s}{dt} + j\omega_s\vec{\Psi}_s \Rightarrow \begin{cases} v_{ds} = R_s i_{ds} + \frac{d\vec{\Psi}_{ds}}{dt} - \omega_s \Psi_{qs} \\ v_{qs} = R_s i_{qs} + \frac{d\vec{\Psi}_{qs}}{dt} + \omega_s \Psi_{ds} \end{cases} \tag{3.29}$$

$$\vec{v}_r = R_r\vec{i}_r + \frac{d\vec{\Psi}_r}{dt} + j\omega_r\vec{\Psi}_r \Rightarrow \begin{cases} v_{dr} = R_r i_{dr} + \frac{d\vec{\Psi}_{dr}}{dt} - \omega_r \Psi_{qr} \\ v_{qr} = R_r i_{qr} + \frac{d\vec{\Psi}_{qr}}{dt} - \omega_r \Psi_{qr} \end{cases} \tag{3.30}$$

Similarly, the stator and rotor flux equations in d-q axis yield:

$$\vec{\Psi}_s = L_s\vec{i}_s + L_m\vec{i}_r \Rightarrow \begin{cases} \Psi_{ds} = L_s i_{ds} + L_m i_{dr} \\ \Psi_{qs} = L_s i_{qs} + L_m i_{qr} \end{cases} \tag{3.31}$$

$$\vec{\Psi}_r = L_s\vec{i}_s + L_r\vec{i}_r \Rightarrow \begin{cases} \Psi_{dr} = L_r i_{dr} + L_m i_{ds} \\ \Psi_{qr} = L_r i_{qr} + L_m i_{qs} \end{cases} \tag{3.32}$$

Figure 3.7 represents the equivalent electrical circuit in *dq* reference frame obtained by combining the equations (3.29) through (3.32).

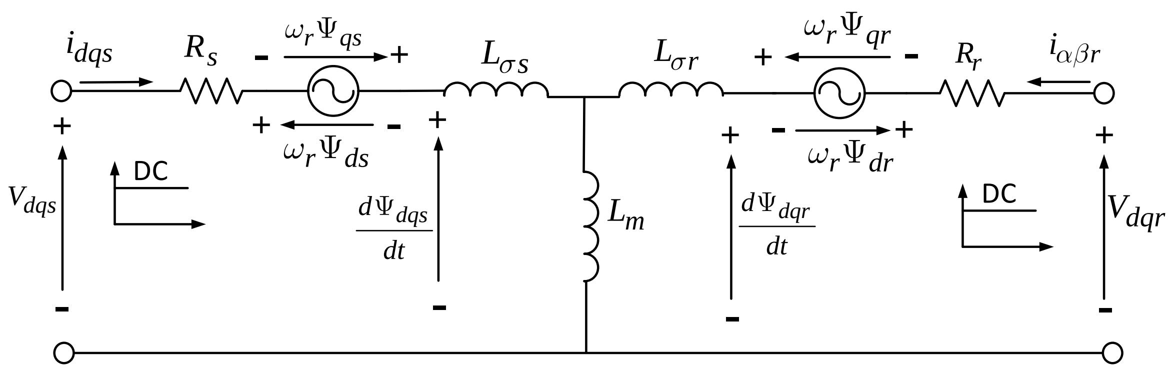


Figure 3. 7: Model of the DFIG in *dq* reference frame

Generally, the relations between the voltages on the generator windings and the currents, and its first derivative in the state space form can be studied on a synchronous (dq) reference frame as follows:

$$\begin{pmatrix} v_{qs} \\ v_{ds} \\ v_{qr} \\ v_{dr} \end{pmatrix} = \begin{pmatrix} L_S & 0 & L_m & 0 \\ 0 & L_S & 0 & L_m \\ L_m & 0 & L_r & 0 \\ 0 & L_m & 0 & L_r \end{pmatrix} \frac{d}{dt} \begin{pmatrix} i_{qs} \\ i_{ds} \\ i_{qr} \\ i_{dr} \end{pmatrix} + \begin{pmatrix} R_s & L_s\omega_s & 0 & L_m\omega_s \\ -L_s\omega_S & r_s & -L_m\omega_s & 0 \\ 0 & sL_m\omega_s & r_r & sL_r\omega_s \\ -sL_m\omega_s & 0 & -sL_s\omega_s & r_r \end{pmatrix} \begin{pmatrix} i_{qs} \\ i_{ds} \\ i_{qr} \\ i_{dr} \end{pmatrix} \tag{3.33}$$

Further, the electromagnetic torque applied on the high speed shaft and the reactive power generated by the stator inductance are represented by equation (3.34) and equation (3.35).

$$T_{em} = \frac{3}{2} pL_m (i_{sq} i_{rd} - i_{sd} i_{rq}), \tag{3.34}$$

$$Q_s = \frac{3}{2} (v_{sq} i_{sd} - v_{sd} i_{sq}), \tag{3.35}$$

3.5. Development of Control System

3.5.1 Control System Modeling

Under this topic, the control techniques of the doubly fed induction machine (DFIM) is presented. The Control system in a wind turbine plays an important role to maintain the magnitudes of the generator, such as torque, active and reactive power, and the DC bus voltage close to their optimum values, so that energy can be generated effectively. In this work, the control together with the modulator is used to perform a converter switch pulse generation function based on the desired values. There are two widely used control philosophies for DFIG: the direct torque control technique and the vector control (field-oriented control) technique (Ayrir, Meriem, and Haddi 2017). From the two control philosophies the vector control (field-oriented control) technique is adapted to this thesis work.

The basic concept of vector control is to refer to the rotor currents in a synchronous reference frame so that its d axis is aligned with the stator flux. In the steady-state, this is almost equivalent to aligning the q axis with the stator voltage, because the flux is shifted by 90 degrees with respect to the stator. Hence, there are two mechanisms to align the axis with the reference frame. One of the options is to estimate the flux and align the d axis to it, and the other option will be to align the q axis with the voltage and delay by 90 on the d axis. The first option is called flux orientation and the second option is voltage orientation (also known as grid flux orientation) (Abad et al. 2011; Hamane and Doumbia 2014).

In this work, the second method (i.e. voltage oriented synchronous reference frame) is used. In this method, the tasks performed are measuring the stator voltages and then aligning the q axis with the voltage space vector, then axis d, on which the stator flux lies, would be shifted 90 from it. A phase-locked loop (PLL) is commonly used with this method to obtain the angle of the fundamental and reduce the effects of voltage harmonics (Abad et al. 2011).

In the vector control of DFIG the machine is controlled in a synchronously rotating dq reference frame, where the d and the q axis of the rotor current components are regulated. If a reference frame orientated with the stator flux is used, the decoupled control of the active and reactive power flows of the stator by means of the quadrature and the direct current, respectively can be achieved Figure 3.8 (Abad et al. 2011).

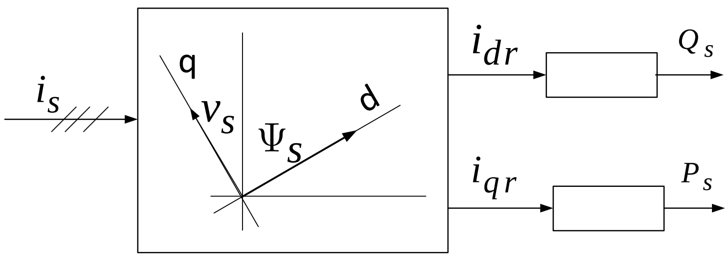


Figure 3. 8: Vector Control of the DFIG.

It can be observed from Figure 3.8 that Qs and Ps can be controlled independently by regulating the rotor current components Idr and Iqr, respectively. Consequently, the reference values of Idr and Iqr can be determined from the outer power control loops (Aarib et al. 2021; Badreldien et al. 2014).

3.5.2 Calculation of the Current References

The dq voltage and stator flux expressions of equations 3.29 to 3.34 can be simplified by using a reference frame aligned with the stator flux as can be seen from Figure 3.9. When the stator flux is set aligned with the d-axis, the quadrature axis flux is assigned to be zero.

(i.e $\Psi_{qs} = 0$). Thus under stator flux orientation the relationship between the current and fluxes may be written as (Abad et al. 2011; Abu-Rub 2014):

$$\begin{aligned} \Psi_{qs} &= L_s i_{qs} + L_m i_{qr} = 0 \\ \Psi_{ds} &= L_s i_{ds} + L_m i_{dr} = \Psi_s \end{aligned} \tag{3.36}$$

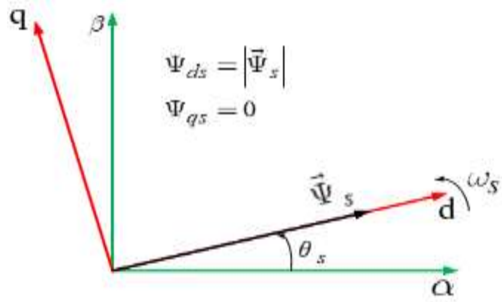


Figure 3. 9: Synchronous rotating *dq* reference frame aligned with the stator flux space vector

Now from the above equation the relationship between the rotor and the stator currents is developed:

$$\begin{aligned} i_{qs} &= -\frac{L_m}{L_s} i_{qr} \\ i_{ds} &= \frac{\Psi_s}{L_s} - \frac{L_m}{L_s} i_{dr} \end{aligned} \tag{3.37}$$

From Equation (3.29), assuming that the voltage drop in the stator resistance is negligible (i.e $R_s i_{ds} = 0$), the stator flux is constant because the stator is connected directly to the grid at constant AC voltage; as a result, the term $d\vec{\Psi}_{ds}/dt$ is zero. This yield:

$$\begin{aligned} v_{ds} &= 0 \\ v_{qs} &= \omega_s \Psi_{ds} = \omega_s \Psi_s \approx V_g \end{aligned} \tag{3.38}$$

From equation (3.38) above it can be understood that in steady-state, the stator flux is proportional to the grid voltage, V_g. Thus, when orientating the direct axis with the stator flux, the voltage aligns with the quadrature axis ($v_{ds} = 0$). The stator active and reactive power flow can then be obtained:

$$\begin{aligned} P_s &= \frac{3}{2} v_{qs} i_{qs} \\ Q_s &= \frac{3}{2} v_{qs} i_{ds} \end{aligned} \tag{3.39}$$

Combining equation (3.37), (3.38) and equation (3.9), the following equation is obtained:

$$P_s = -\frac{3}{2} v_g \frac{L_m}{L_s} i_{qr}$$

$$Q_s = \frac{3}{2} v_g \frac{\Psi_s}{L_s} - \frac{3}{2} v_g \frac{L_m}{L_s} i_{dr} = \frac{3}{2} \frac{v_g^2}{\omega_s L_s} - \frac{3}{2} v_g \frac{L_m}{L_s} i_{dr} \tag{3.40}$$

These equations show that under the stator flux orientation, the active and reactive powers are decoupled and can be controlled via the rotor currents. The active power can be controlled by means of the direct current while the quadrature current can be used to control the reactive. The reference currents can be calculated from the desired powers using equation 3.40 as shown in Figure 3.10. Then, controllers are designed to force the actual current to follow these references.

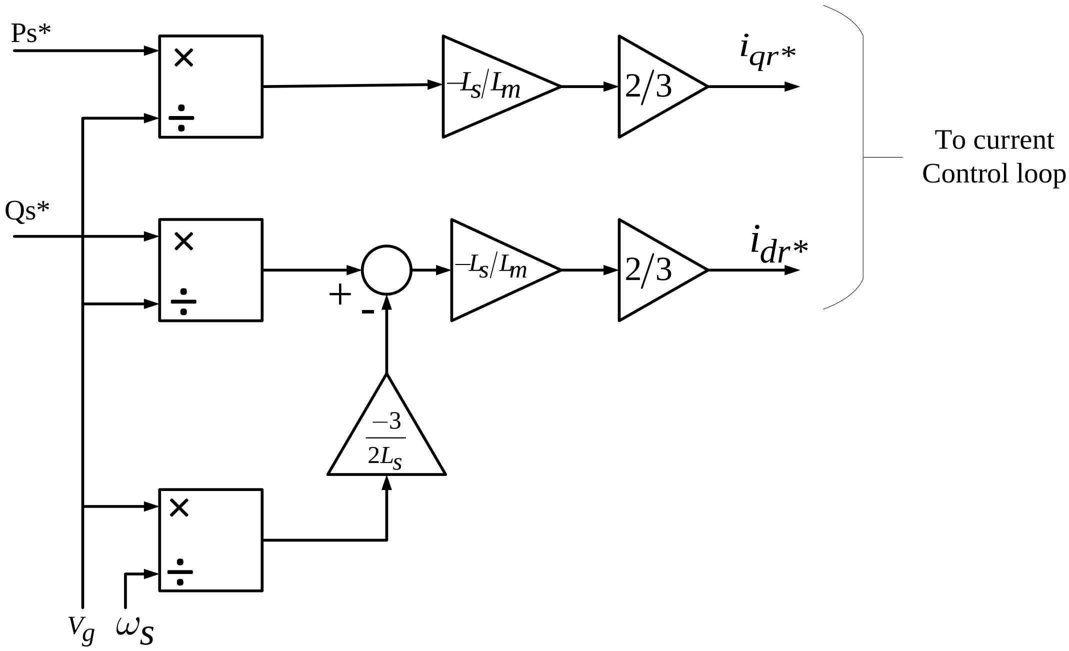


Figure 3. 10: Calculation of the current references

In the previous equations, the actual stator powers can't be exactly the desired values because the stator resistances was neglected for simplicity of calculation, which diminishes exactness of the relationships. To improve the accuracy of the relationships usually two outer power loops are added. In the first control loop the active power is controlled by quadrature axis current, i_{qr} , and in the second control loop, the reactive power is controlled by direct axis current, i_{dr} , as depicted in Figure 3.11. This control loop provides additional relevance as it is useful to correct the inaccuracies in the values of the machine's parameters (Ls, Lm, turns ratio, etc.). These outer power loops provide the advantage of accurate control of the power flow in the stator, regardless of the inexactness in the machine parameters or small misalignments of the reference frame.

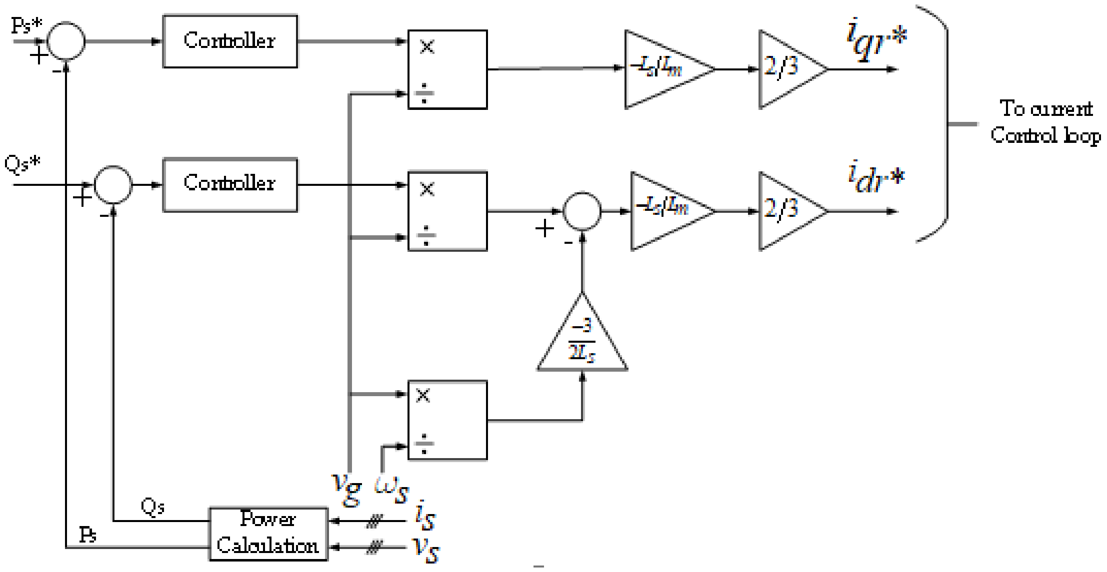


Figure 3. 11: Reference calculation including external power loops.

Most of the time the concern of control in the wind turbine is not of the stator power, but the electromagnetic torque generated by the machine. Thus, we can use the relationship between the torque, the stator flux, and the rotor current to develop the quadrature axis reference current.

$$T_{em} = \frac{3}{2} p \frac{L_m}{L_s} (\Psi_{qs} i_{dr} - \Psi_{ds} i_{qr}) \tag{3.41}$$

Under the stator flux orientation the quadrature axis component of the stator flux is zero so this expression is simplified to equation (3.42), and as shown in Figure 3.12. This mechanism is adapted to this thesis work, since it has wide research attention.

$$T_{em} = -\frac{3}{2} p \frac{L_m}{L_s} \Psi_{ds} i_{qr} \tag{3.42}$$

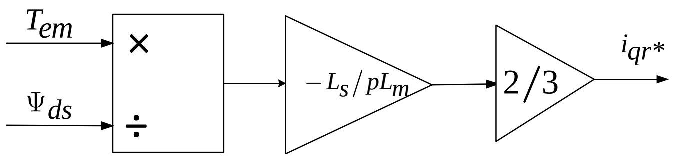


Figure 3. 12: Calculation of the reference current from T_{em}

3.5.3 Calculation of the Torque References

The assumptions used to derive i_{qr^*} have an impact on the controllability of the torque. The i_{qr^*} that is fed into the current controller in order to make the controller to follow reference torque value is not accurate because it does not take into account the misalignment of the flux with the reference frame. Therefore, the actual torque of the machine will always differ from the reference torque, resulting in a steady-state error. This can be eradicated by developing a speed controller in cascade with the current controller that is used to adjust T_{em*} in order to keep the rotor speed constant (Wang et al. 2016).

$$T_e - T_L = \frac{J}{p}\dot{\omega}_m + D\omega_m \tag{3.43}$$

The rotor speed reference value can be obtained based on the power-speed relationship of the wind turbine, typically in the form of a table or graph that shows the relationship between rotor speed and the input mechanical power (or mechanical torque). The actual rotor speed can thus be measured and compared to the reference value. The controller gives a reference torque that is fed into the current Controller to satisfy equation 3.44 and is shown in Figure 3.13 (Badreldien et al. 2014; Wang et al. 2016).

$$T.F = \frac{\omega_m}{T_{em\_ref}} = 1\Big/\left(\frac{J}{p}s + D\right) \tag{3.44}$$

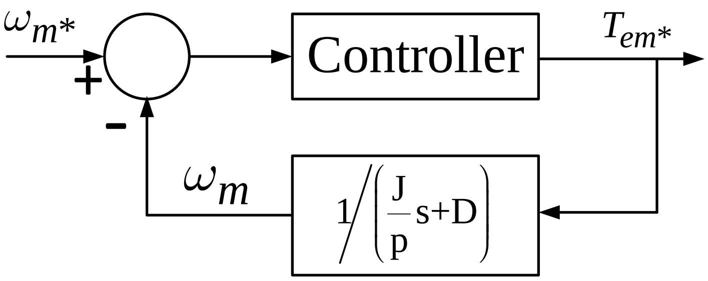


Figure 3. 13: Calculation of the torque references

3.5.4 Limitation of the Current References

The current references must be limited so that they do not exceed the maximum current allowed by the machine and the rotor converter. We will denote the maximum peak value of the rotor current as I_{r_max}. To assure a current of sinusoidal waveform, the magnitude of the space vector must be kept constant. The space vector has two components: the values in the d axis and the q axis. If the current has to be limited, we can act on either of the two components. A commonly used option is to proportionally limit both components, thus keeping the angle of the space vector unchanged so that the modulus of the space vector remain constant. Another usual option is to prioritize one component over the other. If, for example, tracking of the reactive power is considered more important than tracking of the active power, we will give priority to the d-axis current: the limitation will be applied first to the q component, and only if this is not enough, to the d component (Abad et al. 2011). In this work, both components are proportionally controlled by limiting the d component to zero.

3.5.5 Current Control Loops

The rotor side converter control scheme constitutes two control loops, the inner current control loop and outer control loop. The inner control loop regulate the d-axis and q-axis rotor current components, I_{dr} and I_{qr} ,and the outer control loops regulate both the stator active and reactive power independently as can be observed in Figure 3.11**.** For the inner current control loop, **t**he rotor converter must ensure, that the actual currents track the reference rotor currents calculated in equation 3.40 and 3.42 (Badreldien et al. 2014). The rotor side converter is a three-phase inverter that fixes the voltage of the rotor terminals but not the currents. Thus, the control loops must be incorporated to realize that the currents effectively follow their references. The vector control of the DFIM is performed in a synchronously rotating *dq* frame, in which the *d*-axis is aligned, in this work, with the stator flux space vector, as shown in Figure 3.9. Along this axis, the variables are constant during steady state, this helps in the regulation of the currents (Abu-Rub 2014).

There are many different controllers that can be used for the control loop; however, in this case the proportional integral (PI), fuzzy, fuzzy-PI and ANFIS controllers are implemented for efficient control of the DFIG system. To calculate the PI regulator parameters, the relationship between the rotor voltage and the rotor currents must be known. From the assumptions we have already made, neglecting the stator resistances, the voltage lies along the q axis, so its d component becomes zero. In addition to that, the model considers the d axis aligned with the flux, so the flux along the q axis is zero.

The voltage at the rotor terminals constitutes the sum of the voltage drop in the resistor Rr and the voltage induced by the flux rotor:

$$\begin{aligned} v_{dr} &= R_r i_{dr} + \frac{d}{dt}\Psi_{dr} - \omega_r \Psi_{qr} \\ v_{qr} &= R_r i_{qr} + \frac{d}{dt}\vec{\Psi}_{qr} + \omega_r \Psi_{dr} \end{aligned} \tag{3.45}$$

The relationship between this rotor flux and the rotor currents can be found by substituting stator current parameter of equation (3.31) into equation (3.32) as shown in equation (3.46).

$$\begin{aligned} \Psi_{dr} &= \ L_r - L_m^2 \big/ L_s \ \ i_{dr} + \frac{L_m}{L_s} L_m \Psi_{ds} \\ \Psi_{qr} &= \ L_r - L_m^2 \big/ L_s \ \ i_{qr}; \quad \Psi_{qs} = 0 \end{aligned} \tag{3.46}$$

From the first terms of the above equations, we examine that the flux directly related to the rotor current. The proportionality ratio between these fluxes and the currents is referred to as the transitory inductance of the rotor, σL_r, and is the result of adding in series the rotor leakage inductance with the parallel magnetizing and the stator leakage inductance:

$$L_r - L_m^2 / L_s = L_{\sigma r} + L_m / L_{\sigma s} = \sigma L_r \tag{3.47}$$

where σ is the leakage coefficient of the machine.

Substituting expressions (3.46) into the rotor voltage equation (3.45), the relationship between the rotor voltages and the currents are obtained:

$$\begin{aligned} v_{dr} &= R_r i_{dr} + \sigma L_r \frac{d}{dt} i_{dr} - \omega_r \sigma L_r i_{qr}; \qquad \text{where } \frac{d}{dt}\Psi_{ds} = 0 \\ v_{qr} &= R_r i_{qr} + \sigma L_r \frac{d}{dt} i_{qr} + \omega_r \sigma L_r i_{dr} + \omega_r \frac{L_m}{L_s}\left|\Psi_{ds}\right| \end{aligned} \tag{3.48}$$

In the regular operation that is when the grid voltage is constant in amplitude, the derivative of the flux become zero and that is why the last term of the first equation excluded. The graphical representation of the resulting equations is presented in Figure 3.14. In the second equation, the term $\omega_r L_m / L_s \; \Psi_{ds}$ is a perturbation since it depends on the stator flux, which is an external variable independent of the loop, but controller can easily compensate for it since it is constant.

