Adaptive Neuro-Fuzzy Inference System Based Genetic Algorithm Tuned Fractional Order Proportional Integral Derivative Speed Control of Vector Controlled Induction Motor

Belaso Beyene Dergaso

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 Master of Science Degree in Electrical Power and Control Engineering (Control)

Office of Graduate Studies

Adama Science and Technology University

July 2021

Adama, Ethiopia

Adaptive Neuro-Fuzzy Inference System Based Genetic Algorithm Tuned Fractional Order Proportional Integral Derivative Speed Control of Vector Controlled Induction Motor

Belaso Beyene Dergaso

Advisor:

Dr. Tafesse Asrat

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 Master of Science Degree in Electrical Power and Control Engineering (Control)

Office of Graduate Studies

Adama Science and Technology University

July 2021

Adama, Ethiopia

APPROVAL SHEET

We hereby certify that the recommendations and suggestions made by the board of examiners are appropriately incorporated into the final version of the dissertation entitled "Adaptive Neuro-Fuzzy Inference System Based Genetic Algorithm Tuned FOPID Speed Control of Vector Controlled Induction Motor" by **Belaso Beyene**.

Dr. Tafesse Asrat		
Advisor	Signature	Date

We, the undersigned, members of the Board of Examiners of the dissertation open defense by **Belaso Beyene** have read and evaluated the dissertation entitled "Adaptive Neuro-Fuzzy Inference System Based Genetic Algorithm Tuned FOPID Speed Control of Vector Controlled Induction Motor" and examined the candidate during the open defense. This is, therefore, to certify that the dissertation is accepted for partial fulfillment of the requirement of the degree of Master of Science in Electrical Power and Control Engineering (Control).

Chairperson	Signature	Date
Internal Examiner	Signature	Date
External Examiner	Signature	Date

Finally, approval and acceptance of the thesis is contingent upon 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

DECLARATION

I hereby declare that this Master Thesis entitled "Adaptive Neuro-Fuzzy Inference System Based Genetic Algorithm Tuned FOPID Speed Control of Vector Controlled Induction Motor" 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.

Name: Belaso Beyene Dergaso

Signature:________________

This MSc thesis has been submitted for examination with my approval as a thesis advisor.

Name: Dr. Tafesse Asrat

Signature: _______________

RECOMMENDATION

I certify that I have read and revised the dissertation entitled "Adaptive Neuro-Fuzzy Inference System Based Genetic Algorithm Tuned FOPID Speed Control of Vector Controlled Induction Motor" prepared under my guidance by **Belaso Beyene** submitted in partial fulfillment of the requirements for the degree of Master of Science. Therefore, I recommend the submission of the dissertation to the department for further review and defense.

Dr. Tafesse Asrat

______________________	______________	______________
Advisor	Signature	Date

ACKNOWLEDGMENT

First, my hearty gratitude goes to the almighty God. Then I am highly grateful to my supervisor **Dr. Tafesse Asrat** for his supervision, friendly attitude, and sharing of many good ideas. Regardless of his often high workload, he always makes time to answer questions as well as giving encouraging advice. You are highly appreciated!

I am also very grateful to **Dr. Tefera Yetayew** for lots of advice and helping in preparing the thesis documentation, but also for good ideas and discussions regarding everything from implementations of the system in Simulink in general.

I would moreover like to thank **Prof. Gang Gyoo Jin** for his patience, helpful attitude, and sharing of his work regarding the implementation of the Simulink modeling.

Furthermore, I would also like to thank the EPCE department head **Mr. Mesfin Megra (MSc.)**, and department Postgraduate Coordinator **Dr. Milkias Berhanu** for their initiation for accomplishing the thesis in proper time.