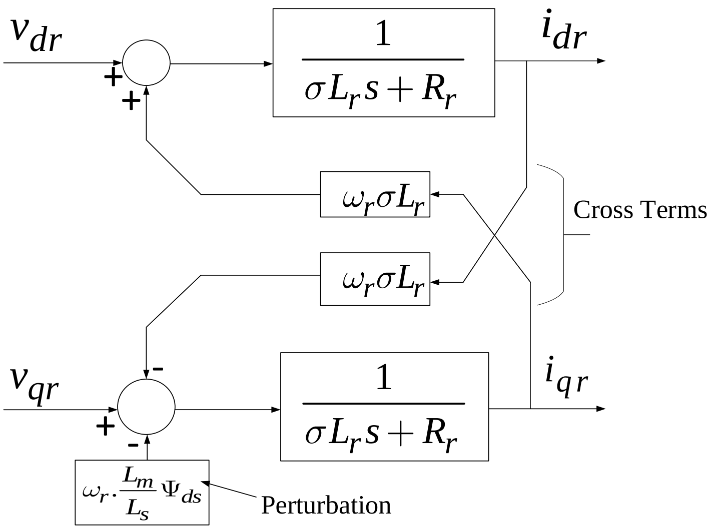


Figure 3. 14: Plant of the current control loop

The cross terms depicted in the middle of the Figure appear because the reference frame turns at a different speed than the rotor terminals. These terms are permanently constant and do not affect the functioning of the control loops however they are estimated and compensated for by the feed-forward controller as shown in Figure 3.15 to notably reduce its negative effects during transitory stages. After compensation of the cross terms the control loop is significantly simplified. Thus, both axes are now identical and their plant is reduced to a first-order transfer function plus, eventually, a current sensor filter transfer function as illustrated in Figure 3.16 (Abad et al. 2011; Ayrir, Meriem, and Haddi 2017).

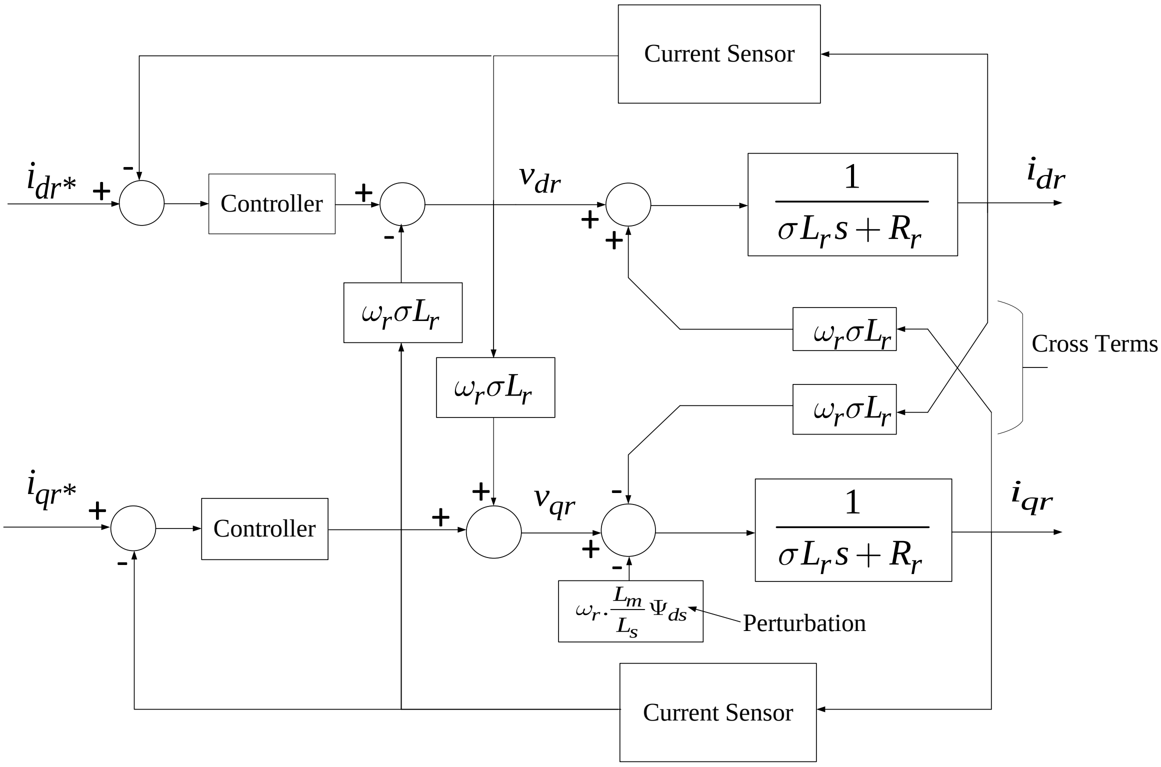


Figure 3. 15: Current control loop with feed-forward compensation of the cross terms

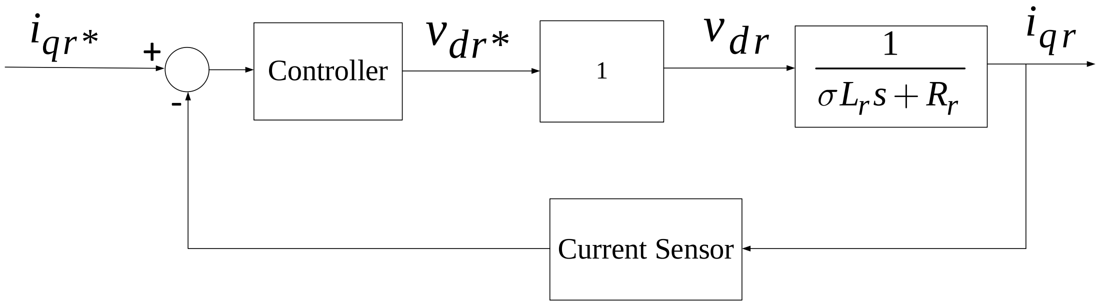


Figure 3. 16: Block diagram of the current control system

3.5.6 Rotor Side Converter Control

Controllers are incorporated into the system to make the system response follow the reference. Thus, in this work PI, fuzzy, fuzzy-PID and ANFIS controllers are utilized to control the rotor current of the DFIG, thus the design of these controllers is discussed in the next section. The controllers are set up to control both the direct and quadrature axis current to control them independently as shown in Figure 3.15. The ultimate goal of the rotor current control in each axis is to eliminate the gap between the reactive power and speed reference and the measured active and reactive power indirectly. To realize these objectives, the stated controllers are studied and designed (Ayrir et al. 2017; Hasnae Elouatouat 2020). The objective of this work is to improve the system performance as precision, stability, rapidity, and robustness by developing a superb control strategy.

3.5.6.1 Proportional Integral Control

The PID controllers are applied to the closed-loop system to improve a system's performance. This can be achieved by adjusting the parameters of the PID controller, such as proportional, integral, and derivative gains to obtain the desired response. Each gain of the PID controller has specific functions, the proportional gain is used for fast-rising time responses, Integral gain for minimizing errors, and derivative gain is used to minimize the error, reduce overshoot, and undershoot. A block diagram used in the control system for the implementation of a closed-loop system is shown in the following Figure 3.17 (Control Tutorials for MATLAB and Simulink - Introduction: PID Controller Design n.d.).

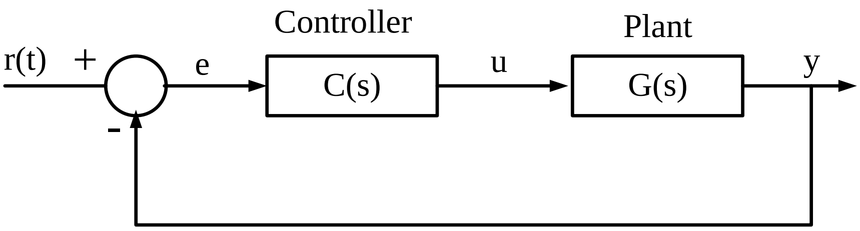


Figure 3. 17: Unity-feedback Control System Loop

The following equations represent the parameters of the figure:

$$e(t) = r(t) - y(t) \tag{3.49}$$

$$u(t) = K_p e(t) + K_i \int e(t) + K_d \frac{d}{dt} e(t)$$

$$C(s) = K_p + \frac{K_i}{s} + K_d s \tag{3.50}$$

$$y(t) = \frac{C(s)G(s)}{1 + C(s)G(s)} r(t)$$

The transfer function of both direct and quadrature axis current is similar;

$$G(s) = \frac{1}{s\sigma L_r + R_r} \tag{3.51}$$

In this work, a variable speed wind turbine rated at 2 MW Mitsubishi design is studied. Thus substituting for the parameters, σ, L_r, R_r of the transfer function in equation 3.51 the following system transfer function is obtained.

$$G(s) = \frac{1}{1.7107e-4s + 0.0029} \tag{3.52}$$

Figure 3.18 illustrates PI based current control loop

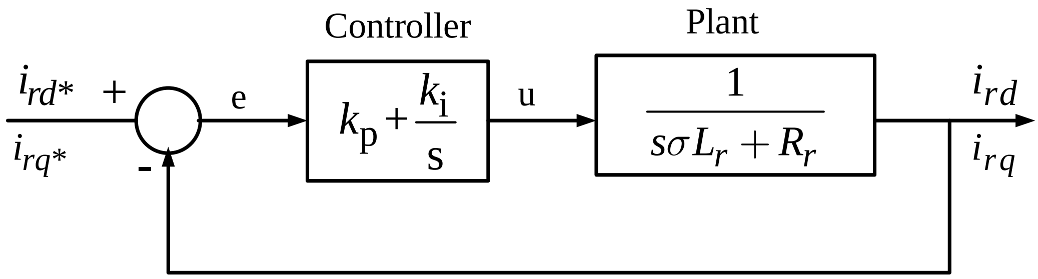


Figure 3. 18: PI based Current Control Loop

The open loop transfer function of the rotor current control loop is derived as:

$$G_o(s) = \left(k_p + \frac{k_i}{s}\right)\left(\frac{1}{s\sigma L_r + R_r}\right) = \frac{k_p s + k_i}{\sigma L_r\, s^2 + R_r s} \tag{3.53}$$

Then the closed loop transfer function is obtained:

$$G_c(s) = \frac{G_o(s)}{1 + G_o(s)} \tag{3.54}$$

$$\Rightarrow \frac{i_{dr}}{i_{dr*}} = \frac{i_{qr}}{i_{qr*}} = \frac{sk_p + k_i}{\sigma Lr\, s^2 + (k_p + R_r)s + k_i} \tag{3.55}$$

The system transfer function is second order, and the general second-order transfer function is given by (Nise 2015):

$$G(s) = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} \tag{3.56}$$

Equation 3.55 can be developed to the form of general second-order transfer function as:

$$\Rightarrow \frac{i_{dr}}{i_{dr*}} = \frac{i_{qr}}{i_{qr*}} = \frac{sk_p + k_i \,/\, \sigma L_r}{s^2 + \left(\frac{k_p + R_r}{\sigma L_r}\right)s + \frac{k_i}{\sigma L_r}} \tag{3.57}$$

Comparing equation 3.56 with 3.57 we obtain:

$$2\zeta\omega_n = \frac{k_p + R_r}{\sigma L_r}$$

$$\omega_n^2 = \frac{k_i}{\sigma L_r} \tag{3.58}$$

The parameters of the PI controller can be found using pole compensation technique. Consider the damping ratio $\zeta = 1$, which means that the system is critically damped. Then, the current the parameters of PI controllers can be expressed as:

$$k_p = 2\sigma L_r \omega_n - R_r$$
$$k_i = \omega_n^2 \sigma L_r \tag{3.59}$$

The value of the natural frequency, ω_n , is the frequency of oscillation of this system, which can be using the transient response performance specifications. When unity step input, r(t)=1 is applied to the system, the transient response performance specifications are related by (Nise 2015):

$$T_s = \frac{4}{\xi\omega_n} = 4T_c$$
$$\Rightarrow \omega_n = \frac{1}{T_c} \tag{3.60}$$

For step input, r(t)=100, for this case:

$$\omega_n = \frac{100}{T_c} \tag{3.61}$$

The values of the PI controller parameters k_P and k_i for both rotor current components obtained based on this method are $k_P = 0.5771$ and $k_i = 491.5995$.

The transfer function of the outer speed control loop is represented by equation 3.62.

$$G(s) = \frac{1}{\frac{J}{p}s + D} \tag{3.62}$$

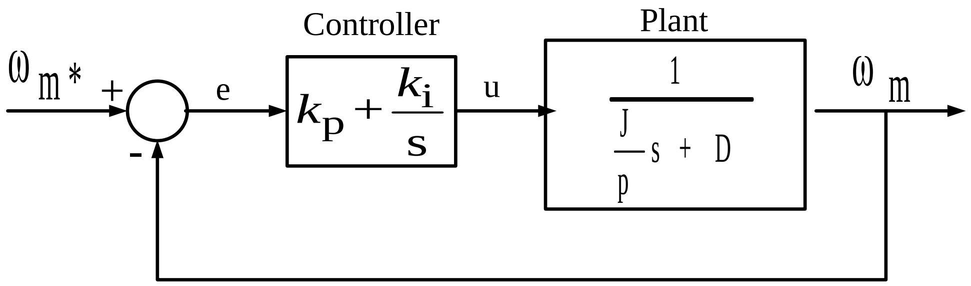


Figure 3. 19: PI-based Speed Control Loop

Figure 3.19 illustrates a PI-based speed control loop, for which the open-loop transfer function of the rotor current control loop is derived as:

$$G_o(s) = \left(k_p + \frac{k_i}{s}\right)\left(\frac{1}{\frac{J}{p}s + D}\right) = \frac{k_p s + k_i}{\frac{J}{p}s^2 + Ds} \tag{3.63}$$

Then the closed loop transfer function is obtained:

$$G_c(s) = \frac{G_o(s)}{1+G_o(s)} \tag{3.64}$$

$$\frac{\omega_m}{\omega_{m*}} = \frac{sk_p + k_i}{\frac{J}{p}s^2 + (k_p + D)s + k_i} \tag{3.65}$$

$$\Rightarrow \frac{\omega_m}{\omega_{m*}} = \frac{sk_p + k_i \Big/ J/p}{s^2 + \left(\frac{k_p + R_r}{J/p}\right)s + \frac{k_i p}{J}} \tag{3.66}$$

The system transfer function of equation 3.66 is of second order form, thus comparing it with general second-order transfer function of 3.56 the following relations are obtained:

$$2\zeta\omega_n = \frac{k_p + R_r}{J/p}$$

$$\omega_n^2 = \frac{k_i p}{J} \tag{3.67}$$

The parameters of the PI controller can then be found in a similar way as that of current control loop. Thun, the parameters of PI controllers are expressed as:

$$k_p = 2\omega_n \frac{J}{p} - D$$

$$k_i = \omega_n^2 \frac{J}{p} \tag{3.68}$$

The values of the PI controller parameters k_P and k_i for the speed control loop obtained are thus, $k_P = 5080$ and $k_i = 203200$.

3.5.6.2 Fuzzy Logic Control

Fuzzy logic control has gained significant attention over the last few years for WECS due to its capabilities of effectively controlling non-linear systems. It provides robust control against parameter uncertainties and fast tracking without the need of the mathematical model of the system, just by incorporating expert's knowledge (Hasnae Elouatouat 2020).

The design of a fuzzy logic control requires the appropriate selection of the size of the rule-base, the universe of input and output variables, the shape and the ranges of the membership functions (MFs), the scaling factor, and the rule inference mechanism. Figure 3.20 shows the basic configuration of fuzzy logic controller, which has two inputs (i.e. error, e (t), and error change, Δe(t)) and one output (i.e. control signal u(t)). The controller is composed of the fuzzification inference, the knowledge base, the inference engine, and defuzzification interface parts (Belkacem and Ouriagli 2019).

Crisp values of the error signals e(t) and derivativae e(t) are converted to normalized inputs {e_n(t), Δe_n(t)}E[-1, 1]; through their scaling factors G_e and $G_{\Delta e}$, respectively. The fuzzification stage transforms the normalized input variables to fuzzy ones using the fuzzy linguistic variables and corresponding MFs to process them according to the rule bases formulated. Fuzzy inference engine uses the fuzzified inputs and rules bases to generate fuzzy outputs. The fuzzy rules usually uses IF THEN expressions to describe several fuzzy conditional statements that are formed according to experts knowledge (Parthasarathy et al. 2017). In defuzzification interface, the fuzzy outputs are converted into crisp values by center of gravity (COG) method. Finally, the output control signal u(t) is derived by multiplying the Δu(t) by output scale factor $G_{\Delta u}$ (Kasbi and Rahali 2020).

The selection of appropriate values of scaling factors (G_e, $G_{\Delta e}$ and $G_{\Delta u}$) inside fuzzy control system is crucial in the design of the controller to ensure a better control behavior. According to the reviewed papers the scaling factors are estimated based on the approximations, which is based on the linearization of the fuzzy operator, which allows us to make the equivalence between the control action of classical PI controller and FLC (Kasbi and Rahali 2020).

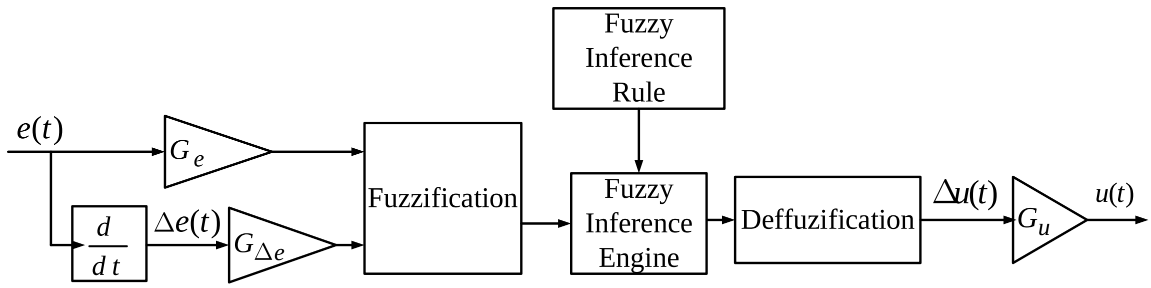


Figure 3. 20: Basic Structure of FLC

From the basic structure of FLC shown in Figure 3.20 it can be observed that the final control signal u(t) is defined as a nonlinear function of e(t) and Δe(t) ; and it gives:

$$u(t) = G_u \int_0^t \Delta u(t) dt = G_u \int_0^t \left[f(e(t), \frac{de(t)}{dt} \right] dt \quad (3.69)$$

Where the function f is the fuzzy input-output law of the fuzzy controller. The linearization of fuzzy operator has allowed to estimate the scaling factors which weigh the input and output variables e(t) , Δe(t) and Δu(t) from the gains k_p and k_i of classical PI controller. Based on the linear approximation the output control law Δu(t) by defuzzification is reduced to:

$$\Delta u(t) \approx E(t) + \Delta E(t) \tag{3.70}$$

Thus, the output control law from FLC becomes:

$$u(t) = G_u \int_0^t \left[E(t) + \frac{dE(t)}{dt} \right] dt \tag{3.71}$$

$$u(t) = G_u \int_0^t \left[G_e.e(t) + G_{\Delta e}.\frac{de(t)}{dt} \right] dt \tag{3.72}$$

$$u(t) = G_u.G_e \int_0^t e(t)dt + G_u.G_{\Delta e}.e(t) \tag{3.73}$$

Comparing the output signal equation 3.73 of FLC with that of classical PI controller equation, so the following relation can be obtained:

$$k_p = G_u.G_{\Delta e} \Rightarrow G_{\Delta e} = k_p / G_u \tag{3.74}$$
$$k_i = G_u.G_e \quad \Rightarrow G_e = k_i / G_u$$

From pole compensation method k_p and k_i were found to be 0.5771 and 491.5775 respectively, thus using these values and fixing the value of G_u to be any number, 230 in this case, then G_e and $G_{\Delta e}$ will become 0.96631 and 0.002 respectively. Hence, the fuzzy logic controller is designed as illustrated in Figure 3.21.

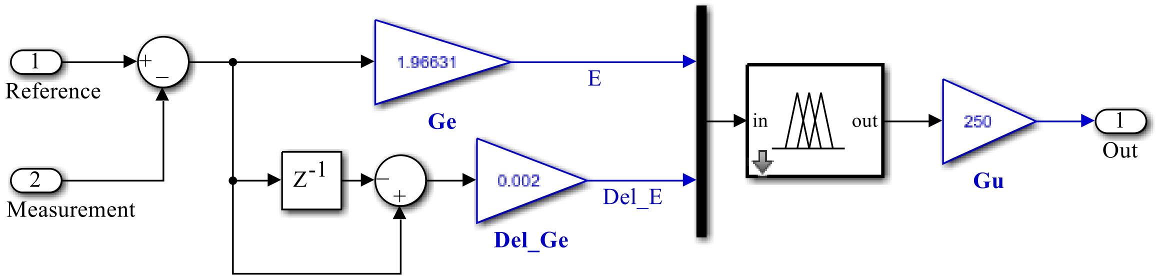


Figure 3. 21: Fuzzy logic controller for the rotor current components

In this thesis, the fuzzy inference engine uses mamdani method due to its simplicity in the fuzzy rules formulation. The membership functions for two inputs (i.e. the system error and change in error), and output (i.e. the control signal), are designed to satisfy the best response results. The range of both inputs is chosen to be [-40,000 40,000] and the output range is set to [-90 90] as exhibited in Figure 3.22, and 3.23 respectively. The design consists of seven triangular membership functions for the inputs and nine triangular membership functions for the output. The rules for the fuzzy logic controller is shown in Table 3.1. A linguistic variable that used in inputs and output are NB: Negative Big, NM: Negative Medium, NS: Negative Small, NVS: Negative Very Small, ZE: Zero, PS: Positive Small, PM: Positive Medium, PB: Positive Big, and PVS: Positive Very Small.

Table 3. 1: Fuzzy rules for fuzzy logic controller

ez^{-1} \ e	NB	NM	NS	ZE	PS	PM	PB
NB	NB	NB	NB	NM	NS	NVS	ZE
NM	NB	NB	NM	NS	NVS	ZE	PVS
NS	NB	NM	NS	NVS	ZE	PVS	PS
ZE	NM	NS	NVS	ZE	PVS	PS	PM
PS	NS	NVS	ZE	PVS	PS	PM	PB
PM	NVS	ZE	PVS	PS	PM	PB	PB
PB	ZE	PVS	PS	PM	PB	PB	PB

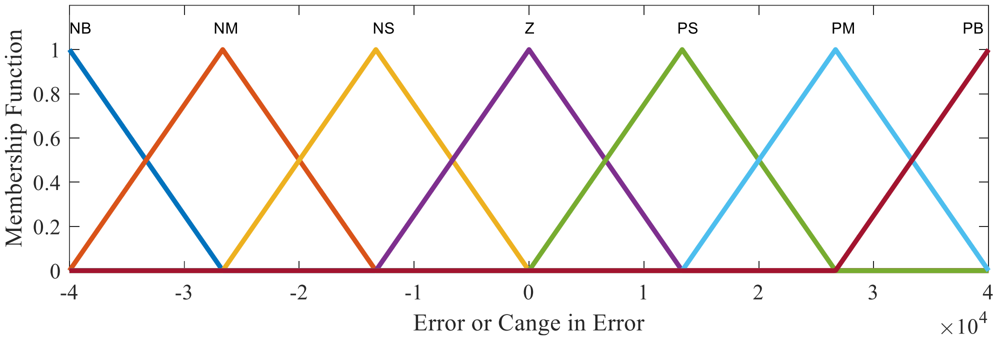


Figure 3. 22: Input membership functions for fuzzy controller

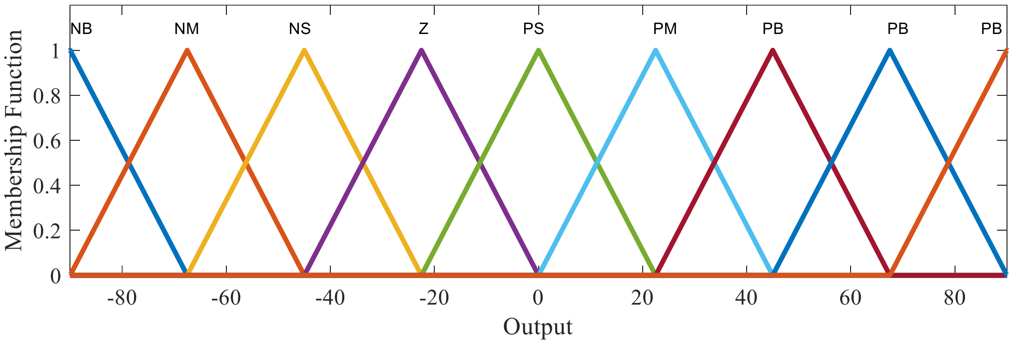


Figure 3. 23: Output membership functions for fuzzy controller

3.5.6.3 Fuzzy-PI Controller

Fuzzy PI controller is a hybrid controller that employs PI gains to create a nonlinear response (Helal 2018). In this work, suitable PI gains were selected using pole compensation tuning method. The obtained PI gains are, k_p=0.5771 and k_i=491.5995. The rotor current error and the one-step delayed rotor current error were chosen to be an input to fuzzy logic controller. The set of rules represented in Table 3.2, were constructed for fuzzy-PI control block. The columns refer to the error(t) whereas the rows refer to change of error signal $\Delta e(\mathrm{t})$. Each pairs (e(t),de(t)) determines the output level u(t). Similar to the proposed FLC design, the Fuzzy PI membership functions of input e(t) are identical to those of Δe(t).

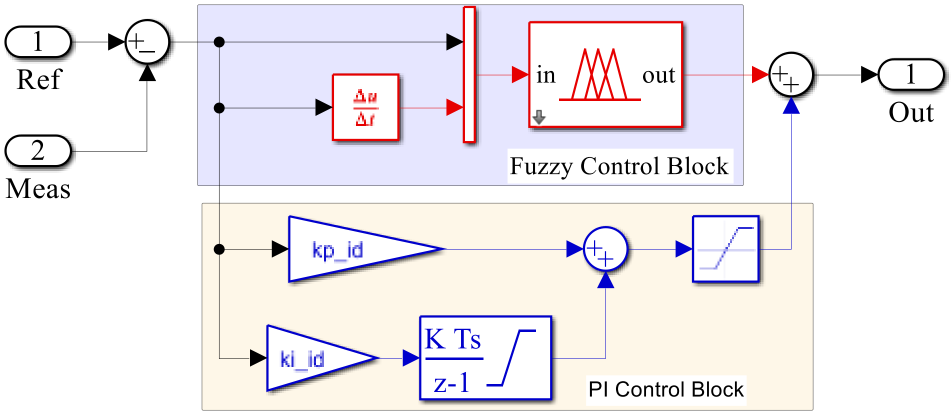


Figure 3. 24: Fuzzy-PI controller for the rotor current components

The fuzzy-PI controller designed with the aforementioned inputs is illustrated in Figure 3.24. The membership functions are developed to produce the best system response. The range of both inputs and the output is set to [-80000 80000] and [-90 90] respectively found by trial. The developed membership function for the inputs and outputs is illustrated in Figure 3.25 and 3.26, respectively. The design consists of five membership functions for both inputs and output. The five membership functions consists of three Gaussian, a Z, and an S membership functions. The fuzzy rules shown in Table 3.2, where developed with variables; NB: Negative Big, N: Negative; Z: Zero, P: Positive and PB: Positive Big.

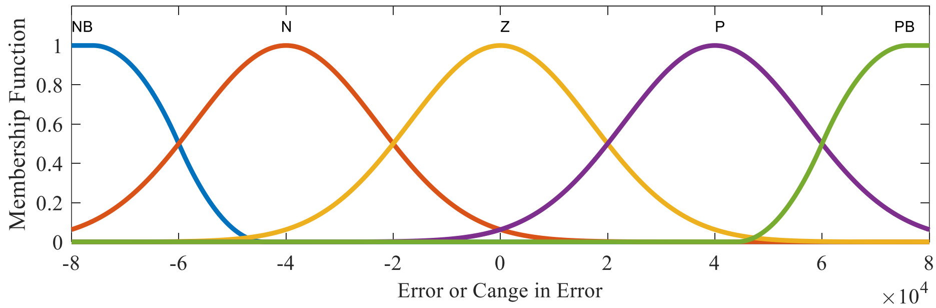


Figure 3. 25: Input membership functions for fuzzy-PI controller

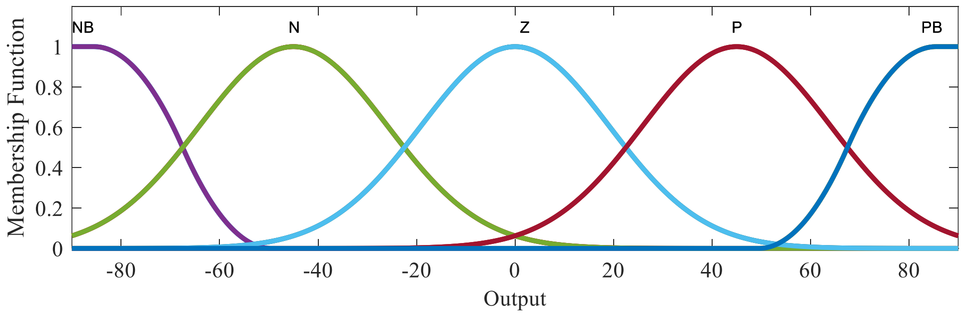


Figure 3. 26: Output membership functions for fuzzy-PI controller

Table 3. 2: Fuzzy rules for Fuzzy-PI controller

de \ e	**NB**	**N**	**Z**	**P**	**PB**
NB	NB	Z	NB	Z	Z
N	Z	N	N	Z	Z
Z	NB	N	Z	P	PB
P	Z	Z	P	P	Z
PB	Z	Z	PB	Z	PB

3.5.6.4 Adaptive Neuro-Fuzzy Inference System (ANFIS) Controller

Adaptive neural network has gained popularity currently in the nonlinear control system. Adaptive neuro-fuzzy inference system combines the advantages of both fuzzy logic and adaptive neural network. The combined effect of fuzzy and neural networks makes ANFIS an effective learner and interpreter. The membership functions in ANFIS controllers are created by using an authentic input-output data set. The parameter associated with the membership function values are obtained based on gradient descent algorithm. ANFIS networks utilize a least squares estimation, back propagation, and hybrid estimation methods for membership function parameter estimation (Amin, Uddin, and Marsadek 2019; Khan, Ali, and Shaikh 2022). The ANFIS utilizes Tackagi-Sugeno fuzzy logic control method. The general structure of ANFIS consists of five-layer which are shown in Figure 3.27 (I. Ahmad and Sharma 2018; Angeline et al. 2017).

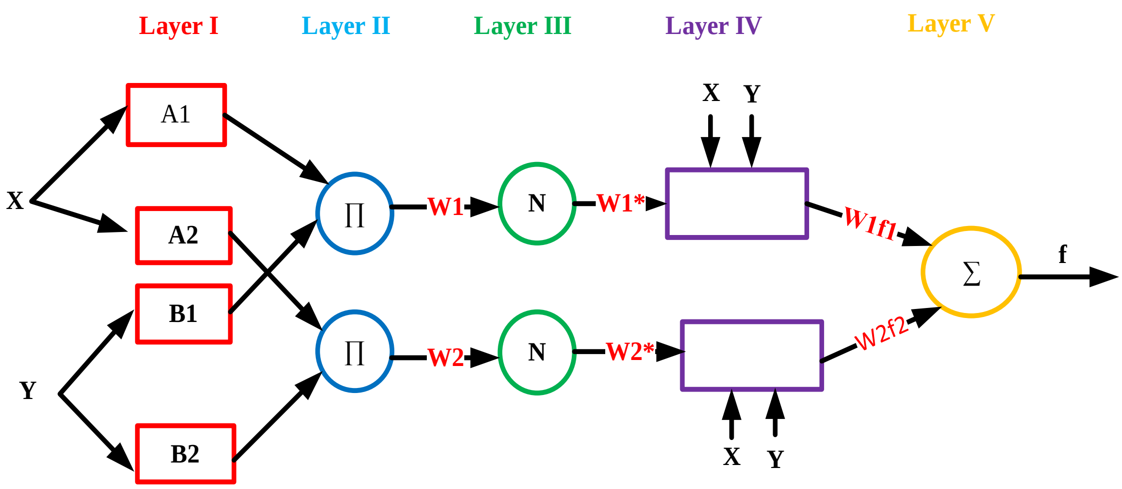


Figure 3. 27: Anfis Architecture

Layer 1- The first layer is also known as the fuzzification layer, a number of Gaussian membership functions are assigned to each input. This input layer maps to the next layer

Layer 2-This layer checks the weight of each MF. It calculates the value of MF that input values belong to which is sent to next layer as input.

Layer 3- Each node of these layers evaluate the weight of the normalized MF. The activation of each rule is compared with the number of the fuzzy rules in this layer.

Layer 4- This layer is a de-fuzzification layer that provides the output values as consequence of inference rules. In this layer, normalization is performed by each neuron.

Layer 5- This is the output layer, which adds all the inputs from layer-4 and transforms the fuzzy classification results into a crisp value(Amin, Uddin, and Marsadek 2019).

The ANFIS inputs are error e(t) and change in error Δe(t), $\Delta e(t) = e(t) - e(t-1)$ and the training data are prepared through the simulated data. The model structure and control block designed for the proposed ANFIS controller is depicted in Figure 3.28 and 3.29 respectively.

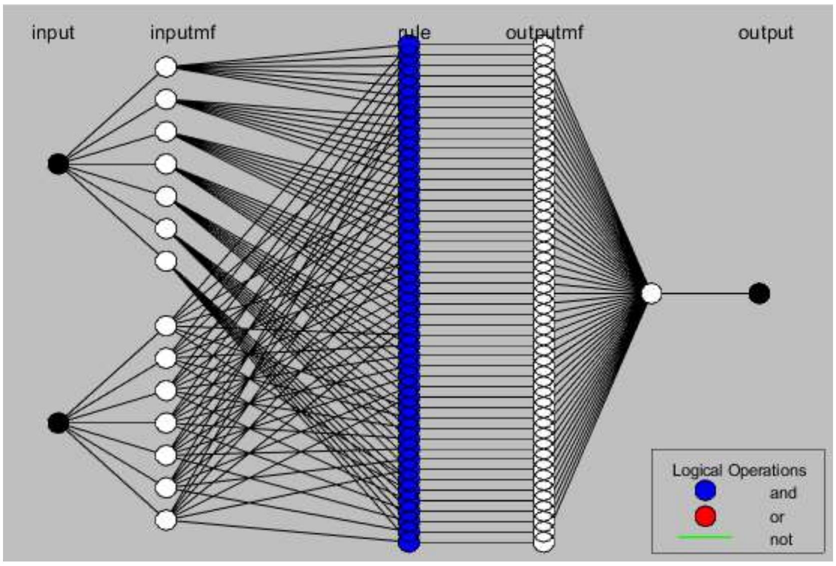


Figure 3. 28: ANFIS model structure

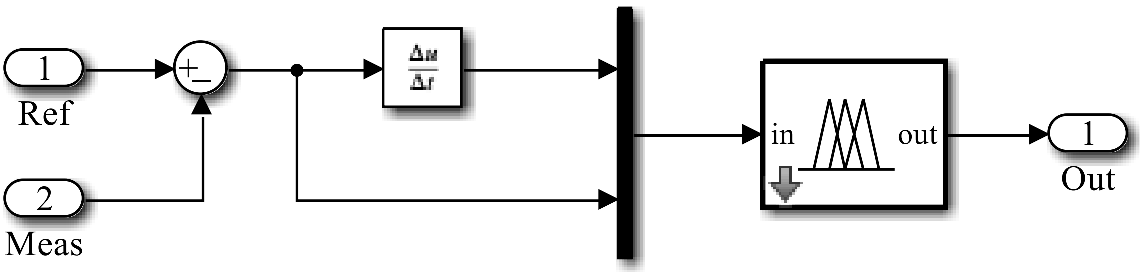


Figure 3. 29: ANFIS controller for the rotor current components

3.6 Simulation of the Modeled System

The wind speed and the wind turbine model that includes the aerodynamic, mechanical, electrical and control systems are simulated in MATLAB Simulink.

3.6.1 Wind Speed Model Simulation

The wind speed model is composed of the ramp component, average wind speed component, gust component, and turbulence component. Figure 3.30 represents the simulated wind speed model. The transfer function of the first order filter is represented by $\frac{300s}{300s+1}$ based on equation 3.5.

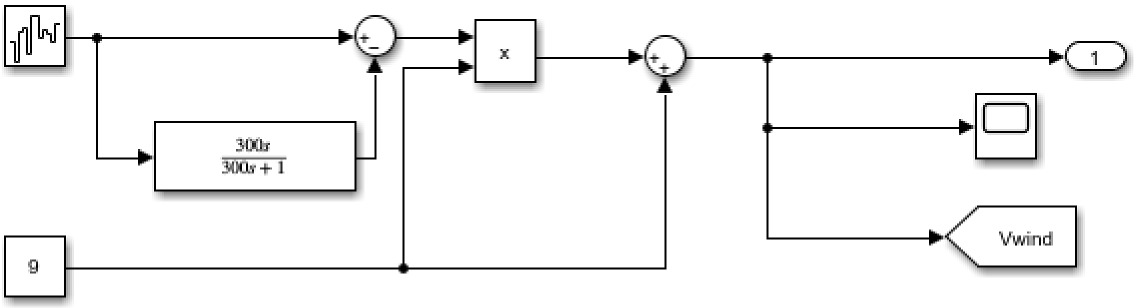


Figure 3. 30: Wind speed model at average speed of 9m/s

3.6.2 Aerodynamic Model Simulation

The aerodynamic model shown in Figure 3.31 is designed to evaluate the aerodynamic torque of the turbine shaft Tae based on equation 3.11 through 3.15.

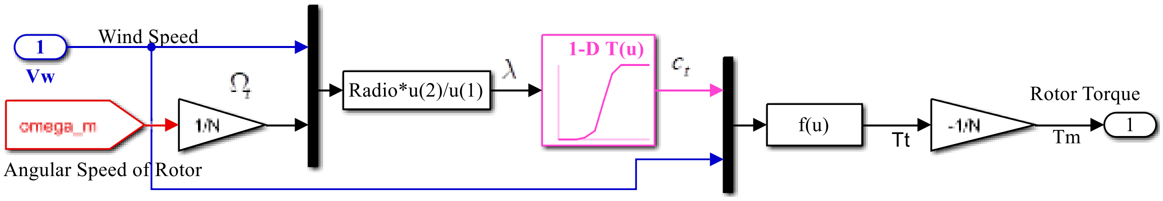


Figure 3. 31: Aerodynamic system simulation of a WT

3.6.3 Electrical Model Simulation

Figure 3.32 illustrates the electrical system model of the DFIG wind turbine, which has its rotor connected to the rotor side converter from the right, and its stator connected to the grid from the left. The grid side converter is replaced by a DC voltage source since grid side converter control was not part of the proposal.

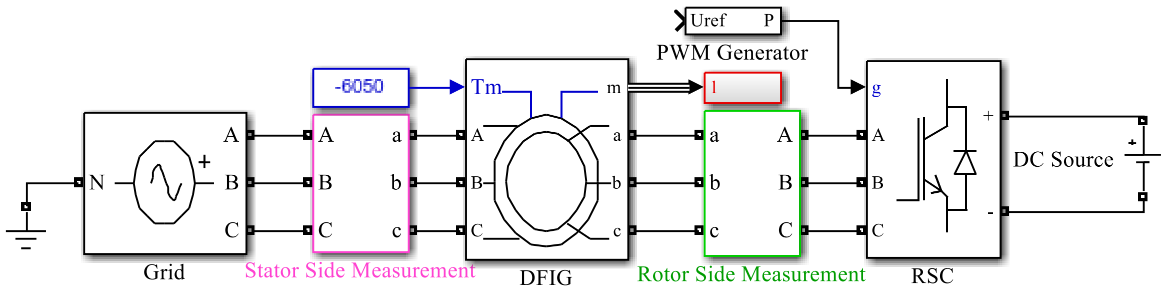


Figure 3. 32: Electrical system component of DFIG wind turbine

3.6.4. Control System Model Simulation

In this work, the control system with two control loops; the outer (speed control loop) and the inner(current control loop) is designed. The speed controller is designed to force the aerodynamic torque Tae to follow the maximum power curve in response to the wind variations and the vector controller is designed for current control loops to regulate the rotor side converter. Figure 3.33 shows the designed control system of the DFIG wind turbine.

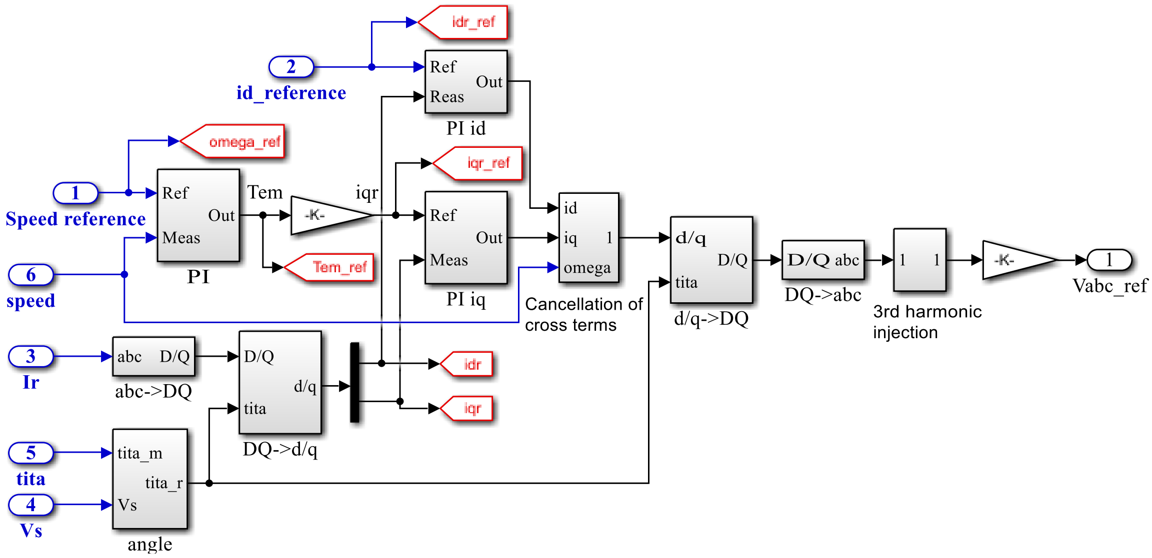


Figure 3. 33: Control system model simulation of a DFIG wind turbine

3.6.5 Overall Wind Turbine Dynamic Model Simulation

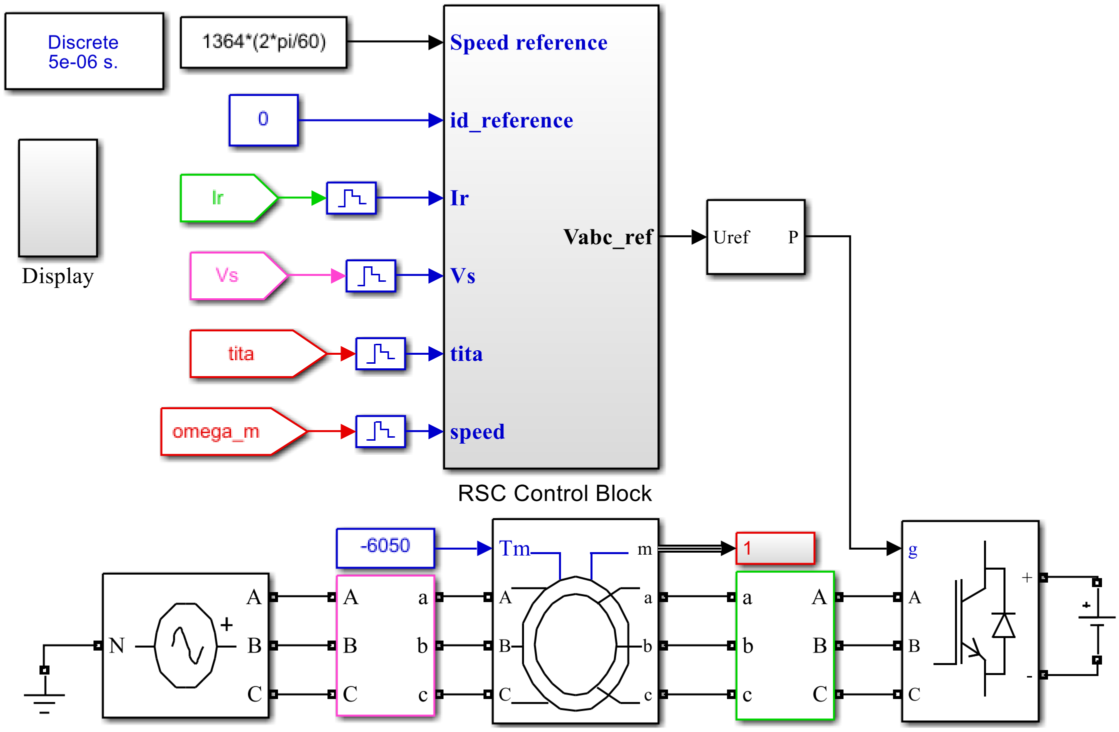


Figure 3. 34: Overall WT dynamic model design in Simulink

Figure 3.34 represents the overall system Simulink design. The design encompasses the DFIG that is equipped with a wound rotor induction generator, voltage source converter connected to the slip rings of the rotor, the grid that is directly coupled to the stator winding, and the power converter connected to the rotor winding.

3.7 Steady State Analysis

Table 3.3 illustrates the 2 MW DFIG parameters as per the manufacturers design (Abad et al. 2011). These parameters are applied to evaluate the steady state magnitudes of the DFIG parameters depicted in Table 3.4, with $i_{dr} = 0$.