Finally, I would like to thank my whole family. Especially, words cannot express how grateful I am to Miss Selam Biru and friends for their love, encouragement, and all of the sacrifices that they have made on my behalf.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ACRONYMS

AC	Alternating Current
ANFIS	Adaptive Neurofuzzy Inference System
ANN	Artificial Neural Network
CSI	Current Source Inverter
DC	Direct Current
DFOC	Direct Field Oriented Control
DQ	Direct and Quadrature axis
DSP	Digital Signal Processor
FLC	Fuzzy Logic Controller
FOC	Field Oriented Control
FOMCON	Fractional Order Modeling and Control
FOPID	Fractional Order Proportional Integral Derivative
GA	Genetic Algorithm
GL	Grunwald-Letnikov
IGBT	Insulated Gate Bipolar Transistor
IM	Induction Motor
IVCIM	Indirect Vector Controlled Induction Motor
MATLAB	Matrix Laboratory
NN	Neural Network
PID	Proportional Integral Derivative
PWM	Pulse Width Modulation
RL	Riemann-Liouville
SCIM	Squirrel Cage Induction Motors
SRRF	Synchronously Rotating Reference Frame
SVM	Space Vector Modulation
SVPWM	Space Vector Pulse Width Modulation
TSK	Tagaki Sugno Kang
VSI	Voltage Source Inverter

LIST OF SYMBOLS

I	current
J	The moment of inertia
K_i	Integral gain
K_p	Proportional gain
L_r	Rotor self-inductance
L_s	Stator self-inductance
L_m	Mutual inductance
N	Natural number
P	Number of poles
R	Real number
R_r	Rotor resistance
R_s	Stator resistance
T_L	Load torque
T_e	Electromagnetic torque
V	Voltage
V_DC	DC Link Voltage
X_m	Stator magnetizing reactance
X_r	Rotor self-reactance
X_s	Stator self-reactance
ω_e	Angular synchronous speed
ω_r	Angular rotor speed
ω_{sl}	Angular slip speed
θ_e	Angle b/n the synchronous and frame
θ_r	rotor position
ψ	flux
λ	Integral Order
Γ	Euler’s gamma function

ABSTRACT

In the industrial sector especially in the area where variable frequency drive is needed, induction motors play a vital role. So without proper controlling of the speed, it is virtually impossible to achieve the desired task for a specific application using an induction motor. Three-phase Induction motor, specifically the squirrel cage induction motors are exhibits a nonlinear behavior on their sudden changes in load and variable speed applications as well as parameter variations. That is why an advanced controller is needed to enhance induction motor performance. Thus, this thesis aims to design an Adaptive Neuro-Fuzzy Inference System based Genetic Algorithm Tuned Fractional Order Proportional Integral Derivative speed controller for a vector-controlled induction motor in which parameters of Fractional Order Proportional Integral Derivative are optimized by using a genetic algorithm. So that the tracking performance of the induction motor drive will be improved. To achieve this, mathematical modeling is done based on vector control, and also for implementation purposes, the corresponding model is done using MATLAB software tools. The performance of induction motor based on the Fractional Order Proportional Integral Derivative speed controller was evaluated under the application of sudden load change, motor parameter variation, and giving variable reference speed. Accordingly, the GA- Fractional Order Proportional Integral Derivative speed controller has a settling time of 0.412sec, a rising time of 0.355, 9% overshoot, and 7% steady-state error under 120 rad/s rated speed. On the other hand, the Adaptive Neuro-Fuzzy Inference System based Genetic Algorithm Fractional Order Proportional Integral Derivative speed controller has better performance than Genetic Algorithm Tuned Fractional Order Proportional Integral Derivative and it has a settling time of 0.403sec, rising time of 0.307sec, 6% overshoot, and 0.5% steady-state error. In general, the performance evaluations of the implemented controllers revealed that Adaptive Neuro-Fuzzy Inference System based Genetic Algorithm Tuned Fractional Order Proportional Integral Derivative speed controller along with space vector pulse width modulation outperformed Genetic Algorithm Tuned Fractional Order Proportional Integral Derivative.