Table 3. 3: 2 MW DFIG Specifications

Stator frequency (Hz)	f=50
Rated stator power (W)	Ps=2 × 106
Rated rotational speed (rev/min)	n=1500
Rated stator voltage (V)	Vs=690
Rated stator current (A)	Is=1760
Rated torque (N.m)	Tem=12732
Pole pair	p=2
Stator/rotor turns ratio	u=1/3
Rated rotor voltage (V)	Vr=2070
Maximum slip	smax=1/3
Rated rotor voltage referred to stator (V)	Vr_stator=(Vr*smax)*u
Stator resistance (ohm)	Rs=2.6 × 10-3
Leakage inductance (stator/rotor) (H)	Lsi=0.087 × 10-3
Magnetizing inductance (L)	Lm=2.5 × 10-3
Rotor resistance referred to stator (ohm)	Rr=2.9 × 10-3
Stator inductance (H)	Ls=Lm+Lsi
Rotor inductance (H)	Lr=Lm+Lsi
DC bus voltage referred to stator (V)	Vbus=Vr_stator× √2

Table 3. 4: Procedure to derive the steady-state magnitudes with Q_s and i_{dr} reference

Given grid voltage: ω_s given operating point: Q_s ω_m T_{em}				
Stator flux	$\left\|\vec{\psi}_s\right\| = \sqrt{\frac{-B \pm \sqrt{B^2 - 4AC}}{2A}}$	$C = \left(\frac{2}{3}\frac{R_s T_{em}}{\rho L_M}\right)^2$	$B = \frac{4}{3}\frac{R_s T_{em}\omega_s}{\rho} - \left\|v_s\right\|^2$	$A = \left(\frac{R_s}{L_s}\right)^2 + \omega_s^2$
Stator currents	$i_{dr} = 0$	$i_{qr} = T_{em} \Big/ \left(-\frac{3}{2}\rho\frac{L_m}{L_s}\left\|\vec{\psi}_s\right\|\right)$	$\left\|\vec{i}_r\right\|^2 = i_{dr}^2 + i_{qr}^2$	$\theta_{i_s} = a\tan\left(\frac{i_{qs}}{i_{ds}}\right)$
Rotor currents	$i_{ds} = \left\|\vec{\psi}_s\right\| \big/ L_s$	$i_{qs} = -(L_m/L_s)\, i_{qr}$	$\left\|\vec{i}_s\right\|^2 = i_{ds}^2 + i_{qs}^2$	$\theta_{i_s} = a\tan\left(\frac{i_{qs}}{i_{ds}}\right)$
Stator voltages	$v_{ds} = R_s i_{ds}$	$v_{qs} = R_s i_{qs} + \omega_s\left\|\vec{\psi}_s\right\|$	$\left\|\vec{v}_s\right\|^2 = v_{ds}^2 + v_{qs}^2$	$\theta_{v_s} = a\tan\left(\frac{v_{qs}}{v_{ds}}\right)$
Slip	$\omega_s = \omega_s - \omega_m$		$s = \omega_r/\omega_s$	
Rotor voltages	$v_{dr} = R_r i_{dr} - \omega_r\sigma L_s i_{qr}$	$v_{qr} = R_r i_{qr} + \omega_s\sigma L_s i_{dr} + \omega_r\frac{L_m}{L_s}\left\|\vec{\psi}_s\right\|$	$\left\|\vec{v}_f\right\|^2 = v_{dr}^2 + v_{qr}^2$	$\theta_{vf} = a\tan(\frac{v_{qr}}{v_{dr}})$
Rotor fluxes	$\psi_{dr} = L_m i_{ds} + L_r i_{dr}$	$\psi_{qr} = L_m i_{qs} + L_s i_{qr}$	$\left\|\vec{\psi}_f\right\|^2 = \psi_{dr}^2 + \psi_{qr}^2$	$\theta_{\psi f} = a\tan(\frac{\psi_{qr}}{\psi_{dr}})$
Active powers	$P_m = T_{em}\frac{\omega_m}{p}$	$P_s = \frac{3}{2}(v_{ds}i_{ds} + v_{qs}i_{qs})$	$P_r = \frac{3}{2}\left(v_{dr}i_{dr} + v_{qr}i_{qr}\right)$	
Reactive power	$Q_s = \frac{3}{2}(v_{qs}i_{ds} - v_{ds}i_{qs})$	$PF_S = \cos(a\tan(Q_s/P_s))$	$Q_r = \frac{3}{2}\left(v_{qr}i_{dr} + v_{qr}i_{qr}\right)$	$Q_s = \frac{3}{2}\left(v_{qs}i_{ds} - v_{ds}i_{qs}\right)$
Efficiency	$\eta = \frac{P_m}{P_s + P_r} \cdots if\ P_m > 0$		$\eta = \frac{P_s + P}{P_m} \cdots if\ P_m < 0$	

The DFIG rotor speed was set to be in between 900 and 1800 rpm, and based on this the other parameters are calculated. Figure 3.35 illustrates the steady state characteristics of a 2 MW DFIG. From the figure, the electromagnetic rotor torque Tem is negative which indicates that the Doubly fed Induction Machine is operating as a generator. The total mechanical power Pt of the shaft is the product of the rotor speed and the electromagnetic torque. The rotor active power Pr is found to be much smaller than stator active power Ps. For instance, the maximum rotor active power is around 400 kW, but the maximum stator active power is about 2 MW. The negative and positive values of the active power indicates the direction of its flow as depicted in figure 3.4. Both the stator and rotor currents are directly related to the rotational speed. The stator voltage is always around 690 V since it is directly connected to the grid, according to the Mitsubishi design. Whereas, the rotor voltage depends on the rotational speed and the minimum rotor voltage occurs at the synchronous speed.

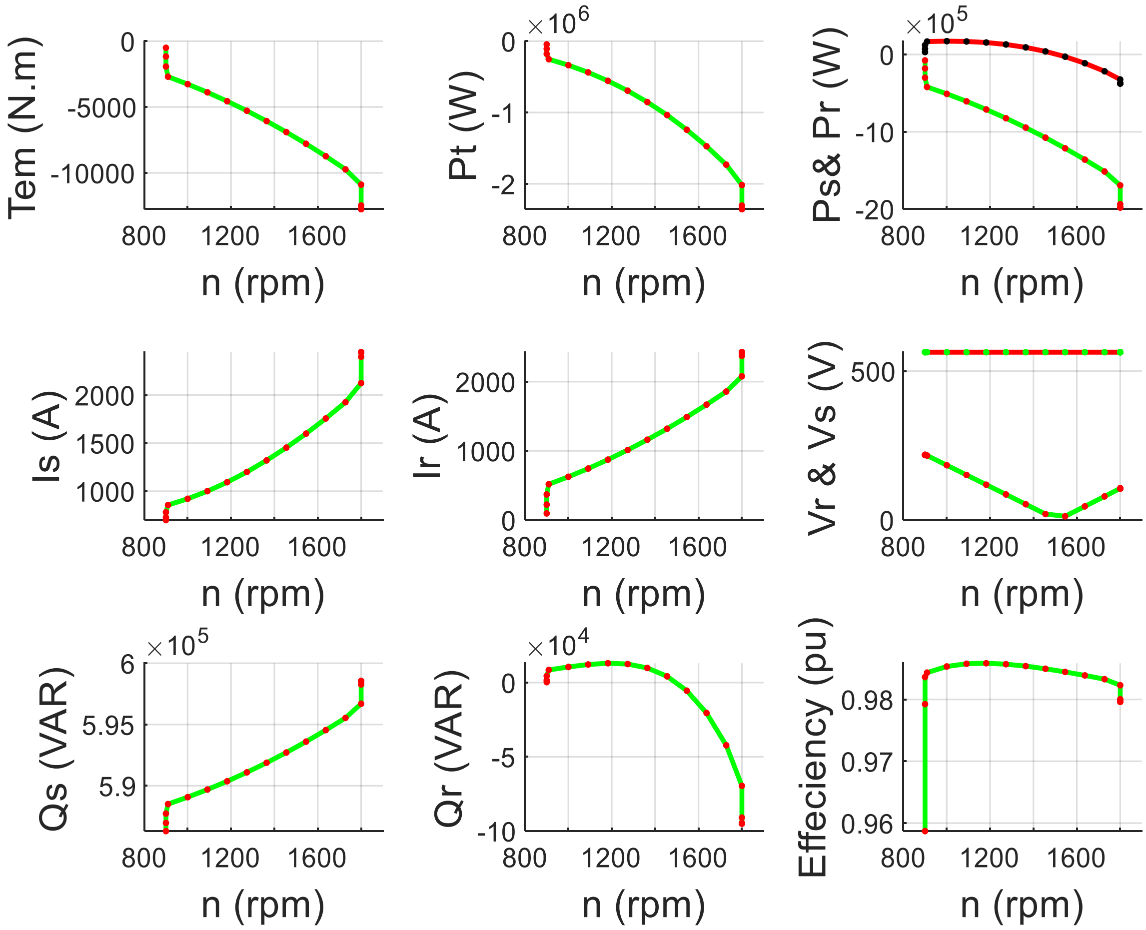


Figure 3. 35: Steady state representation of a 2 MW DFIG.

The rotor reactive power Qr is much smaller than the stator reactive power Qs. Finally, the efficiency of the machine (i.e. the mechanical power in the shaft to motor or generator operation) is almost constant throughout the rotational speed values (Abu-Rub 2014).

The steady-state characteristics of a 2 MW DFIG are presented in Figure 3.35 to set the evaluation platform for the designed system. The designed system is examined at a rotational speed of 1364 rev/min and the corresponding electromagnetic torque of -6050 Nm, and the values of Ir and Vr are compared with the steady-state characteristics of the plant at the set value to verify our design.

CHAPTER FOUR

4. RESULTS AND DISCUSSIONS

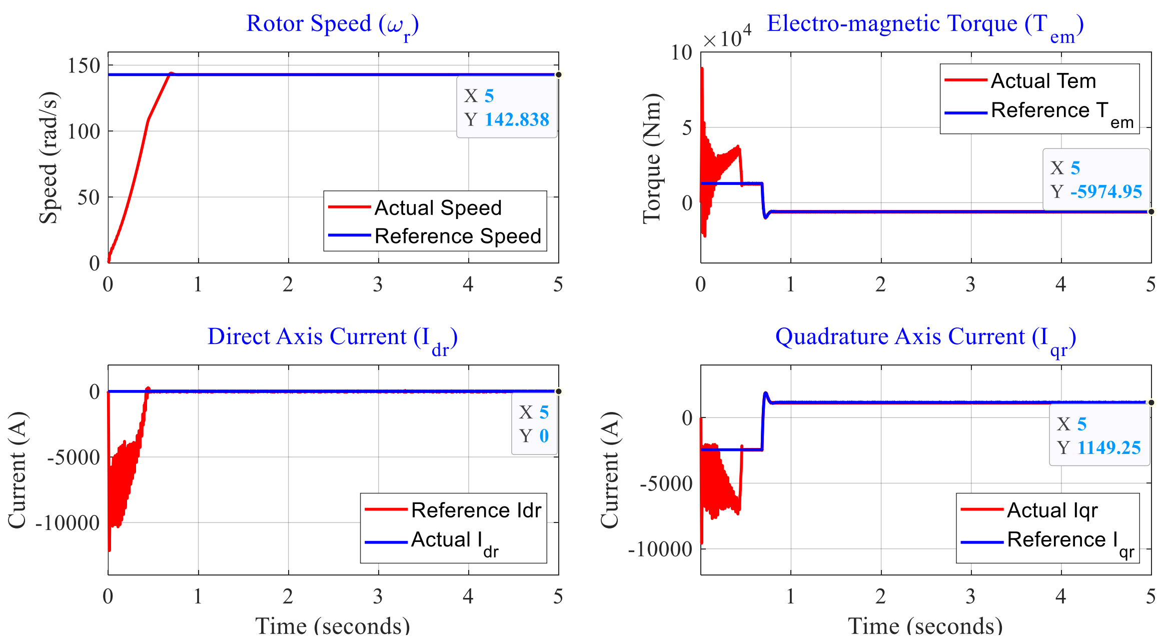
4.1 Simulation Results

In this section, the simulation results of the DFIG-based wind turbine system utilizing different control strategies are presented. The results are based on the modeled system in MATLAB/Simulink software in the material and methods chapter, section 3.6.

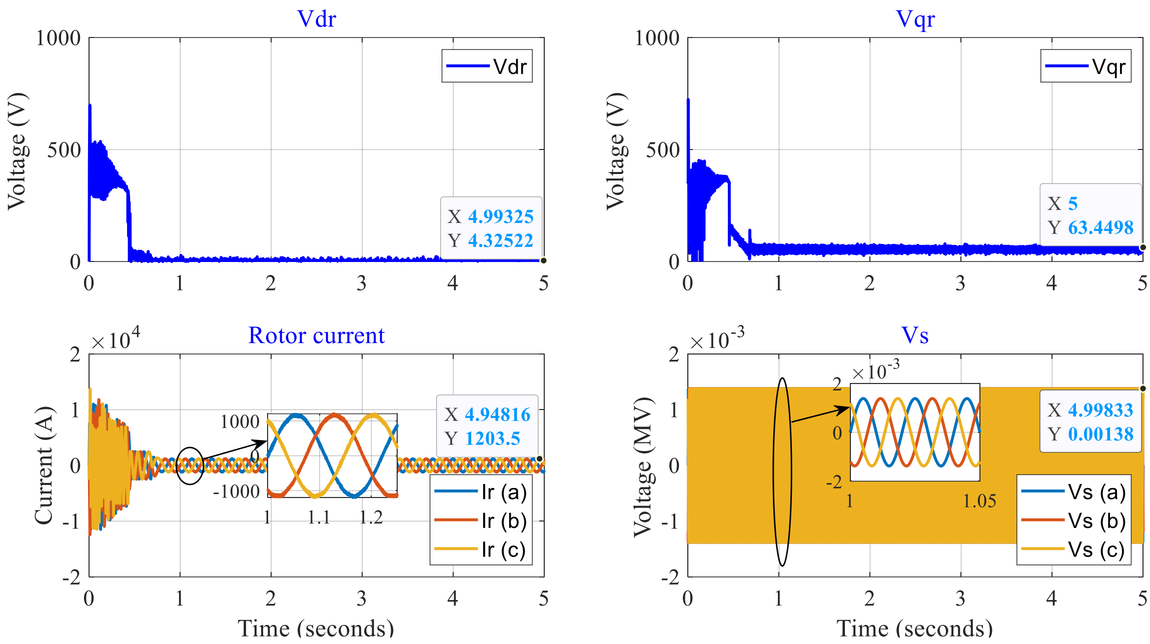
4.1.1 Simulation Results based on PI Controller

The developed system is evaluated at a rotating speed of 1364 rev/min and an associated electromagnetic torque of -6050 Nm by looking at the steady state performance of a 2 MW DFIG, as it was shown in Figure 3.35. Figure 4.1 represents the response of the simulated system at the mentioned reference values. It is noticed from the figure that the steady-state values of the DFIG parameters; rotor speed, torque, rotor current, and rotor voltage match the steady-state characteristics of the DFIG illustrated in Figure 3.35. The rotor speed at the steady state is 142.83788 rad/s, almost equivalent to the reference speed (1364 rev/min or 142.838 rad/s). The electromagnetic torque is -5978.95 Nm, which is so near the reference value.

For simpler management of the current in the q direction, which reaches the value of Iqr_ref in steady state, which corresponds to the peak value of the three-phase rotor current Ir, the rotor current in the d direction is made to go to 0 A (Abad et al. 2011).



(a)



(b)

Figure 4. 1: Steady-state response of the simulated system at a rotor speed of 142.8 rad/sec

It is examined that the value of *Iqr* (*Ir*) in Figure 4.1 is equivalent to the *Ir* value respectively of Figure 3.35. Finally, Equation 4.1 is used to validate the values of Vqr and Vdr. Therefore, it can be inferred from the results that the system is operating correctly and in accordance with the manufacturer's design, however its performances need to be improved.

$$V_r = \sqrt{V_{dr}^2 + V_{qr}^2} = \sqrt{(4.325)^2 + 63.45^2} = 63.6\,V \qquad (4.1)$$

The performance indices like the rise time, peak overshoot, peak undershoot, settling time, and steady-state error of the rotor speed, electromagnetic torque, *iq* and *id* components of the rotor current where the PI controller is used are presented in Table 4.1.

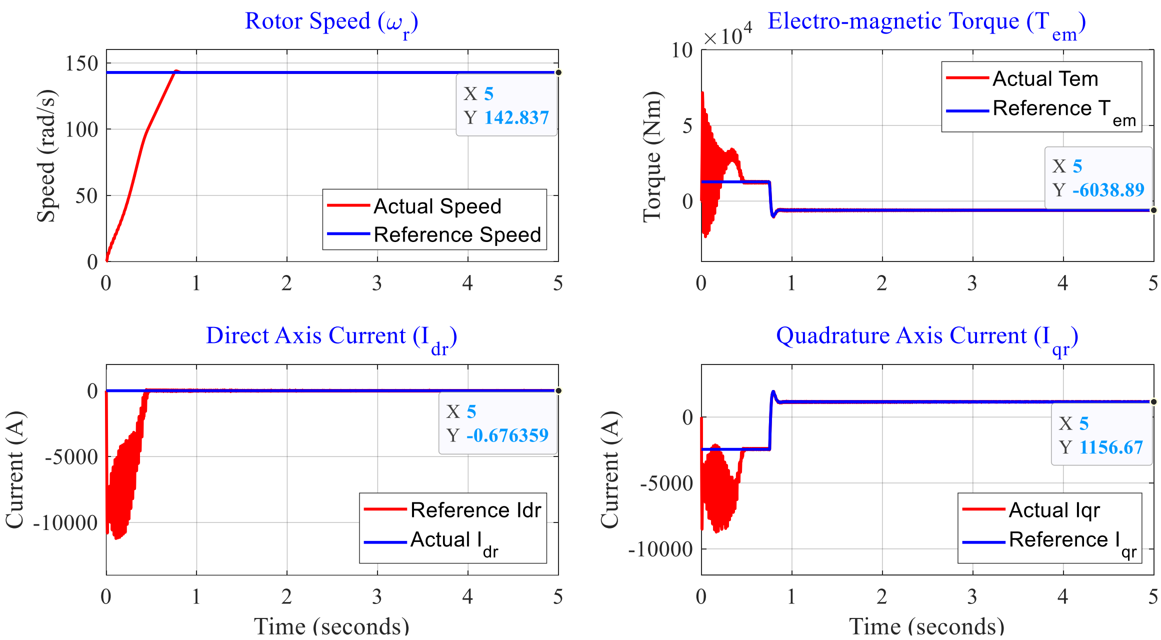
Table 4. 1: Time domain Performance analysis using a PI controller

Time domain Performance indices	Iqr	Idr	Rotor Speed	Torque (Nm)
Rise Time(s)	0.0072	0	0.5111	4.4986e-04
Settling Time(s)	0.7520	0.4419	0.6583	0.7439
Overshoot (%)	66.0579	-	0.8310	272.8187
Undershoot (%)	833.4304	-	0	1.4932e+03
Steady State error	-16.7453 A	-0.8734 A	1.1099e-04 (rad/sec)	71.1057 Nm

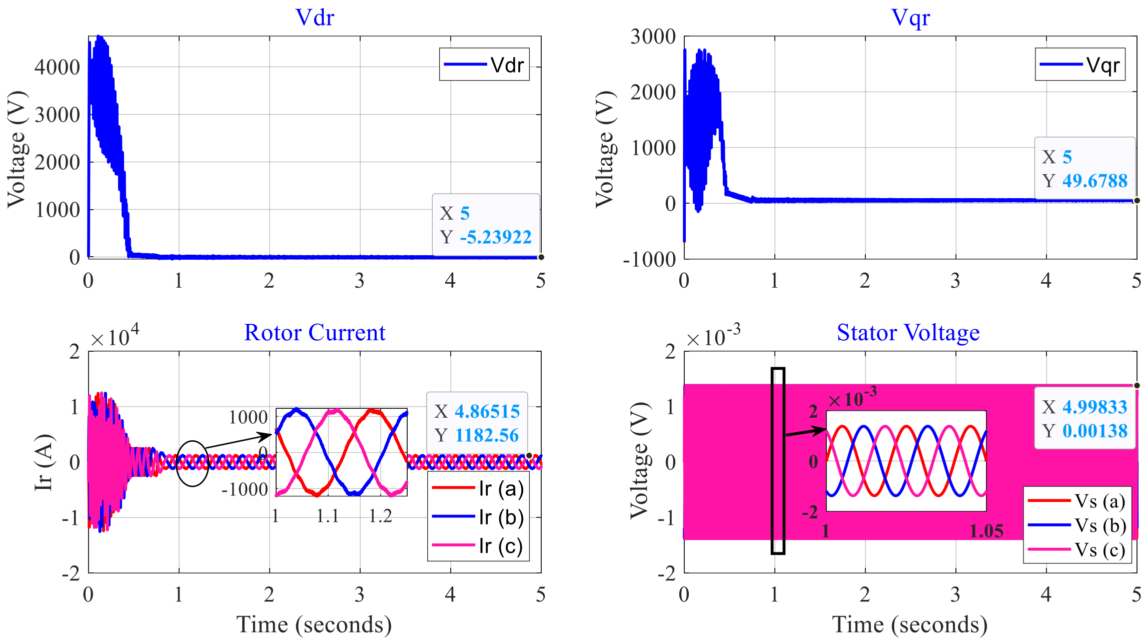
4.1.2 Simulation Results Utilizing Fuzzy Controller

As it has been examined in section 4.1, the system performances such as rise time, settling time, overshoot, undershoot, and steady-state error of *Idr* and *Iqr* when PI controller is used are not satisfactory. Thus, an intelligent controller i.e. fuzzy logic controller, considered an effective and efficient controller to enhance the system response is developed. The fuzzy logic controller illustrated in Figure 3.21, is developed to force the 2 MW DFIG to operate at manufacturer specifications (i.e. steady-state values of Figure 3.35).

It can be observed from the designed system response shown in Figure 4.2 i.e. the measurement values of the system match the steady-state values in Figure 3.35 tested at the reference speed of 1364 rad/s and -6050 Nm torque. Table 4.2 illustrates the system's time domain performance subject to the proposed fuzzy logic controller, from which the steady state error of the quadrature and direct currents are observed as being small. At the time of connection to load the system has large perturbation as observed in Figure 4.2, which is the limitation of this controller, however, this problem can be solved by using the proper starting mechanisms.



(a)



(b)

Figure 4. 2: Steady-state response of the simulated system at a rotor speed of 142.8 rad/sec based on fuzzy controller

Table 4. 2: Time domain Performance analysis using a fuzzy controller

Time domain Performance indices	Iqr (A)	Idr (A)	Rotor Speed (rad/sec)	Torque (Nm)
Rise Time	0.0069	0	0.5953	4.7001e-04
Settling Time	0.8260	0.4576	0.7339	0.8235
Overshoot	68.3055	-	0.8532	291.5128
Undershoot	754.3721	-	0	1.1909e+03
Steady State error	-0.5421	-0.6764	5.6884e-04	-22.5707

4.1.3 Simulation Result based on Fuzzy and PI controller

Figure 4.3 illustrates the system response for PI and Fuzzy logic controllers from which it can be observed that the system response for the fuzzy logic controller has fewer oscillations, smaller undershoot, and smaller peak values compared to the system response for the PI controller.

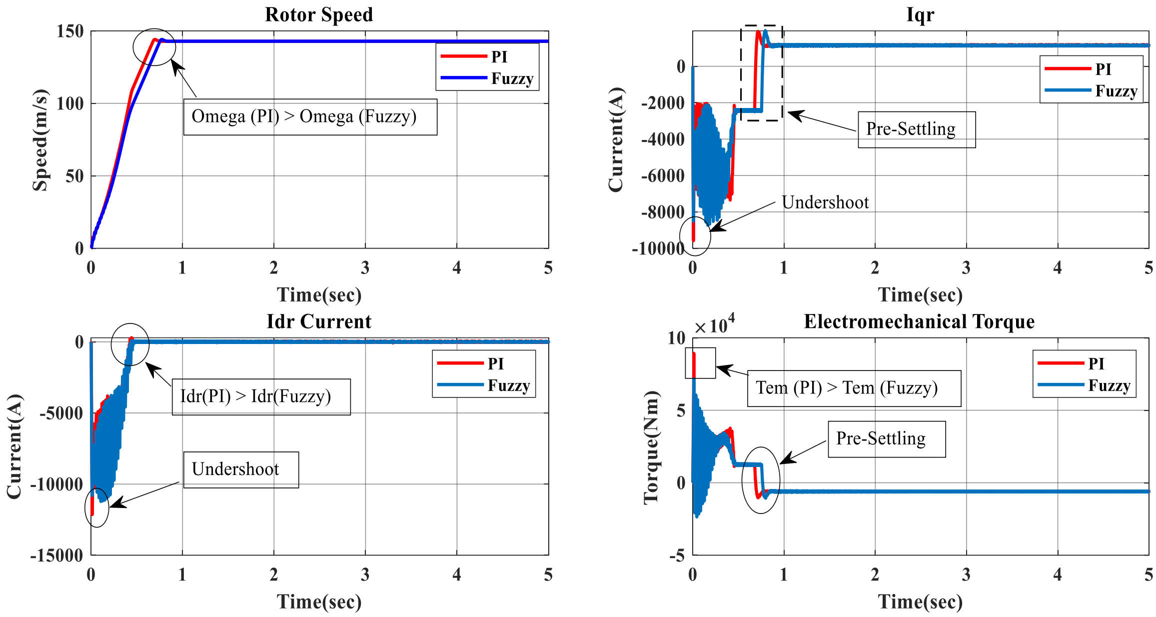


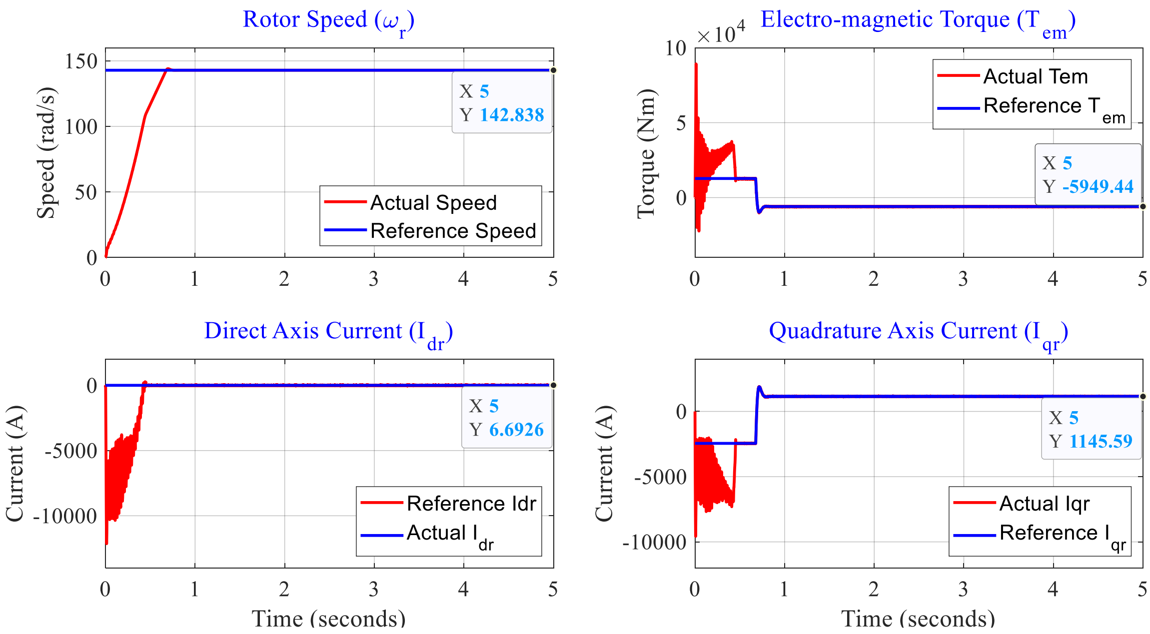
Figure 4. 3: The system response for PI and Fuzzy logic controllers

4.1.4 Simulation Results Using Fuzzy-PI controller

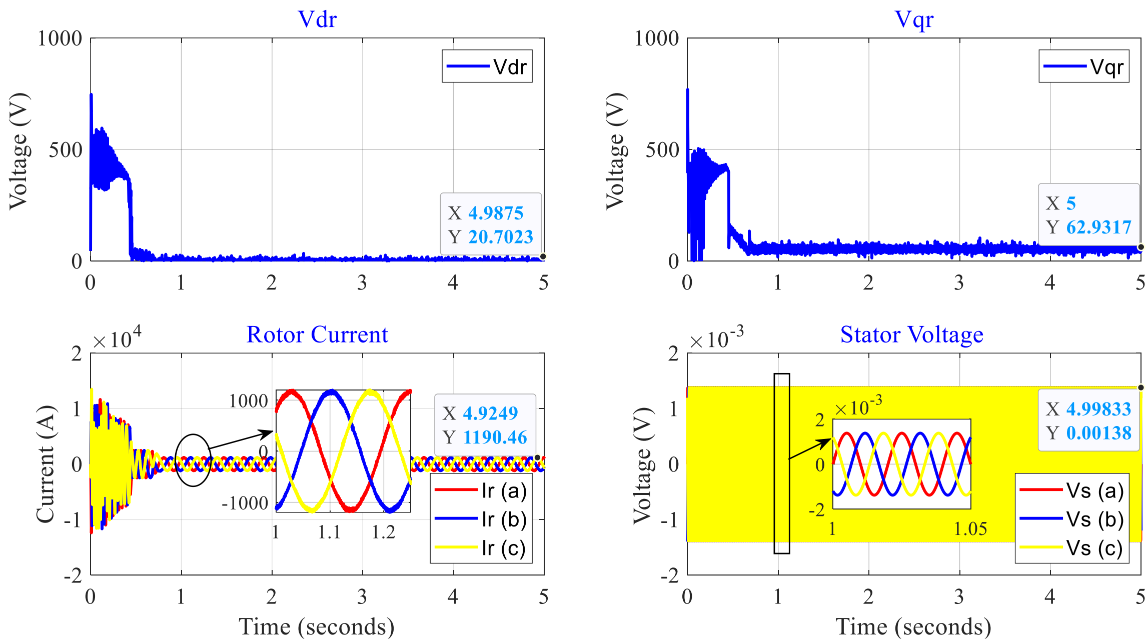
It can be observed from sections 4.1.1 and 4.1.2 that the PI and fuzzy logic controllers show poor speed, and torque control performance from the perspective of steady-state error values. Thus, to enhance the system performance in their limitation area a fuzzy-PI controller represented in Figure 3.24 was developed.

As understood from the developed system, the system response shown in Figure 4.5 matches the steady state values of Figure 3.35 at the reference speed of 1364 rad/s and -6050 Nm torque. Table 4.3 illustrates the system’s time domain performance subject to the proposed fuzzy-PI controller. From the obtained result, it can be seen that the controller could achieve good rise time, settling time, overshoot, and steady-state error values for the speed and torque.

However, the system still needs an improvement in the quadrature and direct axis current regulation, for which the steady-state values are a little bit high, which will be better if kept as minimum as possible. At the time of connection to the load, the system exhibits large oscillation and undershoot as observed in Figure 4.4, which is the limitation of this controller. Nevertheless, this case is common in the reviewed literature and they recommended the utilization of soft starters.



(a)



(b)

Figure 4. 4: Response of the simulated system at rotor speed of 142.8 rad/sec based on fuzzy-PI controllers

Table 4. 3: Time domain Performance analysis, fuzzy-PI controller

Time domain Performance indices	Iqr (A)	Idr (A)	Rotor Speed (rad/sec)	Torque (Nm)
Rise Time	0.0072	0	0.5113	4.5062e-04
Settling Time	0.7529	0.4424	0.6584	0.7430
Overshoot	64.9291	-	0.8308	273.6830
Undershoot	833.3935	-	0	1.4931e+03
Steady State Error	-3.7141	6.6926	1.8876e-04	2.1355

4.1.5 Simulation Result Based on PI, Fuzzy logic, and Fuzzy-PI controllers

Figure 4.5 represents the system response based on PI, Fuzzy logic, and Fuzzy-PI controllers.

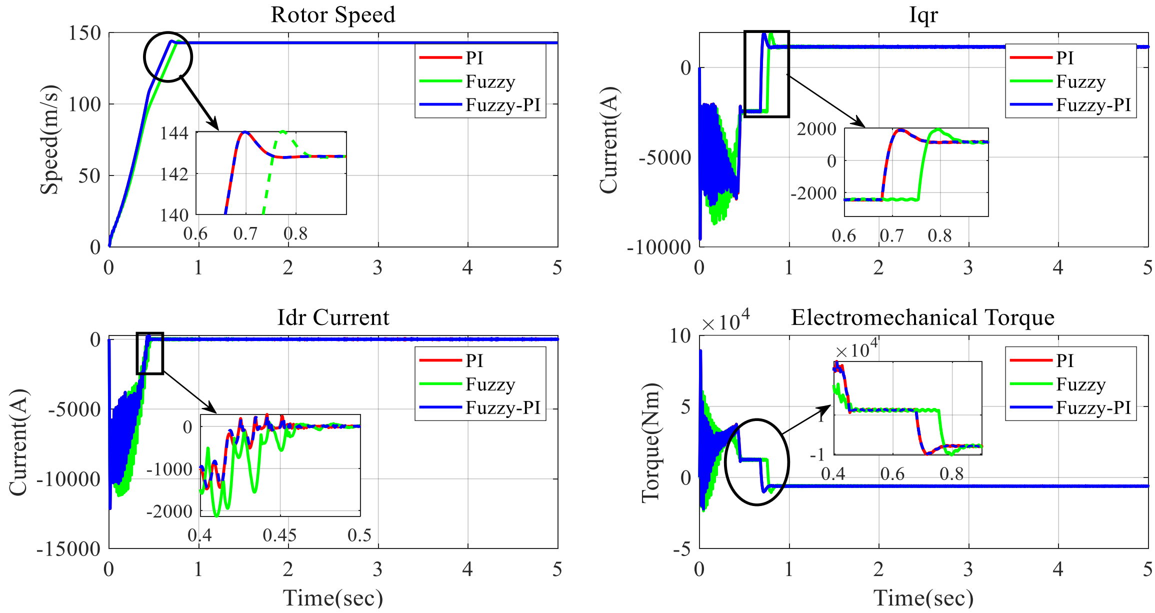
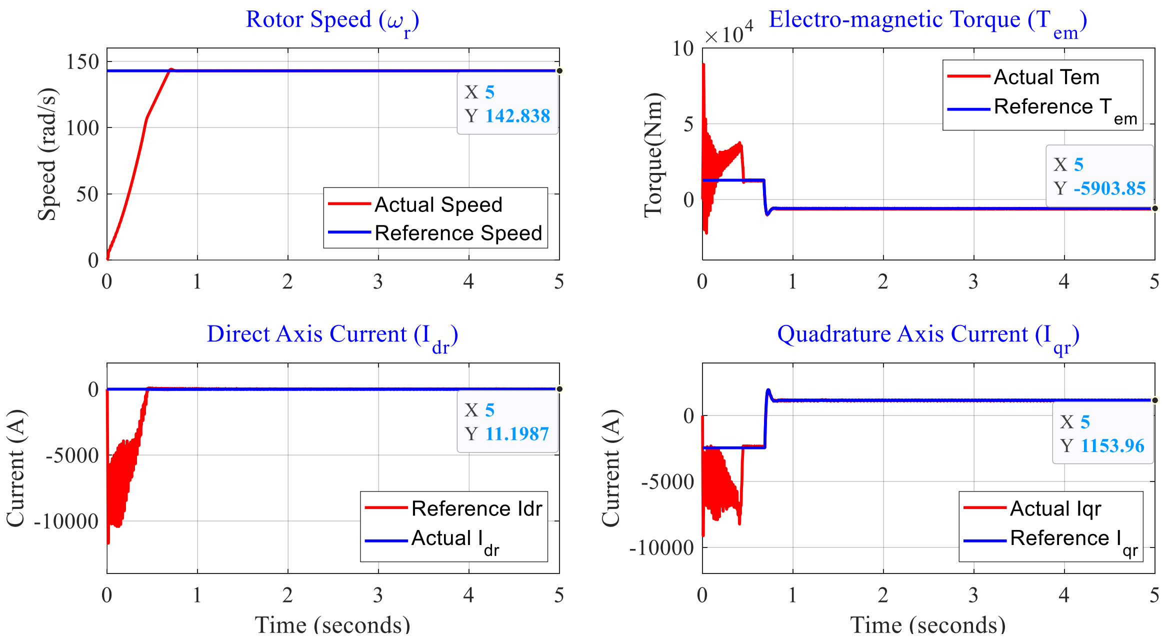


Figure 4. 5: The system response for PI, Fuzzy logic, and Fuzzy-PI controllers

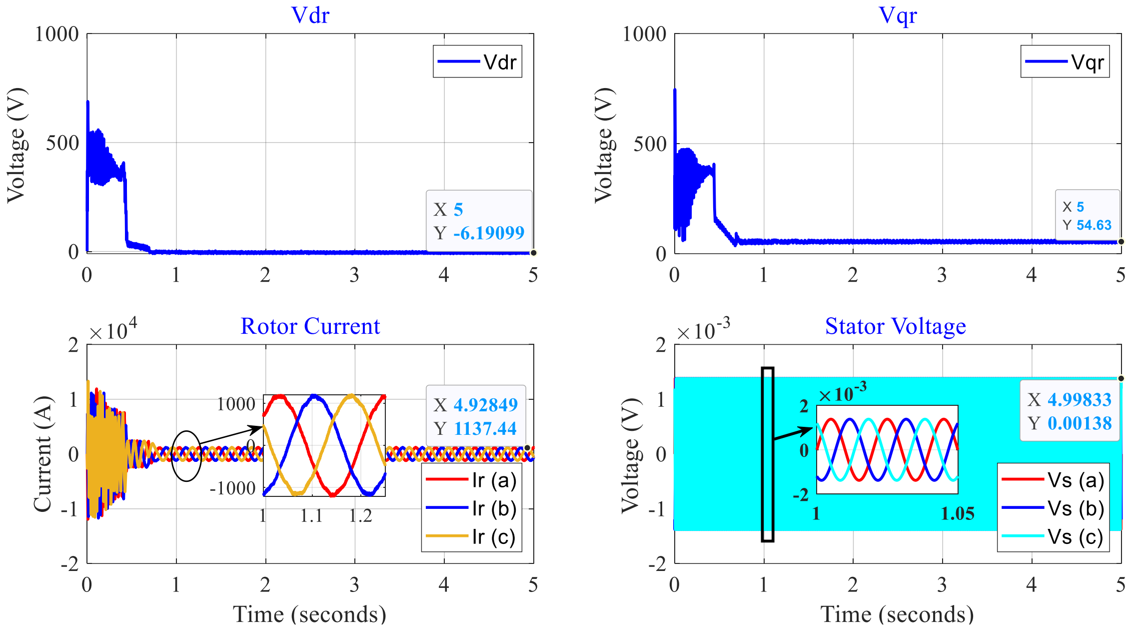
4.1.6 Simulation Results Using ANFIS controller

In the previous sections, the performance analysis of the three controllers; PI, Fuzzy, and Fuzzy-PI controllers have been covered. The designed controllers fulfill most of the performance requirements, but their design takes a long time thus the controller that takes the minimum time of design, and providing flexible and robust control should be considered. Therefore adaptive neuro-fuzzy inference system (ANFIS) has been designed. In this section, the performance of the designed controller is discussed as it has been done for the aforementioned controllers based on Figure 4.6, which represents the system performance subject to the ANFIS controller. The response of the developed system has been illustrated in Figure 4.6, and it has been figured out that the system response matches the steady-state values illustrated in Figure 3.35.

Table 4.4 illustrates the system's time domain performance when the ANFIS controller is used. The controller could provide good rise time, settling time, overshoot, and steady-state error values even if they do not perform as they are expected, as represented in the table. However, the system has a smooth response, even though they need an improvement in the steady state error reduction. At the time of connection to the load, the system exhibits large oscillation and undershoot like the previous controllers, but with a little improvement as depicted in Figure 4.6. The designed controller can perform better than what is represented in this thesis, but it needs more accurate data. Since this controller generates a fuzzy inference engine by itself the only thing required is accurate data. The data is taken from the PI controller control block for this thesis, which forced the ANFIS controller to inherit the limitation of the PI controller. To eliminate or reduce its limitations the trained data needs to be accurate, which only be obtained from the manufacturing industry.



(a)



(b)

Figure 4. 6: Response of the simulated system at rotor speed of 142.8 rad/sec based on ANFIS controller

Table 4. 4: Time domain Performance analysis, ANFIS controller

Time domain Performance indices	Iqr (A)	Idr (A)	Rotor Speed (rad/sec)	Torque (Nm)
Rise Time	0.0069	0	0.5175	4.4986e-04
Settling Time	0.7608	0.4390	0.6674	0.7439
Overshoot	68.3458	-	0.8613	272.8187
Undershoot	792.6241	-	0	1.4932e+03
Steady State Error	2.7742	11.1987	2.9967e-04	71.1057

4.1.7 Simulation Result based on the Four Controllers

Figure 4.9 demonstrates the response of the system based on PI, fuzzy-PI, fuzzy, and ANFIS controllers.

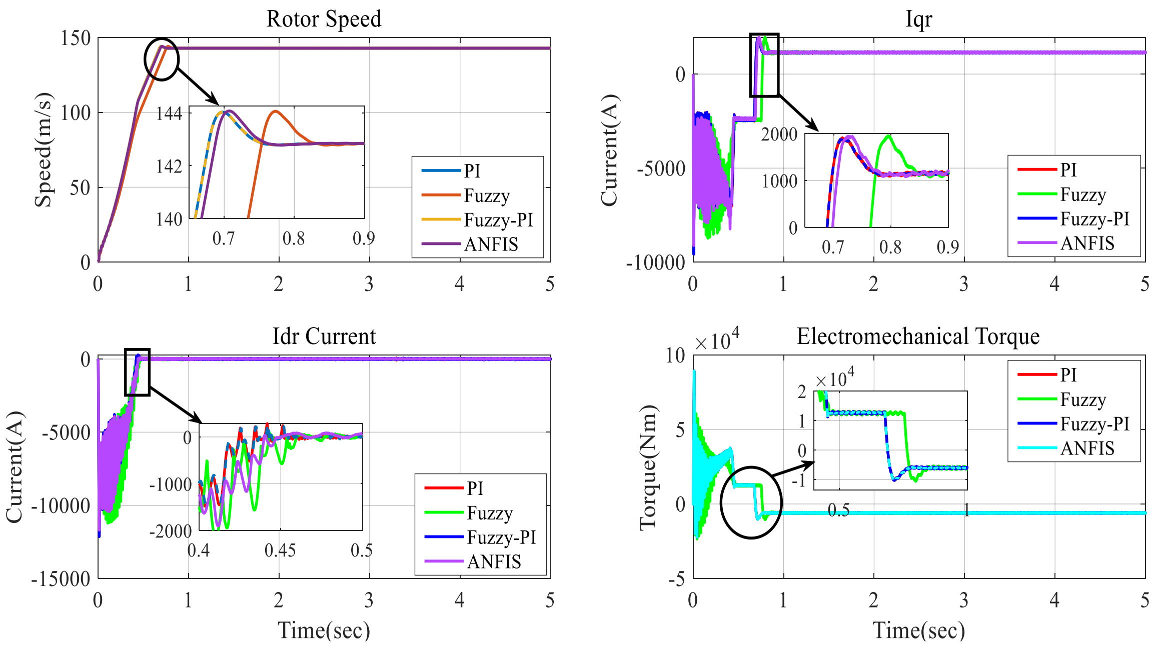


Figure 4. 7: The system response for the four controllers

4.2 Discussions

This section presents the discussion of the simulation results illustrated in the previous sections.

4.2.1 Performance Comparison between PI and Fuzzy Based System

By comparing the corresponding parameters in Table 4.1 and Table 4.2, the DFIG wind turbine response with fuzzy controller shows enhancements in terms of the peak undershoot, rise time, and steady-state error of quadrature axis rotor current, direct axis rotor currents and electromechanical torque compared to the system response subject to PI controller. However, the steady-state error of the rotor speed is minimum for the PI controller, and the system overshoot value for the fuzzy logic controller is larger compared to that of the PI controller. Since this thesis aims to control the rotor current to protect the generator rotor current fuzzy logic controller is found to be a better compared to PI controller to meet the requirements, from the perspective of rise time, settling time, and undershoot.

From Figure 4.3 it is observed that the system response for the fuzzy logic controller has fewer oscillations, smaller undershoot, and smaller peak values compared to the system response for the PI controller. However, the PI controller has a fast settling response compared to the fuzzy logic controller. Further, as observed from the performance comparison bar graph i.e. Figure 4.8 it can be generalized that the fuzzy logic controller has low rise time, undershoot, and steady-state error compared to Proportional integral controller.

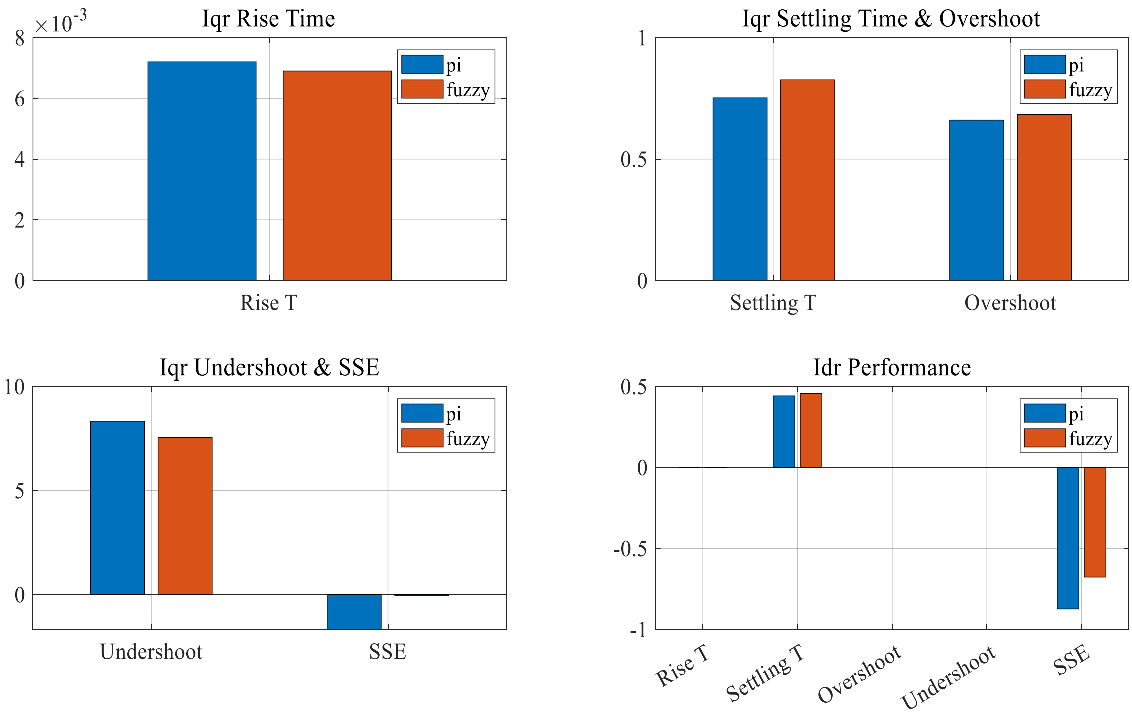


Figure 4. 8: Performance comparison bar graph for PI and Fuzzy logic controllers

4.2.2 Comparison between PI, Fuzzy Logic, and Fuzzy-PI Controllers-based System Performance

Figure 4.9 illustrates the bar graph that is developed from Tables 4.1, 4.2, and 4.3 to compare the performance of the three controllers. In the graph, the measurement parameters of DFIG are compared from the perspective of rise time, settling time, overshoot, undershoot, and steady-state error. It has been figured out from Figure 4.5 and 4.9 that the fuzzy-PI controller exhibit superior performance in terms of settling time, overshoot, and steady-state error for quadrature current. This controller shows the medium undershoot and rise time. A fuzzy controller has the minimum rise time and undershoots value, which makes it a selective controller where rise time and undershoot values are the primary criteria for control action. However, when the control action is required to operate at the minimum settling time, overshoot and steady-state error fuzzy-PI controller can perform the best.