Key Words: *Adaptive Neuro-fuzzy inference system, Field Oriented Control (vector control), Fractional order PID (FOPID), Genetic algorithm, Induction Motor, MATLAB software, Self-tuning, Space vector pulse width modulation, Speed Control.*

CHAPTER ONE

INTRODUCTION

1.1 Background

In 1888, Tesla introduced the alternating current (AC) induction motor primarily. Tesla believed that his invention is more useful and applicable than the DC motor. The Three-phase induction motor has been a reliable electromechanical energy conversion device for over 100 years. Induction motors, particularly the squirrel cage induction motors (SCIM) have been widely used in industrial applications because of their several inherent advantages such as their simple construction, robustness, reliability, low cost, and fewer maintenance needs [1]. It is an electromagnetic device, which converts electrical energy to mechanical energy. It has stator and rotor set up on bearings and separated with the aid of using an air gap. Both the stator and rotor windings of IM convey alternating current (AC). The AC current is applied to the stator winding and the rotor receives AC via induction. A three-phase IM is designed to operate from a three-phase AC supply. For variable speed drives, the supply is generally an inverter that uses power switches to produce approximately sinusoidal voltages and currents of controllable magnitude, frequency, and phase [2]. IM is of two sorts squirrel cage and wound rotor. In squirrel cage kind IM, the rotor winding is inaccessible. Squirrel cage IM is so common in the industry that in many plants nearly no other type of machine can be found. In IM, no moving contacts, commutator, and brushes as DC motor or slip rings, and brushes as in synchronous machines are needed. This arrangement of IM substantially will increase the reliability of IM and removes the chance of sparking, allowing squirrel cage IM to be safely used in harsh environments, even in an explosive atmosphere. An additional degree of ruggedness is provided by the lack of wiring in the rotor, whose winding consists of an uninsulated metal bar forming a “squirrel cage” that gives the name to the motor. Such a robust rotor can run at high speed and resist heavy electrical and mechanical overload [3]. Previously, IM was operated uncontrolled, running at constant speed near to rated speed, even an application where efficient control over their speed could be very advantageous [4]. However, control of IM is not a simple issue due to its multivariable, and nonlinear nature. In addition, some of the parameters are time-variant, for example, the mutual inductance between the stator and rotor, and some of the state variables like stator flux, rotor current (for squirrel cage rotor) are not accessible for measurement [2],[5]. In the past, DC motors were used extensively in areas where variable speed operation is needed since the field and armature current could control their flux and torque easily. Now a day the shift from DC motor to IM is continued because of the development of power electronics converters, high computing microprocessors, and DSP, in addition to the development of AC motor control theory such as field-oriented control, and intelligent control. An electric drive consists of various components: driven mechanical system (load), electric machine, electric power converter, controller, etc. interconnected as shown in figure (1.1) below. Without the right control, it is virtually impossible to achieve the desired mission for any industrial application. However, there are different control schemes for induction motor drives. These are scalar control, vector or field-oriented control, direct torque control, and adaptive control, etc [6],[7]. This thesis briefly describes the design of the most efficient form of a vector control scheme of the indirect field-orientated control method. The major parts of the indirect vector control method are the speed control method, the independent torque, and flux control, the transformation of a multiphase stationary coordinate system into two axes stationary and rotating coordinate, and the effective Space Vector Pulse Width Modulation pattern generation. In the last several years, an excellent effort has been made on the elimination of speed sensors or shaft positions sensors for drives. The ongoing research has been concentrated on the elimination of the speed sensor at the machine shaft without reducing the dynamic performance of the drive control system [8],[9]. Generally, high-performance vector control of induction motor drives requires speed or position information for their operation. However, those Sensors hooked up at the machine shaft aren't acceptable for a lot of reasons. First of all, the cost is substantial. Secondly, their mounting requires a machine with two shaft ends available - one for the sensor and therefore the other one for the load coupling. Thirdly, electrical signals from the shaft sensor should be taken to the controller, and consequently, the sensor needs a power supply that requires extra cabling. Finally, the presence of a shaft sensor reduces the mechanical robustness of the machine and its reliability [10]. Due to those reasons, speed Sensorless systems, in which rotor speed measurements aren't available, are favored and find applications in many areas for speed regulation, load torque variation, and speed tracking purposes. Variable frequency drives are controlled by a pulse width modulator (PWM) switching technique.