For the direct axis current control, the system performance has been evaluated in terms of rise time, settling time, and steady-state error. Out of the perspective of the performance requirements set as criteria, the fuzzy-PI controller has minimum settling time and the fuzzy controller has the maximum. Where steady-state error is concerned Fuzzy controller is the best but the PI controller is the weakest. The fuzzy-PI controller has a positive steady-state error, which shows that the system steady-state response is lower than the reference value.

It is good for the current to be kept lower than the reference value rather than exciding it because higher currents cause system burnout.

By comparing the performance values of the speed and torque in Tables 4.1, 4.2, and 4.3, it is observed that the fuzzy-PI controller performs best from the perspective of steady-state error and settling time. Fuzzy-PI controller performs better than PI control for torque regulation in terms of steady-state error reduction. Note that the torque and speed regulation is indirect for this thesis. The systems controlled by the conventional PI controller exhibit performance degradation in the performance when compared to the other controllers.

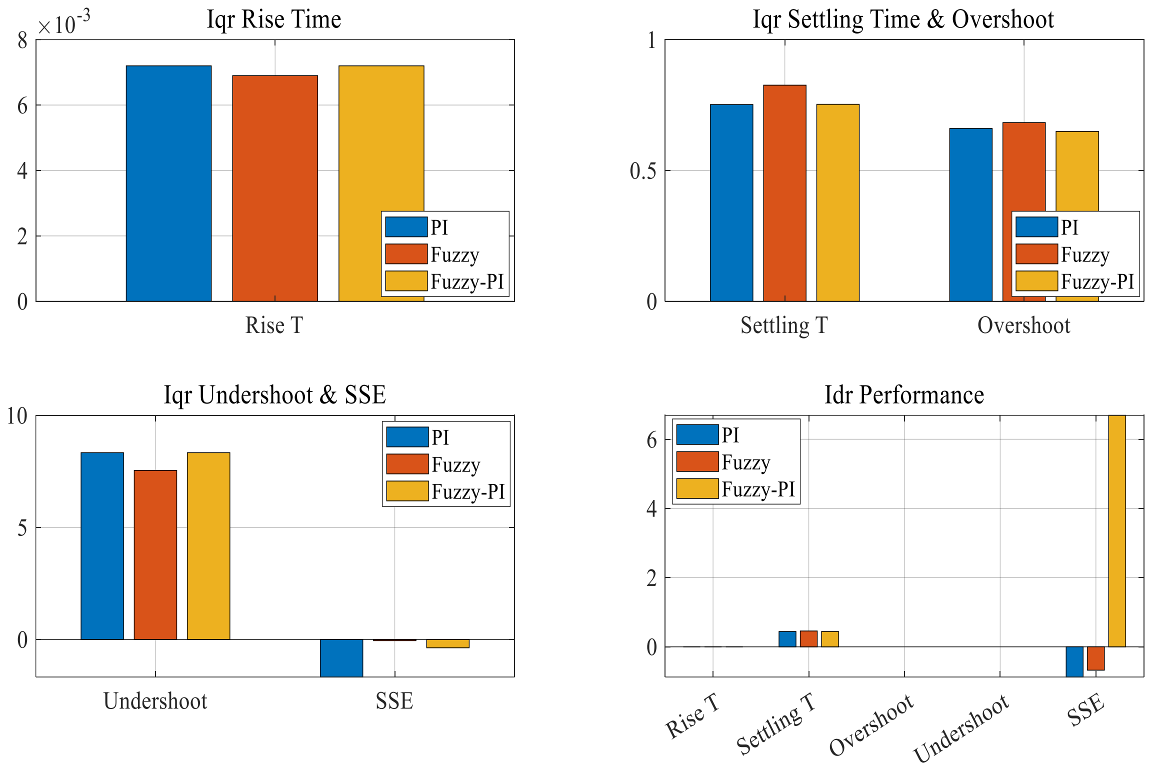


Figure 4. 9: Performance comparison bar graph for PI, Fuzzy logic, and Fuzzy-PI controllers

4.2.3 System Performance Comparison based on the four Controllers

Figure 4.10 illustrates the bar graph that is developed based on Table 4.5 to compare the performance of the four controllers. In the graph, the measurement parameters of DFIG are compared similarly to the previous sections. It is observed in Figure 4.7, which demonstrates the performance of the four controllers, and in Figure 4.10, when the quadrature axis current is considered that PI, fuzzy, and ANFIS controller based system exhibit the same rise time but the fuzzy-PI controller has a slower rise time. The fuzzy-PI controller based system has superior performance in terms of settling time and overshoot. This controller shows poor regulation action when the case is the rise time and undershoot. A system subject to fuzzy logic controller has smallest steady-state error and largest settling time, thus fuzzy logic controller could not be selected if a minimum steady-state error is required. However, the ANFIS controller has a similar performance to the PI controller when quadrature axis current control gains superior requirements. This controller shows the same performance as the PI controller because the training data used to design it is based on the Fuzzy-PI controller performance. Nevertheless, it could perform the best if it is trained for the best experimental training data.

For the direct axis current control, the system performance has been analyzed from the perspective of rise time, settling time, and steady-state error. From those perspectives, the ANFIS controller exhibits the smallest settling time and largest steady-state error. However, the fuzzy logic controller has the smallest steady-state error and largest settling time. The fuzzy logic controller shows the best performance for both current components when the steady state error control takes superior control requirements. However, the ANFIS controller has the best settling time for both current components.

When the torque performance values of the designed controllers in Table 4.5 are compared Fuzzy-PI controller shows the best performance from the perspective of steady-state error and settling time. The conventional PI controller has poor performance compared to the other controllers.

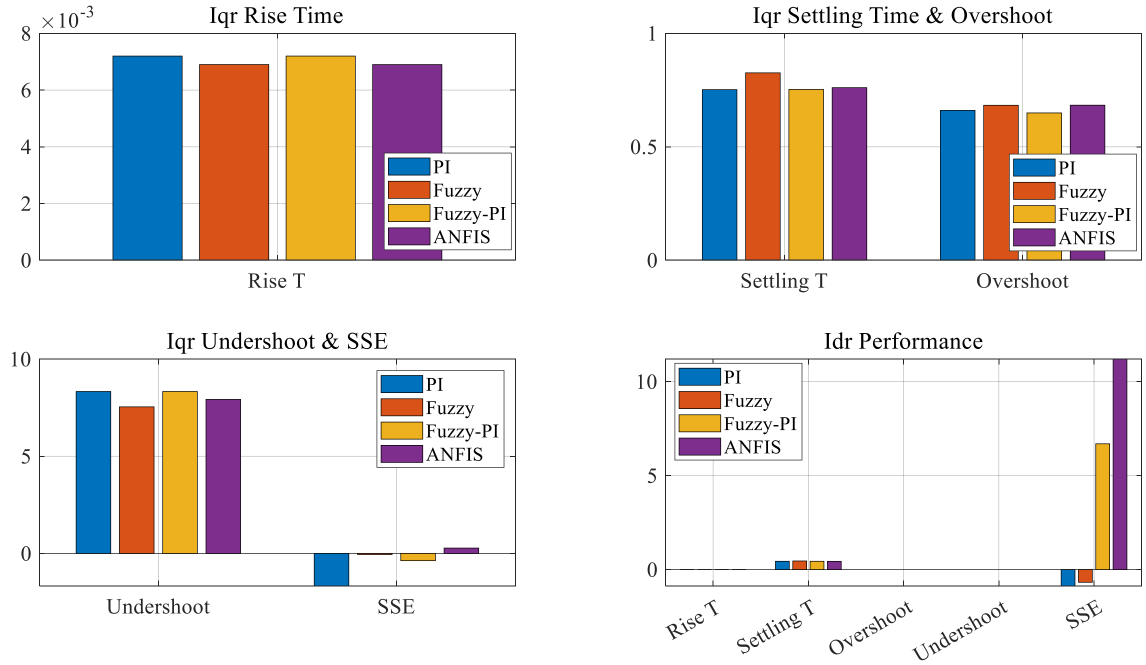


Figure 4. 10: Performance comparison bar graph for the four controllers

Table 4. 5: Time domain Performance analysis of the four controllers

Quadrature axis current (Iqr) response				
Time domain Performance indices	PI	Fuzzy	Fuzzy-pi	ANFIS
Rise Time (s)	0.0072	0.0069	0.0072	0.0069
Settling Time (s)	0.7520	0.8260	0.7529	0.7608
Overshoot (%)	66.0579	68.3055	64.9291	68.3458
Undershoot (%)	833.4304	754.3721	833.3935	792.6241
Steady state error (A)	-16.7453	-0.5421	-3.7141	2.7742
Direct axis current (Idr) response				
Rise Time (s)	0	0	0	0
Settling Time (s)	0.4419	0.4576	0.4424	0.4390
Overshoot (%)	-	-	-	-
Undershoot (%)	-	-	-	-
Steady state error (A)	-0.8734 A	-0.6764	6.6926	11.1987
Rotor Speed response				
Rise Time (s)	0.5111	0.5953	0.5113	0.5175
Settling Time (s)	0.6583	0.7339	0.6584	0.6674
Overshoot (%)	0.8310	0.8532	0.8308	0.8613
Undershoot (%)	0	0	0	0
Steady state error (A)	1.1099e-04	5.6884e-04	1.8876e-04	2.9967e-04
Torque Response				
Rise Time (s)	4.4986e-04	4.7001e-04	4.5062e-04	4.4986e-04
Settling Time (s)	0.7439	0.8235	0.7430	0.7439
Overshoot (%)	272.8187	291.5128	273.6830	272.8187
Undershoot (%)	1.4932e+03	1.1909e+03	1.4931e+03	1.4932e+03
Steady state error (A)	71.106 Nm	-22.5707	2.1355	71.1057

5. CONCLUSIONS AND RECOMMENDATIONS

5.1. Conclusions

The dynamic features of DFIG mainly depend upon rotor current, stator current, and stator flux. Since these three parameters of DFIG are non-linear; they increase the overall complexity of the system. One of the main parameters of DFIG is the rotor current, which needs to be controlled because the output current of the wind turbine has a greater overshoot, a very large settling time, and a large steady-state error. To tackle these problems, this thesis presented an artificial intelligence-based controller for the rotor current of DFIG in a wind turbine system.

The mathematical modeling of the wind speed, aerodynamics, mechanical component, and electrical component of the wind turbine along with the appropriate control mechanisms are presented. The conventional PI controllers have been designed for the rotor speed control loop and direct and quadrature axis currents of DFIG and simulated in MATLAB/Simulink. The simulated system responses were compared with the steady-state characteristics of 2 MW DFIG to study and analyze the accuracy of the designed control actions. The designed PI controller was found to match the steady-state characteristics of the DFIG system with little error, thus it was used as the pillar of comparison to the newly designed controllers. Intelligent control strategies such as Fuzzy, Fuzzy-PI, and artificial neuro-fuzzy inference system has been designed for the RSC of DFIG to control the rotor current replacing the PI controller.

The PI controller gain parameters, Kp and Ki were tuned to obtain desired performance values, however, the PI design could not satisfy the required system performance of the rotor currents. Thus, the aforementioned intelligent controllers were introduced. The performance indices like rising time, settling time, peak overshoot, peak undershoot, and steady-state error were used as a means of comparison for the designed controllers subject to the rotor speed, electromagnetic torque, quadrature axis, and direct axis current components. The fuzzy logic controller shows the best performance for both current components when the steady state error control takes superior control requirements. When the objective of control is the steady state error (SSE) of the system's torque, the SSE of the system used with; PI is 71.1057 Nm, fuzzy,-22.5707 Nm, fuzzy-PI, 2.1355 Nm, ANFIS, 71.1057 Nm. The fuzzy-PI controller based system is found to be the best candidate for SSE control of rotor torque. However, the ANFIS controller has the best settling time for both current components, and the fuzzy-PI controller shows the best performance from the perspective of steady-state error and settling time for torque control purposes. The conventional PI controller shows poor performance compared to the other controllers for most of the design requirements.

It has been examined in the work that the ANFIS controller has almost similar performance to the PI controller. This controller shows the same performance as the PI controller because the training data used to design it is based on the fuzzy-PI controller performance. However, it could perform the best if it is trained for the best experimental training data. The steady-state values of certain parameters of the DFIG such as; the rotor speed, electromagnetic torque, quadrature axis, and direct axis current components of the wind turbine were evaluated after designing each controller, to assure that the simulation results follow the design characteristics.

Generally, the comparison of PI, fuzzy, fuzzy-PI, and ANFIS controllers has been done, and it was examined that the intelligent controllers show the best performance compared to the conventional, PI controller. However, the three intelligent controllers are difficult to select from because they show a competitive performance from different perspectives even though the ANFIS controller shows weak performance compared to fuzzy and fuzzy-PI because of the training data.

5.2. Recommendations

The designed system can be used by the scientific community for a detailed study of the control strategies to be used in wind turbine based-DFIG such as WT behavior under fault, performance under various test conditions, and for further research on the topic. Soft starters have better if used to overcome the higher transient errors of the system responses. In future work, pitch angle control can be designed to analyze the performance of DFIG subject to variable wind speeds. Further, it may be promising to extend the design to a wind farm that consists of multiple wind turbines.

Research fund Acknowledgment

This research project is funded by Adama Since and Technology University under the grant number ASTU/SM-R/571/22, Adama, Ethiopia.

REFERENCES

A Ashglaf, Mohmed. 2019. “Development of Hybridization Concept for Horizontal Axis Wind / Tidal Systems Using Functional Similarities and Advanced Real-Time Emulation Methods.” *Conference Proceedings*.

Aarib, Abdelali, Aymane El Moudden, Abdelhadi El Moudden, and Abdelhamid Hmidat. 2021. “Control and Investigation of Operational Characteristics of Variable Speed Wind Turbines with Doubly Fed Induction Generators.” *Walailak Journal of Science and Technology* 18(4).

Abad, Gonzalo et al. 2011. Doubly Fed Induction Machine *Doubly Fed Induction Machine*.

Abo-Khalil, Ahmed G. 2013. *Impacts of Wind Farms on Power System Stability*. NTECH Open Access Publisher.

Abo-Khalil, Ahmed G., Ali Alghamdi, I. Tlili, and Ali M. Eltamaly. 2019. “Current Controller Design for DFIG-Based Wind Turbines Using State Feedback Control.” *IET Renewable Power Generation* 13(11).

Abu-Rub, H. 2014. Chichester *Power Electronics for Renewable Energy Systems, Transportation, and Industrial Applications*. IEEE Press and John Wiley.

Ahmad, Hamza et al. 2021. “Improved Current Controlled Doubly Fed Induction Generator Model with Grid Integration Under Sub and Super Synchronous Conditions.” *Journal of Electrical Engineering and Technology* 16(1).

Ahmad, Ishteyaq, and Rakesh Sharma. 2018. “OPEN ACCESS ANFIS Based Doubly Fed Induction Generator for WECS with LLCL-2TRP Filter.” 8(6): 9–15.

Amin, Ifte K, M Nasir Uddin, and M Marsadek. 2019. “ANFIS Based Neuro-Fuzzy Control of DFIG for Wind Power Generation in Standalone Mode.” : 2077–82.

Angeline, K Rebecca, Tripura Pidikiti, Srinivasa Kishore, and Babu Yadlapati. 2017. “Modelling and Simulation of ANFIS Controlled Doubly FED Induction Generator Based Wind Energy System for Performance Enhancement.” 10(5): 61–73.

Ayrir, Wiam, Ourahou Meriem, and Ali Haddi. 2017. “Fuzzy Logic Rotor Currents Control of a DFIG-Based Wind Turbine.” In *Proceedings of Engineering and Technology - PET*, , 3–8.

Badreldien, Mahmoud, Rana Usama, Amged El-wakeel, and Almoataz Y Abdelaziz. 2014.

“Modeling , Analysis and Control of Doubly Fed Induction Generators for Wind Turbines.” (January 2015).

Belkacem, Yasser, and Mohammed Ouriagli. 2019. “Fuzzy Controllers for a High Performance of the Doubly Fed Induction Generator Based on Wind Turbine in Variable Speed.” *2019 International Conference on Intelligent Systems and Advanced Computing Sciences (ISACS)* (1): 1–7.

Cao, Wenping, Ying Xie, and Zheng T. 2014. “Wind Turbine Generator Technologies.” In *Advances in Wind Power*,.

Chijioke, John Nwankwo, and Vedat Kiray. 2020. “A Review of Power Electronics Applications for Wind Energy Systems in Microgrids.” 26(4): 139–48.

“Control Tutorials for MATLAB and Simulink - Introduction: PID Controller Design.” https://ctms.engin.umich.edu/CTMS/index.php?example=Introduction&section=ControlPID (June 7, 2022).

Deshpande, Anjali, and V A Kulkarni. 2020. “Dfig For Variable Speed Wind Energy Generation And Its Characteristics.” 9(03).

Dida, Abdelhak, Farid Merahi, and Saad Mekhilef. 2020. “New Grid Synchronization and Power Control Scheme of Doubly-Fed Induction Generator Based Wind Turbine System Using Fuzzy Logic Control.” *Computers and Electrical Engineering* 84.

“Different Types and Parts of a Horizontal Axis Wind Turbines - DolceraWiki.” https://www.dolcera.com/wiki/index.php?title=Different_Types_and_Parts_of_a_Horizontal_Axis_Wind_Turbines (May 6, 2022).

Eloghene Okedu, Kenneth. 2019. “Introductory Chapter: Power System Stability.” *Power System Stability*.

Gomroki, Saeid, Sadegh Vaez-Zadeh, and Alireza Jabbarnejad. 2020. “Combined Control of Machine Grid Side Converters for PMSGs in Wind Energy Conversion Systems with Fast and Robust Performance.” In *2020 11th Power Electronics, Drive Systems, and Technologies Conference, PEDSTC 2020*,.

Gribkov, S. V., and S. N. Chizhma. 2021. “Vertical-Axis Wind Turbines. Design Technique.” In *IOP Conference Series: Earth and Environmental Science*,.

Gupta, R A. 2015. “Wind Energy Conversion System Using PMSG.” : 199–203.

Hamane, B. et al. 2012. "Control for Variable Speed Wind Turbine Driving a Doubly Fed Induction Generator Using Fuzzy-PI Control." *Energy Procedia* 18(August 2015): 476–85.

Hamane, B., and M. L. et al. Doumbia. 2014. "Control of Wind Turbine Based on DFIG Using Fuzzy-PI and Sliding Mode Controllers." In *2014 9th International Conference on Ecological Vehicles and Renewable Energies, EVER 2014,*.

Hansen, Anca D., Paul Sørensen, Frede Blaabjerg, and John Becho. 2012. "Dynamic Modelling of Wind Farm Grid Interaction." *Wind Engineering* 26(4).

Hasnae Elouatouat, Tamou Nasser et al. 2020. "Control of a Doubly-Fed Induction Generator for Wind Energy Conversion Systems." *IEEE Xlore* 1(4): 7281–4341.

Helal, Asmaa El-sayed. 2018. "Control and Performance Analysis for A Doubly- Fed Induction Generator Driven by A Wind Turbine Using Different Optimized Controllers." *2018 Twentieth International Middle East Power Systems Conference (MEPCON)*: 221–27.

"How Do Wind Turbines Work? | Department of Energy." https://www.energy.gov/eere/wind/how-do-wind-turbines-work (May 6, 2022).

"How Electricity Is Generated - U.S. Energy Information Administration (EIA)." https://www.eia.gov/energyexplained/electricity/how-electricity-is-generated.php (April 6, 2022).

Hu, J. Nian, H. Xu and Y. He. 2016. "Dynamic Modeling and Improved Control of DFIG Under Distorted Grid Voltage Conditions." *IEEE Transactions on Energy Conversion* 26(no.1): 163–75.

Ibrahim, Raed Khalaf, and Simon Watson. 2015. "Wind Turbine Simulation Model for the Study of Combined Mechanical and Electrical Faults." In *European Wind Energy Association Annual Conference and Exhibition 2015, EWEA 2015 - Scientific Proceedings,*.

Kaloi, Ghulam Sarwar, Jie Wang, and Mazhar Hussain Baloch. 2016a. "Dynamic Modeling and Control of DFIG for Wind Energy Conversion System Using Feedback Linearization." *Journal of Electrical Engineering and Technology* 11(5): 1137–46.

"Dynamic Modeling and Control of DFIG for Wind Energy Conversion System Using Feedback Linearization."2016. *Journal of Electrical Engineering and Technology*

11(5): 1137–46.

Kasbi, Abdellatif, and Abderrafii Rahali. 2020. “A Simple Methodology for Optimal Fuzzy Control of DFIG Based Wind Turbine.” In *2020 1st International Conference on Innovative Research in Applied Science, Engineering and Technology, IRASET 2020*,.

Khan, Muhammad Adnan, Ahsan Ali, and Inam-ul-hassan Shaikh. 2022. “Hybrid Fuzzy-PI and ANFIS Controller Design for Rotor Current Control of DFIG Based Wind Turbine.” : 35–41.

Lamnadi, Mouna, Mourad Trihi, Badre Bossoufi, and Abdelkader Boulezhar. 2016. “Modeling and Control of a Doubly-Fed Induction Generator for Wind Turbine-Modeling and Control of a Doubly-Fed Induction Generator for Wind Turbine-Generator Systems.” (April 2017): 982–94.

Liu, Jinkun. 2018. *Intelligent Control Design and MATLAB Simulation*. Beijing: Springer Nature.

Marugán, Alberto Pliego, Fausto Pedro García Márquez, Jesus María Pinar Perez, and Diego Ruiz-Hernández. 2018. “A Survey of Artificial Neural Network in Wind Energy Systems.” *Applied Energy* 228: 1822–36.

Medjber, Ahmed, Abderrezak Guessoum, Hocine Belmili, and Adel Mellit. 2016. “New Neural Network and Fuzzy Logic Controllers to Monitor Maximum Power for Wind Energy Conversion System.” *Energy* 106.