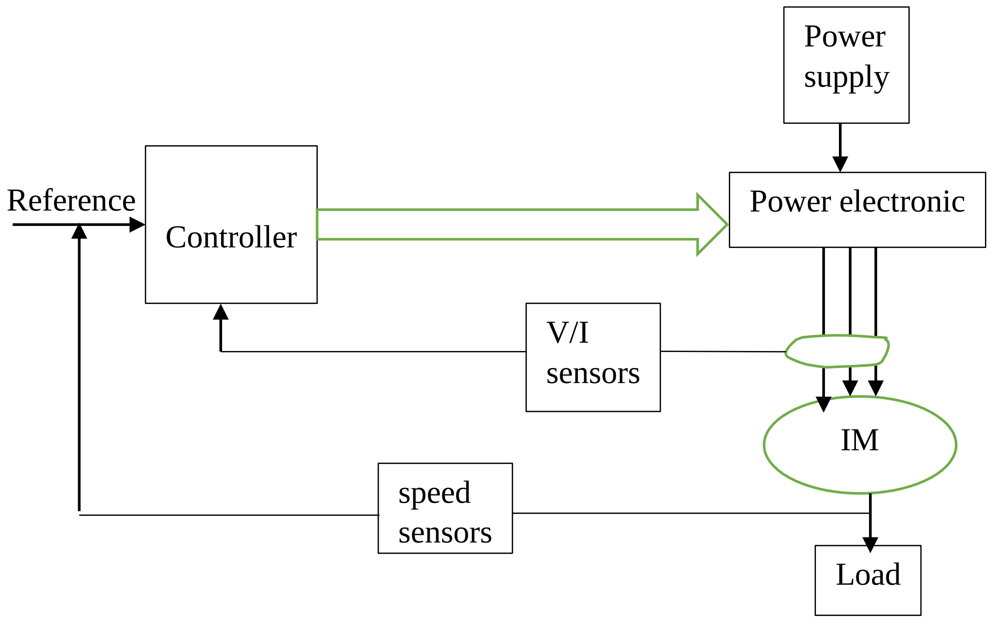


Figure 1. 1 Electric drive general block diagram

1.2 Problem Statement

An induction motor has many interesting features like high power to mass ratio, relatively, low cost, and maintenance-free. Three-phase Induction motor, specifically the squirrel cage induction motors are exhibits a nonlinear behavior on their sudden changes in load and variable speed applications as well as parameter variations. Due to this when the motor starts without load, it has high starting torque and at the same time, the motor can attain maximum overshoot which results in more vibration on the machine. Due to this high vibration, the motor life will be very short. Furthermore, IM is a dynamic nonlinear system with uncertainty in the machine parameters.

Many works on the controller have focused only on the advantages of the selected controllers. So it needs a critical investigation on its possible disadvantages and limitations must be considered.

For instance, people working in fuzzy logic control say no mathematical model is needed to design a fuzzy controller. Moreover, the most important advantage of FLC is considered to be no need for the mathematical model in the design of FLC. This eliminates the challenges control engineers encounter to bring accurate mathematical models. However, without the mathematical model and simulation analysis, the controller cannot be implemented and the performance of FLCs depends on rules and membership functions (MFs), which are determined by a trial and error procedure. The PID controller, on the other hand, has a well-defined design procedure. PID controller parameters can be determined from the linearized system model but systems in nature are nonlinear so the PID controller is good in the linear operating region. But, the performance of the PID controller diminishes under the uncertainties of the motor which include unknown load on the motor and its resistance variations, parameter variation, and disturbances. Another thing is that the induction motor is sensitive to parameter variation in which speed is immediately fluctuating when the value of the motor parameter is changed from the set point.

In the speed control of the induction motor, it is possible to use a speed sensor for accessing actual speed information. This speed sensor (position sensor which senses the position of rotor continuously and it should be very high precision) has a disadvantage like an extra cost, system configuration, and reducing reliability.

To overcome these problems from an uncertainty perspective, an intelligent controller is needed to enhance IM performance. Based on this I want to apply ANFIS based FOPID speed controller in which GA optimizes their parameters.

1.3 Objective of the Thesis

1.3.1 General Objective

The main objective of this thesis is to design ANFIS based genetic algorithm optimized fractional-order PID controllers for speed control of indirect vector controlled Induction motor.