Naidu, N K Swami, Student Member, and Bhim Singh. 2015. “Doubly Fed Induction Generator for Wind Energy Conversion Systems with Integrated Active Filter Capabilities.” 3203(c): 1–11.

Naik, Kanasottu Anil, and Chandra Prakash Gupta. 2018. “Output Power Smoothing and Voltage Regulation of a Fixed Speed Wind Generator in the Partial Load Region Using STATCOM and a Pitch Angle Controller.” *Energies* 11(1).

Nayar, C.N, S.M Islam, H Dehbonei, and K Tan. 2011. Power Electronics Handbook: Devices, Circuits, and Applications *Power Electronics Handbook - Devices, Circuits, and Applications (3rd Edition).*

Nise, Norman S. 2015. *CONTROL SYSTEMS ENGINEERING*. Seventh. California State Polytechnic University ,Pomona: Wiley.

Noussi, K., A. Abouloifa, H. Katir, and I. Lachkar. 2019. "Modeling and Control of a Wind Turbine Based on a Doubly Fed Induction Generator." *Proceedings of 2019 IEEE World Conference on Complex Systems, WCCS 2019*: 1–5.

Okedu, Kenneth. 2020. "Augmentation of Wind Farm Fault Ride Through by DFIG-Based Variable Speed Wind Generator." In *Enhanced Power Grid Stability Using Doubly-Fed Induction Generators*,.

Okedu, Kenneth E. 2019. "Enhancing the Performance of DFIG Variable Speed Wind Turbine Using a Parallel Integrated Capacitor and Modified Modulated Braking Resistor." *IET Generation, Transmission and Distribution* 13(15).

Okedu, Kenneth E, and Hind F A Barghash. 2021. "Enhancing the Performance of DFIG Wind Turbines Considering Excitation Parameters of the Insulated Gate Bipolar Transistors and a New PLL Scheme." 8(January): 1–11.

Parthasarathy, S S et al. 2017. "ScienceDirect ScienceDirect A Novel Fuzzy Logic Controller for Vector Heating Controlled and Cooling Induction Motor Drive Assessing the Feasibility of Using the Heat Demand-Outdoor Temperature Function for a Long-Term District Heat Demand Forecast." *Energy Procedia* 138: 686–91.

Qin, Boyu, and Haoyuan Sun. 2018. "State Dependent Riccati Equation Based Rotor-Side Converter Control for Doubly Fed Wind Generator." In *IEEE Access*,.

Rajendran, Saravanakumar, and Debashisha Jena. 2014. "Control of Variable Speed Variable Pitch Wind Turbine at Above and Below Rated Wind Speed." *Journal of Wind Energy* 2014.

Soliman, Mohamed Hassan, Bassel I. Nesser, and Krzysztof Pytel. 2019. "Local Benefits Factored in Utility Scale Wind Energy for a Remote Location." *New Trends in Production Engineering* 2(1).

Syahputra, Ramadoni, Indah Soesanti, Jl Grafika, and Kampus Ugm. 2017. "Modeling of Wind Power Plant with Doubly-Fed Induction Generator." 1(3): 126–34.

Tang, Yufei et al. 2014. "Reactive Power Control of Grid-Connected Wind Farm Based on Adaptive Dynamic Programming." *Neurocomputing* 125.

"The Parts of a Wind Turbine: Major Components Explained - Energy Follower." https://energyfollower.com/parts-of-a-wind-turbine/ (May 6, 2022).

Wang, Cong, Zhengchun Du, Yu Ni, and Chongtao Li. 2016. “Simplified Model of Doubly Fed Induction Generator in Normal Operation.” In *2016 IEEE International Conference on Power System Technology, POWERCON 2016*,.

“Wind Energy | Alternative Energy Tutorials.” https://www.alternative-energy-tutorials.com/wind-energy/wind-energy (May 6, 2022).

“WINDExchange: Utility-Scale Wind Energy.” https://windexchange.energy.gov/markets/utility-scale (April 6, 2022).

Zhang, Yongchang, and Jian Jiao. 2018. “Model Predictive Rotor Current Control for Doubly Fed Induction Generators under Unbalanced Grid Voltages.” In *Proceedings - 2018 IEEE International Power Electronics and Application Conference and Exposition, PEAC 2018*,.

Zou, Yu, Malik E. Elbuluk, and Yilmaz Sozer. 2013. “Simulation Comparisons and Implementation of Induction Generator Wind Power Systems.” *IEEE Transactions on Industry Applications* 49(3): 1119–28.

APPENDICES

A. MATLAB/Simulink model

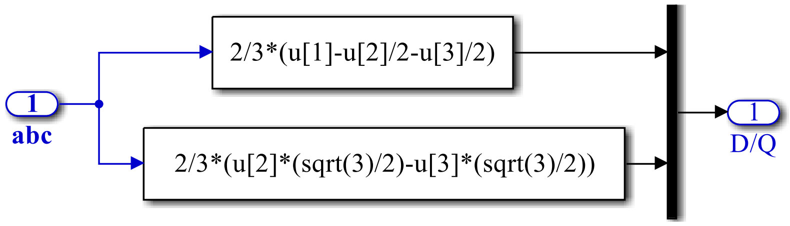


Figure A 1: Inverse Clark transformation

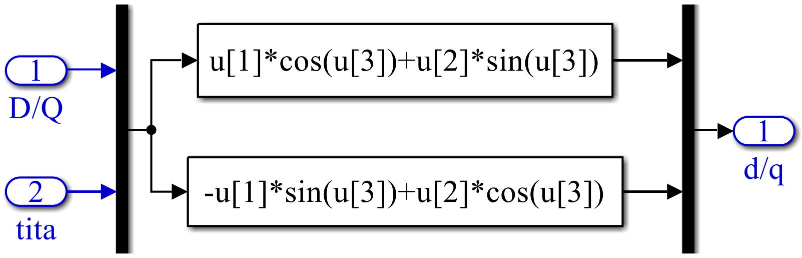


Figure A 2: Inverse Park transformation

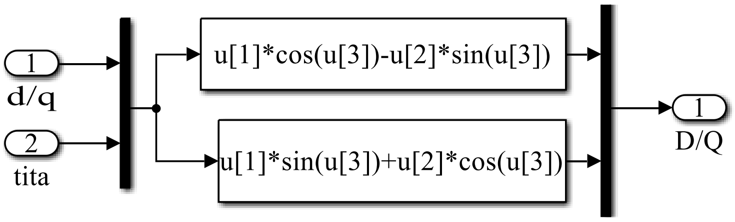


Figure A 3: Park transformation

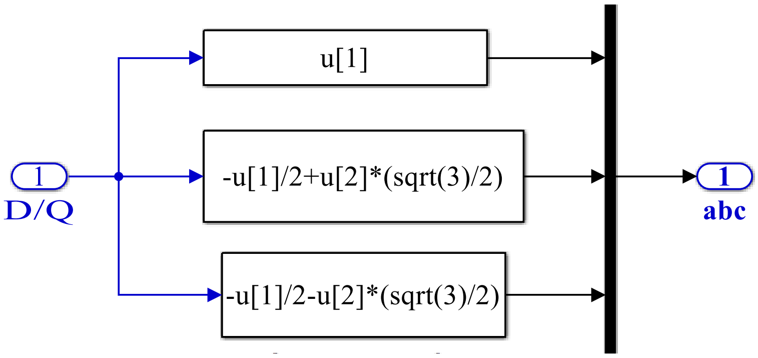


Figure A 4: Clark transformation

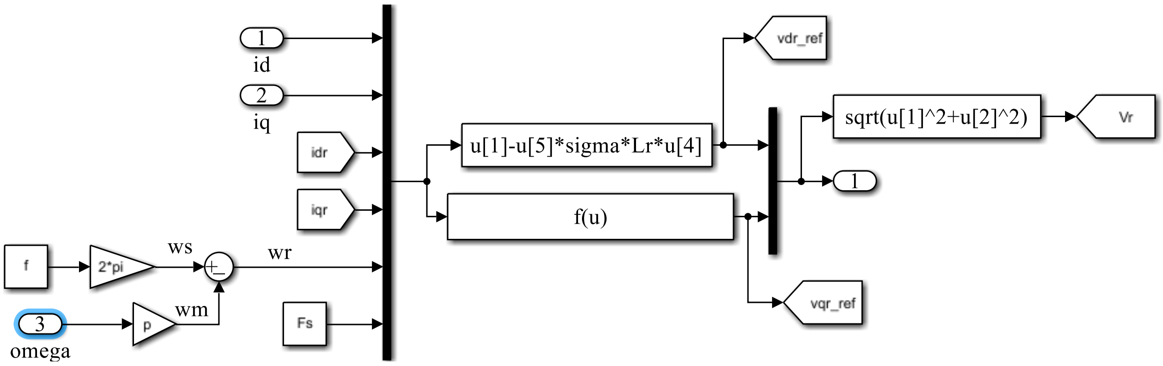


Figure A 5: Cancellation of Cross terms

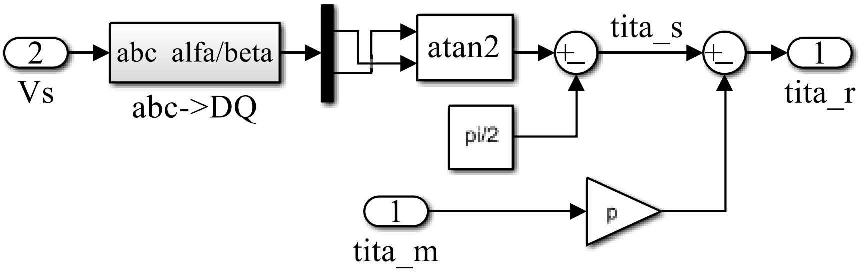


Figure A 6: Angle calculation using atan2 function

B. MATLAB Script Program

1. PI Controller

```
clc

close all

clear all

%DFIG parameters (Rotor parameters referred to the stator side)

f=50;        %Satator frequency(Hz)

Ps=2e6;      %Rated stator power(W)

n=1500;      %Rated rotational speed(rev/min)

Is=1760;     %Rated stator current

Vs=690;      %Rated stator voltage

Tem=12732;   %Rated torque

p=2;         %Pole pair

u=1/3;       %stator/rotor turns ratio

Vr=2070;     %rated rotor votage(non-reached)(V)
```

```
smax=1/3;    %maximum slip
Vr_stator=(Vr*smax)*u;%rated rotor voltage(non-reached((V)
Rs=2.6e-3;           %Stator resistance(ohm)
Lsi=0.087e-3;        %Leakage inductance(Stator and rotor)(H)
Lm=2.5e-3;           %Leakage mutual inductance(H)
Rr=2.9e-3;           %Rotor resistance refered to the rotor
Ls=Lm+Lsi;           %stator inductance(H)
Lr=Lm+Lsi;           %rotor inductance(H)
Vbus=Vr_stator*sqrt(2);%Dc bust refered to the stator
sigma =1-Lm^2/(Ls*Lr); Fs=Vs*sqrt(2/3)/(2*pi*f);%stator Flux(aprox.)(Wb)
J=127;                  %inertia
D=1e-3;                 %Damping constant
fsw=4e3;                %switching frequency
Ts=1/fsw/50; N=100;
% PI Control Parameters
tau_i=(sigma*Lr)/Rr; tau_n=0.05/4; wni=100*(1/tau_i); wnn=1/tau_n;
kp_id=(2*wni*sigma*Lr)-Rr;     % Proportional gain
kp_iq=kp_id; ki_id=(wni^2)*Lr*sigma;       %Integral gain
ki_iq=ki_id; kp_n=(2*wnn*J)/p; ki_n=((wnn^2)*J)/p;
n_r=[kp_id/(sigma*Lr),ki_id/(sigma*Lr)]; d_r=[1,2*wni,wni^2];
s=tf('s') yd=tf(n_r,d_r)
step(yd)
title('The Step response of the current loop with PI regulator')
ylabel('direct axis curent (id)')
```

2. Steady State Analysis

```
close all; clear all; clc
```

```
% DFIG parameters -> Rotor parameters refered to the stator side

f=50; Ps=2e6; n=1500; Is=1760; Vs=690; Tem=12732; p=2; u=1/3; Vr=2070; smax=1/3;
Vr_stator=(Vr*smax)*u; Rs=2.6e-3; Lsi=0.087e-3; Lm=2.5e-3; Rr=2.9e-3; Ls=Lm+Lsi;
Lr=Lm+Lsi; Vbus=Vr_stator*sqrt(2); sigma =1-Lm^2/(Ls*Lr); Vs=690*sqrt(2/3);
ws=f*2*pi;

%Gearbox Ratio

N=100;

% Minimum and Maximum speed of the 2MW

wt_min=9; wt_nom=18;

% Speed and Torque Arrays(Inputs)

wt=[0.9425,0.9425,0.9425,0.9524,1.0476,1.1429,1.2381,1.3333,1.4286,1.5238,1.6190,1.7
143,1.8095,1.8850,1.8850,1.8850,1.8850,1.8850,1.8850,1.8850,1.8850];

Torque= -
1e6*[0.0501,0.1164,0.1925,0.2696,0.3262,0.3883,0.4557,0.5285,0.6067,0.6902,0.7792,0.8
736,0.9733,1.0894,1.2474,1.2745,1.2745,1.2745,1.2745,1.2745,1.2745];

Temm=Torque/N; %convert to machine side

wmm=wt*(60/2/pi)*N;%convert to machine side(in rpm)

sss=(1500-wmm)/1500;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%Reference current control strategy at (idr=0)

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

i=0;

for ss=sss,

i=i+1; Tem=Temm(i); Wm=ws*(1-ss); s(i)=ss;  A=ws*ws*Lm*Lm +(Rs*Lm/Ls)^2;
B=4*Rs*Tem*ws/3/p-((Vs)^2); C=4*Rs*Rs*Tem*Tem/(9*p*p*Lm*Lm);

X1=(-B+sqrt(B.^2-4*A.*C))./(2*A); X2=(-B-sqrt(B.^2-4*A.*C))./(2*A);

ims=sqrt(X1); ims_(i)=ims; Fs=ims*Lm;

%Rotor currents
```

```
idr=0; iqr=-2*Tem*Ls./(3*p*Lm*Fs) ; Ir(i)=sqrt(idr^2+iqr^2);
%Stator Currents
ids=Fs/Ls; iqs=-iqr*Lm/Ls; Is(i)=sqrt(ids^2+iqs^2);
%Stator Voltages
uds=Rs*ids; uqs=Rs*iqs+ws*Fs; Us(i)=sqrt(uds^2+uqs^2);
%Stator Power
Ps(i)=1.5*(uds*ids+uqs*iqs);
Qs(i)=1.5*(uqs.*ids-uds.*iqs); Mod_Ss=sqrt(Ps.^2+Qs.^2);
%rotor speed
wr=ws*ss;
%Rotor voltage
udr=Rr*idr-Lr*wr.*iqr-Lm*wr.*iqs; uqr=Rr*iqr+Lr*wr.*idr+Lm*wr.*ids;
Vr(i)=sqrt(udr^2+uqr^2);
%Rotor power
Pr(i)=1.5*(udr.*idr+uqr.*iqr); Qr(i)=1.5*(uqr.*idr-udr.*iqr);
Mod_Sr=sqrt(Pr.^2+Qr.^2);
%Stator flux
Fsd=Ls*ids+Lm*idr; Fsq=Ls*iqs+Lm*iqr; Fs(i)=sqrt(Fsd*Fsd+Fsq*Fsq);
%Rotor flux
Frd=Lr*idr+Lm*ids; Frq=Lr*iqr+Lm*iqs; Fr(i)=sqrt(Frd*Frd+Frq*Frq);
%Mechanical power
Pmec(i)=Tem*Wm/p;
%Effeciency
if Pmec(i)>=0
R(i)=Pmec(i)/(Ps(i)+Pr(i)); %Motor Mode
else
R(i)=(Ps(i)+Pr(i))/Pmec(i); %Generator Mode
```

```
end
end
figure(1)
subplot(3,3,1)
hold on
plot(wmm,Temm,'g',wmm,Temm,'.r','linewidth',1.5); xlabel('n (rpm)','fontsize',12);
ylabel('Tem (N.m)','fontsize',12); grid
xlim([800 1900]); subplot(3,3,2)
hold on
plot(wmm,Ps+Pr,'g',wmm,Ps+Pr,'.r','linewidth',1.5); xlabel('n (rpm)','fontsize',12);
ylabel('Pt (W)','fontsize',12); grid
xlim([800 1900]);  subplot(3,3,3)
hold on
plot(wmm,Pr,'r',wmm,Pr,'.k','linewidth',1.5); xlabel('n (rpm)','fontsize',12);  ylabel('Pr
(W)','fontsize',12); xlim([800 1900]);
subplot(3,3,3)
hold on
plot(wmm,Ps,'g',wmm,Ps,'.r','linewidth',1.5); xlabel('n (rpm)','fontsize',12);  ylabel('Ps& Pr
(W)','fontsize',12);
grid
xlim([800 1900]); subplot(3,3,4)
hold on
plot(wmm,Is,'g',wmm,Is,'.r','linewidth',1.5); xlabel('n (rpm)','fontsize',12);  ylabel('Is
(A)','fontsize',12); grid
xlim([800 1900]);  subplot(3,3,5)
hold on
```

```
plot(wmm,Ir,'g',wmm,Ir,'.r','linewidth',1.5); xlabel('n (rpm)','fontsize',12); ylabel('Ir (A)','fontsize',12); grid

xlim([800 1900]);  subplot(3,3,6)

hold on

plot(wmm,Vr,'g',wmm,Vr,'.r',wmm,Us,'r',wmm,Us,'.g','linewidth',1.5); xlabel('n (rpm)','fontsize',12); ylabel('Vr & Vs (V)','fontsize',12); grid

xlim([800 1900]); subplot(3,3,7)

hold on

plot(wmm,Qs,'g',wmm,Qs,'.r','linewidth',1.5); xlabel('n (rpm)','fontsize',12); ylabel('Qs (VAR)','fontsize',12); grid

xlim([800 1900]);  subplot(3,3,8)

hold on

plot(wmm,Qr,'g',wmm,Qr,'.r','linewidth',1.5); xlabel('n (rpm)','fontsize',12); ylabel('Qr (VAR)','fontsize',12); grid

xlim([800 1900]); subplot(3,3,9)

hold on

plot(wmm,R,'g',wmm,R,'.r','linewidth',1.5); xlabel('n (rpm)','fontsize',12);
ylabel('Effeciency (pu)','fontsize',12); grid
xlim([800 1900]);
```

3. Performance of the designed system for different controllers

```
%% System performance Subject to PI controller
load 'omega_pi.mat'     %Rotor Speed
xo_pi=ans.time; yo_pi=ans.data; yo_1=yo_pi(:,1); yo_2=yo_pi(:,2);
ome=stepinfo(yo_2,xo_pi,yo_1(end)) ome_SSE=yo_2(end)-yo_1(end)
figure(1)
subplot(2,2,1)                  % Rotor Speed
plot(xo_pi,yo_2,'r',xo_pi,yo_1,'b','linewidth',2)
%xlabel('Time(seconds)', 'FontSize',16 ,'FontName','TimesNewRoman')
ylabel ('Speed (rad/s)', 'FontSize',16 ,'FontName','TimesNewRoman')
axis([0 5 0 160])
```

```
title('Rotor Speed (\omega_r)','Color','blue',
'FontSize',16 ,'FontName','TimesNewRoman','FontWeight','normal')
legend ({'Actual Speed','Reference
Speed'},'Location','southeast','FontSize',14 ,'FontName','TimesNewRoman')
grid on
load 'Tem_pi.mat'      %Tem
xt_pi=ans.time; yt_pi=ans.data; yt_1=yt_pi(:,1); yt_2=yt_pi(:,2);
tem=stepinfo(yt_2,xt_pi,yt_1(end)); tem_SSE=yt_2(end)-yt_1(end)
subplot(2,2,2)
plot(xt_pi,yt_2,'r',xt_pi,yt_1,'b','linewidth',2)
%xlabel('Time(seconds)','FontSize',14 ,'FontName','TimesNewRoman')
ylabel ('Torque (Nm)','FontSize',14 ,'FontName','TimesNewRoman')
axis([0 5 -40000 100000])
title('Electro-magnetic Torque (T_e_m)','Color','blue', 'FontSize',14 ,'FontName',
'TimesNewRoman','FontWeight','normal')
legend ({'Actual Tem','Reference T_e_m'},'Location','Northeast','FontSize',14 ,'FontName',
'TimesNewRoman')
grid on
load 'iqr_pi.mat'    %iqr
x_pi=ans.time; y_pi=ans.data; y_1=y_pi(:,1); y_2=y_pi(:,2);
iqr=stepinfo(y_2,x_pi,y_1(end)); iqr_SSE=y_2(end)-y_1(end)
subplot(2,2,4)
plot(x_pi,y_2,'r',x_pi,y_1,'b','linewidth',2)
xlabel('Time (seconds)','FontSize',14 ,'FontName','TimesNewRoman')
ylabel ('Current (A)','FontSize',14 ,'FontName','TimesNewRoman')
axis([0 5 -12000 4000])
title('Quadrature Axis Current (I_q_r)','Color','blue', 'FontSize',14 ,'FontName',
'TimesNewRoman','FontWeight','normal')
legend ({'Actual Iqr','Reference I_q_r'},'Location','southeast','FontSize',14 ,'FontName',
'TimesNewRoman')
grid on
load 'idr_pi.mat'    %idr
xd_pi=ans.time; yd_pi=ans.data; yd_1=yd_pi(:,1); yd_2=yd_pi(:,2);
idr=stepinfo(yd_2,xd_pi,yd_1(end)); idr_SSE=yd_2(end)-yd_1(end)
```

```
subplot(2,2,3)
plot(xd_pi,yd_2,'r',xd_pi,yd_1,'b','linewidth',2)
xlabel('Time (seconds)','FontSize',14 ,'FontName','TimesNewRoman')
ylabel ('Current (A)','FontSize',14 ,'FontName','TimesNewRoman')
title('Direct Axis Current (I_d_r)','Color','blue', 'FontSize',14 ,'FontName',
'TimesNewRoman','FontWeight','normal')
legend ({'Reference Idr','Actual
I_d_r'},'Location','southeast','FontSize',14 ,'FontName','TimesNewRoman')
axis([0 5 -14000 2000])
grid on
figure(2)
load 'Vq_pi.mat'     % Quadraturevoltage
xvq_pi=ans.time; yvq_pi=ans.data; yvq_1=yvq_pi(:,1); subplot(2,2,2)                %
Quadrature voltage
plot(xvq_pi,yvq_1,'b','linewidth',2)
%xlabel('Time(seconds)', 'FontSize',16 ,'FontName','TimesNewRoman')
ylabel ('Voltage (V)', 'FontSize',16 ,'FontName','TimesNewRoman')
axis([0 5 0 1000])
title('Vqr','Color','blue', 'FontSize',16 ,'FontName','TimesNewRoman',
'FontWeight','normal')
legend ({'Vqr'},'Location','Northeast','FontSize',14 ,'FontName','TimesNewRoman')
grid on
load 'Vd_pi.mat'     % Direct axis reference voltage
xvd_pi=ans.time; yvd_pi=ans.data; yvd_1=yvd_pi(:,1);
subplot(2,2,1)                 % Direct axis reference voltage
plot(xvd_pi,yvd_1,'b','linewidth',2)
%xlabel('Time(seconds)', 'FontSize',16 ,'FontName','TimesNewRoman')
ylabel ('Voltage (V)', 'FontSize',16 ,'FontName','TimesNewRoman')
axis([0 5 0 1000])
title('Vdr','Color','blue', 'FontSize',16 ,'FontName','TimesNewRoman','FontWeight',
'normal')
legend ({'Vdr'},'Location','Northeast','FontSize',14 ,'FontName','TimesNewRoman')
grid on
load 'Ir_pi.mat'     % Rotor Current
```

```
xr_pi=ans.time; yr_pi=ans.data; yr_1=yr_pi(:,1); yr_2=yr_pi(:,2); yr_3=yr_pi(:,3);
subplot(2,2,3)                % Rotor Current
plot(xr_pi,yr_2,xr_pi,yr_1,xr_pi,yr_3,'linewidth',2); xlabel('Time (seconds)',
'FontSize',16 ,'FontName','TimesNewRoman'); ylabel ('Current (A)',
'FontSize',16 ,'FontName','TimesNewRoman'); axis([0 5 -20000 20000])
title('Rotor current','Color','blue',
'FontSize',16 ,'FontName','TimesNewRoman','FontWeight','normal')
legend ({'Ir (a)','Ir (b)','Ir (c)'},'Location','southeast','FontSize',14 ,'FontName',
'TimesNewRoman')
grid on
load 'Vs_pi.mat'    % Stator voltage
xv_pi=ans.time; yv_pi=ans.data; yv_1=yv_pi(:,1); yv_2=yv_pi(:,2); yv_21=yv_pi(:,3);
subplot(2,2,4)                % Stator voltage
plot(xv_pi,yv_1,xv_pi,yv_2,xv_pi,yv_21,'linewidth',2)
xlabel('Time (seconds)', 'FontSize',16 ,'FontName','TimesNewRoman')
ylabel ('Voltage (MV)', 'FontSize',16 ,'FontName','TimesNewRoman')
axis([0 5 -0.002 0.002])
title('Vs','Color','blue', 'FontSize',16 ,'FontName','TimesNewRoman','FontWeight','normal')
legend ({'Vs (a)','Vs (b)','Vs (c)'},'Location','southeast','FontSize',14 ,'FontName',
'TimesNewRoman')
grid on
%% System performance Subject to F controller
load 'omega_f.mat'    %Rotor Speed
xo_f=ans.time; yo_f=ans.data; yof_1=yo_f(:,1); yof_2=yo_f(:,2);
ome=stepinfo(yof_2,xo_f,yof_1(end)); ome_SSE=yof_2(end)-yof_1(end)
figure(1)
subplot(2,2,1)                % Rotor Speed
plot(xo_f,yof_2,'r',xo_f,yof_1,'b','linewidth',2)
%xlabel('Time(seconds)', 'FontSize',16 ,'FontName','TimesNewRoman')
ylabel ('Speed (rad/s)', 'FontSize',16 ,'FontName','TimesNewRoman')
axis([0 5 0 160])
title('Rotor Speed (\omega_r)','Color','blue', 'FontSize',16 ,'FontName','TimesNewRoman',
'FontWeight','normal')
```

```
legend ({'Actual Speed','Reference Speed'},'Location','southeast','FontSize',14 ,'FontName',
'TimesNewRoman')
grid on
load 'iqr_f.mat'    %iqr
xq_f=ans.time; y_f=ans.data; yf_1=y_f(:,1); yfq_2=y_f(:,2);
iqr=stepinfo(yfq_2,xq_f,yf_1(end)); iqr_SSE=yfq_2(end)-yf_1(end)
subplot(2,2,4)
plot(xq_f,yfq_2,'r',xq_f,yf_1,'b','linewidth',2)
xlabel('Time (seconds)','FontSize',16 ,'FontName','TimesNewRoman')
ylabel ('Current (A)','FontSize',16 ,'FontName','TimesNewRoman')
axis([0 5 -12000 4000])
title('Quadrature Axis Current (I_q_r)','Color','blue', 'FontSize',16 ,'FontName',
'TimesNewRoman','FontWeight','normal')
legend ({'Actual Iqr','Reference I_q_r'},'Location','southeast','FontSize',14 ,'FontName',
'TimesNewRoman')
grid on
load 'idr_f.mat'    %idr
xd_f=ans.time; yd_f=ans.data; yd_1=yd_f(:,1); yofd_2=yd_f(:,2);
idr=stepinfo(yofd_2,xd_f,yd_1(end)); idr_SSE=yofd_2(end)-yd_1(end)
subplot(2,2,3)
plot(xd_f,yofd_2,'r',xd_f,yd_1,'b','linewidth',2)
xlabel('Time (seconds)','FontSize',16 ,'FontName','TimesNewRoman')
ylabel ('Current (A)','FontSize',16 ,'FontName','TimesNewRoman')
title('Direct Axis Current (I_d_r)','Color','blue', 'FontSize',16 ,'FontName',
'TimesNewRoman','FontWeight','normal')
legend ({'Reference Idr','Actual I_d_r'},'Location','southeast','FontSize',14,'FontName',
'TimesNewRoman')
axis([0 5 -14000 2000])
grid on
load 'Tem_f.mat'      %Tem
xt_f=ans.time; yt_f=ans.data; yt_1=yt_f(:,1); yoft_2=yt_f(:,2);
tem=stepinfo(yoft_2,xt_f,yt_1(end)); tem_SSE=yoft_2(end)-yt_1(end)
subplot(2,2,2)
plot(xt_f,yoft_2,'r',xt_f,yt_1,'b','linewidth',2)
```

```
%xlabel('Time (seconds)','FontSize',14 ,'FontName','TimesNewRoman')
ylabel ('Torque (Nm)','FontSize',16 ,'FontName','TimesNewRoman')
axis([0 5 -40000 100000])
title('Electro-magnetic Torque (T_e_m)','Color','blue', 'FontSize',16 ,'FontName',
'TimesNewRoman','FontWeight','normal')
legend ({'Actual Tem','Reference T_e_m'},'Location','Northeast','FontSize',14,'FontName',
'TimesNewRoman')
grid on
figure(2)
load 'Vs_f.mat'     % Stator voltage
xv_f=ans.time; yv_f=ans.data; yv_1=yv_f(:,1); yv_2=yv_f(:,2); yv_21=yv_f(:,3);
subplot(2,2,4)                  % Stator voltage
plot(xv_f,yv_1,'r',xv_f,yv_2,'b',xv_f,yv_21,'g','linewidth',2)
xlabel('Time (seconds)', 'FontSize',16 ,'FontName','TimesNewRoman')
ylabel ('Voltage (V)', 'FontSize',16 ,'FontName','TimesNewRoman')
axis([0 5 -0.002 0.002])
title('Stator Voltage','Color','blue', 'FontSize',16 ,'FontName','TimesNewRoman',
'FontWeight','normal')
legend ({'Vqr'},'Location','southeast','FontSize',14 ,'FontName','TimesNewRoman')
grid on
load 'Vq_f.mat'     % Quadraturevoltage
xvq_f=ans.time; yvq_f=ans.data; yvq_1=yvq_f(:,1); subplot(2,2,2)   % Quadrature voltage
plot(xvq_f,yvq_1,'b','linewidth',2)
%xlabel('Time(seconds)', 'FontSize',16 ,'FontName','TimesNewRoman')
ylabel ('Voltage (V)', 'FontSize',16 ,'FontName','TimesNewRoman')
title('Vqr','Color','blue', 'FontSize',16 ,'FontName','TimesNewRoman','FontWeight',
'normal'); legend ({'Vqr'},'Location','Northeast','FontSize',14 ,'FontName',
'TimesNewRoman')
grid on
load 'Vd_f.mat'     % Direct axis reference voltage
xvd_f=ans.time; yvd_f=ans.data; yvd_1=yvd_f(:,1);
subplot(2,2,1)                  % Direct axis reference voltage
plot(xvd_f,yvd_1,'b','linewidth',2)
%xlabel('Time (seconds)', 'FontSize',16 ,'FontName','TimesNewRoman')
```

```
ylabel ('Voltage (V)', 'FontSize',16 ,'FontName','TimesNewRoman')
title('Vdr','Color','blue', 'FontSize',16 ,'FontName','TimesNewRoman','FontWeight',
'normal'); legend ({'Vdr'},'Location','Northeast','FontSize',14,'FontName',
'TimesNewRoman');grid on
load 'Ir_f.mat'     % Rotor Current
xr_f=ans.time; yr_f=ans.data; yr_1=yr_f(:,1); yr_2=yr_f(:,2); yr_3=yr_f(:,3);
subplot(2,2,3)                  % Rotor Current
plot(xr_f,yr_1,'r',xr_f,yr_2,'b',xr_f,yr_3,'g','linewidth',2)
xlabel('Time (seconds)', 'FontSize',16 ,'FontName','TimesNewRoman')
ylabel ('Ir (A)', 'FontSize',16 ,'FontName','TimesNewRoman'); axis([0 5 -20000 20000]);
title('Rotor Current','Color','blue', 'FontSize',16 ,'FontName', 'TimesNewRoman',
'FontWeight','normal'); legend ({'Ir (a)','Ir (b)','Ir (c)'},'Location','southeast','FontSize',14,
'FontName', 'TimesNewRoman'); grid on
%%  System performance Subject to FPI controller
load 'omega_fpi.mat'     %Rotor Speed
xo_fp=ans.time; yo_fp=ans.data; yo_1=yo_fp(:,1); yofp_2=yo_fp(:,2);
ome=stepinfo(yofp_2,xo_fp,yo_1(end)); ome_SSE=yofp_2(end)-yo_1(end)
figure(1)
subplot(2,2,1)                  % Rotor Speed
plot(xo_fp,yofp_2,'r',xo_fp,yo_1,'b','linewidth',2)
%xlabel('Time (seconds)', 'FontSize',16 ,'FontName','TimesNewRoman')
ylabel ('Speed (rad/s)', 'FontSize',16 ,'FontName','TimesNewRoman')
axis([0 5 0 160]); title('Rotor Speed (\omega_r)','Color','blue', 'FontSize',16 ,'FontName',
'TimesNewRoman','FontWeight','normal'); legend ({'Actual Speed','Reference Speed'},
'Location','southeast','FontSize',14 ,'FontName','TimesNewRoman')
grid on
load 'iqr_fpi.mat'    %iqr
xq_fp=ans.time; y_fp=ans.data; y_1=y_fp(:,1); yqfp_2=y_fp(:,2);
iqr=stepinfo(yqfp_2,xq_fp,y_1(end)); iqr_SSE=yqfp_2(end)-y_1(end); subplot(2,2,4)
plot(xq_fp,yqfp_2,'r',xq_fp,y_1,'b','linewidth',2)
xlabel('Time (seconds)','FontSize',16 ,'FontName','TimesNewRoman');  ylabel ('Current
(A)','FontSize',16,'FontName','TimesNewRoman'); axis([0 5 -12000 4000]);
title('Quadrature Axis Current (I_q_r)','Color','blue', 'FontSize',14, 'FontName',
```

```
'TimesNewRoman','FontWeight','normal'); legend ({'Actual Iqr','Reference I_q_r'},
'Location', 'southeast','FontSize',14,'FontName', 'TimesNewRoman'); grid on
load 'idr_fpi.mat'    %idr
xd_fp=ans.time; yd_fp=ans.data; yd_1=yd_fp(:,1); ydfp_2=yd_fp(:,2);
idr=stepinfo(ydfp_2,xd_fp,yd_1(end)); idr_SSE=ydfp_2(end)-yd_1(end)
subplot(2,2,3)
plot(xd_fp,ydfp_2,'r',xd_fp,yd_1,'b','linewidth',2)
xlabel('Time (seconds)','FontSize',16 ,'FontName','TimesNewRoman')
ylabel ('Current (A)','FontSize',16,'FontName','TimesNewRoman')
title('Direct Axis Current (I_d_r)','Color','blue', 'FontSize',16 ,'FontName',
'TimesNewRoman','FontWeight','normal'); legend ({'Reference Idr','Actual
I_d_r'},'Location','southeast','FontSize',14,'FontName', 'TimesNewRoman')
axis([0 5 -14000 2000])
grid on
load 'Tem_fpi.mat'      %Tem
xt_fp=ans.time; yt_fp=ans.data; yt_1=yt_fp(:,1); ytfp_2=yt_fp(:,2);
tem=stepinfo(ytfp_2,xt_fp,yt_1(end)); tem_SSE=ytfp_2(end)-yt_1(end)
subplot(2,2,2)
plot(xt_fp,ytfp_2,'r',xt_fp,yt_1,'b','linewidth',2)
%xlabel('Time (seconds)','FontSize',16 ,'FontName','TimesNewRoman')
ylabel ('Torque (Nm)','FontSize',16 ,'FontName','TimesNewRoman')
axis([0 5 -40000 100000]); title('Electro-magnetic Torque (T_e_m)','Color','blue',
'FontSize',16 ,'FontName', 'TimesNewRoman','FontWeight','normal')
legend ({'Actual Tem','Reference T_e_m'},'Location','Northeast','FontSize',14 ,'FontName',
'TimesNewRoman'); grid on
figure(2)
load 'Vs_fp.mat'     % Stator voltage
xv_fp=ans.time; yv_fp=ans.data; yv_1=yv_fp(:,1); yv_2=yv_fp(:,2); yv_21=yv_fp(:,3);
subplot(2,2,4)                  % Stator voltage
plot(xv_fp,yv_1,'r',xv_fp,yv_2,'b',xv_fp,yv_21,'y','linewidth',2)
xlabel('Time (seconds)', 'FontSize',16 ,'FontName','TimesNewRoman')
ylabel ('Voltage (V)', 'FontSize',16 ,'FontName','TimesNewRoman')
```

```
axis([0 5 -0.002 0.002]); title('Stator Voltage','Color','blue', 'FontSize',16 ,'FontName',
'TimesNewRoman', 'FontWeight','normal'); legend ({'Vs (a)','Vs (b)','Vs (c)'},'Location',
'southeast','FontSize',14,'FontName', 'TimesNewRoman'); grid on
load 'Vq_fp.mat'     % Quadraturevoltage
xvq_fp=ans.time; yvq_fp=ans.data; yvq_1=yvq_fp(:,1)
subplot(2,2,2)         % Quadrature voltage
plot(xvq_fp,yvq_1,'b','linewidth',2)
%xlabel('Time(seconds)', 'FontSize',16 ,'FontName','TimesNewRoman')
ylabel ('Voltage (V)', 'FontSize',16 ,'FontName','TimesNewRoman'); axis([0 5 0 1000])
title('Vqr','Color','blue', 'FontSize',16 ,'FontName','TimesNewRoman','FontWeight',
'normal'); legend ({'Vqr'},'Location','Northeast','FontSize',14 ,'FontName',
'TimesNewRoman'); grid on
load 'Vd_fp.mat'     % Direct axis reference voltage
xvd_fp=ans.time; yvd_fp=ans.data; yvd_1=yvd_fp(:,1);
subplot(2,2,1)   % Direct axis reference voltage
plot(xvd_fp,yvd_1,'b','linewidth',2)
%xlabel('Time(seconds)', 'FontSize',16 ,'FontName','TimesNewRoman')
ylabel ('Voltage (V)', 'FontSize',16 ,'FontName','TimesNewRoman');axis([0 5 0 1000])
title('Vdr','Color','blue', 'FontSize',16 ,'FontName','TimesNewRoman','FontWeight',
'normal') legend; ({'Vdr'},'Location','Northeast','FontSize',14,'FontName',
'TimesNewRoman'); grid on
load 'Ir_fp.mat'     % Rotor Current
xr_fp=ans.time; yr_fp=ans.data; yr_1=yr_fp(:,1); yr_2=yr_fp(:,2); yr_3=yr_fp(:,3);
 subplot(2,2,3)                  % Rotor Current
plot(xr_fp,yr_1,'r',xr_fp,yr_2,'b',xr_fp,yr_3,'y','linewidth',2)
xlabel('Time (seconds)', 'FontSize',16 ,'FontName','TimesNewRoman')
ylabel ('Current (A)', 'FontSize',16 ,'FontName','TimesNewRoman')
axis([0 5 -20000 20000]); title('Rotor Current','Color','blue', 'FontSize',16 ,'FontName',
'TimesNewRoman','FontWeight','normal'); legend ({'Ir (a)','Ir (b)','Ir (c)'},'Location',
'southeast','FontSize',14, 'FontName', 'TimesNewRoman'); grid on
%% System performance Subject to ANFIS controller
load 'omega_anfis.mat'     %Rotor Speed
xo_an=ans.time; yo_an=ans.data; yo_1=yo_an(:,1); yoan_2=yo_an(:,2);
ome=stepinfo(yoan_2,xo_an,yo_1(end)); ome_SSE=yoan_2(end)-yo_1(end)
```

```
figure(1)
subplot(2,2,1)                  % Rotor Speed
plot(xo_an,yoan_2,'r',xo_an,yo_1,'b','linewidth',2)
%xlabel('Time(seconds)', 'FontSize',16 ,'FontName','TimesNewRoman')
ylabel ('Speed (rad/s)', 'FontSize',16 ,'FontName','TimesNewRoman'); axis([0 5 0 160]);
title('Rotor Speed (\omega_r)','Color','blue', 'FontSize',16 ,'FontName','TimesNewRoman',
'FontWeight','normal'); legend ({'Actual Speed','Reference Speed'},'Location','southeast',
'FontSize',14,'FontName', 'TimesNewRoman'); grid on
load 'iqr_anfis.mat'    %iqr
xq_an=ans.time; y_an=ans.data; y_1=y_an(:,1); yan_2=y_an(:,2);
iqr=stepinfo(yan_2,xq_an,y_1(end)); iqr_SSE=yan_2(end)-y_1(end); subplot(2,2,4)
plot(xq_an,yan_2,'r',xq_an,y_1,'b','linewidth',2); xlabel('Time (seconds)','FontSize',16,
'FontName','TimesNewRoman'); ylabel ('Current (A)','FontSize',16 ,'FontName',
'TimesNewRoman'); axis([0 5 -12000 4000]); title('Quadrature Axis Current (I_q_r)',
'Color','blue', 'FontSize',16,'FontName', 'TimesNewRoman','FontWeight','normal')
legend ({'Actual Iqr','Reference I_q_r'},'Location','southeast','FontSize',14,
'FontName','TimesNewRoman')
grid on
load 'idr_anfis.mat'    %idr
xd_an=ans.time; yd_fp=ans.data; yd_1=yd_fp(:,1); ydan_2=yd_fp(:,2);
idr=stepinfo(ydan_2,xd_an,yd_1(end)); idr_SSE=ydan_2(end)-yd_1(end)
subplot(2,2,3)
plot(xd_an,ydan_2,'r',xd_an,yd_1,'b','linewidth',2); xlabel('Time (seconds)','FontSize',16,
'FontName','TimesNewRoman'); ylabel ('Current (A)','FontSize',16,'FontName',
'TimesNewRoman'); title('Direct Axis Current (I_d_r)','Color','blue', 'FontSize',16,
'FontName','TimesNewRoman','FontWeight','normal'); legend ({'Reference Idr','Actual
I_d_r'},'Location','southeast','FontSize',14 ,'FontName', 'TimesNewRoman')
axis([0 5 -14000 2000]); grid on
load 'Tem_pi.mat'      %Tem
xt_an=ans.time; yt_an=ans.data; yt_1=yt_an(:,1); ytan_2=yt_an(:,2);
tem=stepinfo(ytan_2,xt_an,yt_1(end)); tem_SSE=ytan_2(end)-yt_1(end)
subplot(2,2,2)
plot(xt_an,ytan_2,'r',xt_an,yt_1,'b','linewidth',2)
%xlabel('Time(seconds)','FontSize',16 ,'FontName','TimesNewRoman')
```

```
ylabel ('Torque(Nm)','FontSize',16,'FontName','TimesNewRoman')
axis([0 5 -40000 100000]); title('Electro-magnetic Torque (T_e_m)','Color','blue',
'FontSize',16 ,'FontName','TimesNewRoman','FontWeight','normal')
legend ({'Actual Tem','Reference T_e_m'},'Location','Northeast','FontSize',14,'FontName',
'TimesNewRoman'); grid on
figure(2)
 load 'Vs_anfis.mat'     % Stator voltage
xv_an=ans.time; yv_an=ans.data; yv_1=yv_an(:,1); yv_2=yv_an(:,2); yv_21=yv_an(:,3);
subplot(2,2,4)                  % Stator voltage
plot(xv_an,yv_1,'r',xv_an,yv_2,'b',xv_an,yv_21,'g','linewidth',2)
xlabel('Time (seconds)', 'FontSize',16 ,'FontName','TimesNewRoman')
ylabel ('Voltage (V)', 'FontSize',16 ,'FontName','TimesNewRoman')
axis([0 5 -0.002 0.002]); title('Stator Voltage','Color','blue', 'FontSize',16 ,'FontName',
'TimesNewRoman','FontWeight','normal'); legend ({'Vs (a)','Vs (b)','Vs (c)'},'Location',
'southeast','FontSize',14 ,'FontName','TimesNewRoman')
grid on
load 'Vq_anfis.mat'     % Quadraturevoltage
xvq_an=ans.time; yvq_an=ans.data; yvq_1=yvq_an(:,1);
subplot(2,2,2)                  % Quadrature voltage
plot(xvq_an,yvq_1,'b','linewidth',2)
%xlabel('Time(seconds)', 'FontSize',16 ,'FontName','TimesNewRoman')
ylabel ('Voltage (V)', 'FontSize',16 ,'FontName','TimesNewRoman')
axis([0 5 0 1000]); title('Vqr','Color','blue', 'FontSize',16 ,'FontName',
'TimesNewRoman','FontWeight','normal'); legend ({'Vqr'},'Location','northeast',
'FontSize',14,'FontName','TimesNewRoman')
grid on
load 'Vd_anfis.mat'     % Direct axis reference voltage
xvd_an=ans.time; yvd_an=ans.data; yvd_1=yvd_an(:,1);
subplot(2,2,1)                  % Direct axis reference voltage
plot(xvd_an,yvd_1,'b','linewidth',2)
%xlabel('Time(seconds)', 'FontSize',16 ,'FontName','TimesNewRoman')
ylabel ('Voltage (V)', 'FontSize',16 ,'FontName','TimesNewRoman')
```

```
axis([0 5 -10 1000]); title('Vdr','Color','blue','FontSize',16 ,'FontName','TimesNewRoman',
'FontWeight', 'normal'); legend ({'Vdr'},'Location','Northeast','FontSize',14,'FontName',
'TimesNewRoman'); grid on
load 'Ir_anfis.mat'     % Rotor Current
xr_an=ans.time; yr_an=ans.data; yr_1=yr_an(:,1); yr_2=yr_an(:,2); yr_3=yr_an(:,3);
subplot(2,2,3)                  % Rotor Current
plot(xr_an,yr_1,'r',xr_an,yr_2,'b',xr_an,yr_3,'linewidth',2); xlabel('Time (seconds)',
'FontSize',16 ,'FontName','TimesNewRoman') ylabel ('Current (A)', 'FontSize',16,
'FontName','TimesNewRoman'); axis([0 5 -20000 20000]); title('Rotor Current',
'Color','blue', 'FontSize',16 ,'FontName','TimesNewRoman’, 'FontWeight','normal')
legend ({'Ir (a)','Ir (b)','Ir (c)'},'Location','southeast','FontSize',14,'FontName',
'TimesNewRoman'); grid on
```