1.3.2 Specific Objective

- To investigate fractional-order calculus.
- To use the Genetic algorithm tuning method to determine fractional order PID parameters.
- Analyzing the characteristics and modeling of the IM drive system.
- Studying adaptive neuro-fuzzy inference system.
- Developing a FOPID speed control law based on the nonlinear model of the IM drive system.
- To simulate ANFIS based GA-FOPID indirect field-oriented speed control of an induction motor drive in MATLAB/Simulink software.

1.4 Scope of the Study

This thesis focuses on the mathematical modeling and simulation design for indirect vector control of an induction motor. Design the closed-loop ANFIS based GA-FOPID and speed controller for an induction motor. The performance of the ANFIS based GA- FOPID controller for speed control of an induction motor is discussed. Generally, the scope of this thesis is analyzing the performance of FOPID through the GA optimization method and the use of ANFIS to train error input to mimic the IM motor output to set point. The overall system for indirect vector control of an induction motor is realized in MATLAB/Simulink Software.

1.5 Limitations of the Thesis

The thesis will be limited to simulation of the system using MATLAB/Simulink by showing how the different components of the system interact with each other in which there is no prototype or hardware implementation. This is because fact that implementation of the real system is difficult because the component used to implement is not available in our country and along with it the additional time and cost incurred in making the actual implementation.

1.6 Significance of the Study

This thesis will contribute the listed advantages below in different working areas. Running an induction motor without a position mechanical sensor has an advantage over an induction motor with a position sensor:

- It minimizes the drive cost.
- It decreases the mechanical size of the machine
- It increases the reliability of the system.
- It has a fast response due to FOPID has an additional 2DOF.
- It eliminates manual training data and generating fuzzy rules.

1.7 Organization of the thesis

The thesis has been organized into five chapters. Chapter one presents the introduction, statement of the problem, objectives of the study, scope, limitations, motivation of the study, significance, and organization of the thesis. The second chapter discusses fractional-order modeling of induction motor, fractional calculus, reference frame transformation, reference frame model of induction motor. Chapter three presents about control of induction motor; concept of vector control, ANFIS, FOPID control method, SVPWM, and GA optimization mechanism. Chapter four presents simulation results obtained using MATLAB/SIMULINK and discussions of these results. Finally, Chapter five is talking about the conclusion and recommendations for future works.

CHAPTER TWO

LITERATURE REVIEW AND BASIC THEORIES

2.1 Chapter Overview

In this chapter, the background and varieties of electric motors are discussed. Here basic closely related works especially in speed control mechanisms as well as flux and speed estimation for induction motor are reviewed. The introduction to induction motor and varieties of the control techniques, construction, and principle of operation are reviewed and discussed.

2.2 Previous Works

Some of the papers and articles that have been reviewed to do this thesis are the following:-

1. In [11], the novel approach for speed control of three-Phase induction motor is primarily based totally on indirect field-oriented control presented in the literature. It is shown that using an appropriate transformation matrix (TM) for stator current variables, the unbalanced three-phase induction motor equations can be changed into balanced equations. Simulation results confirmed the good speed and torque responses obtained using the proposed technique due to rotor speed follows reference speed. The downside of the presented approach is that the motor speed of a three-phase induction motor must be measured, which needs a speed sensor.

   To overcome this problem this thesis has applied Sensor-less speed control of the indirect FOC method for an induction motor.

2. In [12], Sensor-less Control of Induction Motor using Simulink by Direct Synthesis Technique presented in the literature. The sensor-less control of induction motor using direct synthesis from the state equations has been presented for starting, steady-state, transient, and speed reversal without using any shaft encoder as in the case of vector control. In the simulation result, the estimated speed has a high steady-state error and works only at high-speed range and also high starting current. But for any drive high starting current is not advisable as well as it needs direct control of i_q& i_d and it has high distortion. Therefore, this thesis work tries to improve the direct synthesis technique by using an ANFIS controller to train speed error until getting minimum error response.

3. In [13], the performance comparison between conventional PI and PI-Like fuzzy speed controller for a three-phase induction motor is performed in the literature. This thesis introduced a dynamic model of a three-phase induction motor and speed control of the induction motor for field-oriented control. The comparison between Conventional PI and PI-Like fuzzy speed controller for a three-phase induction motor is done in this research work. From the result the performance of IM with the fuzzy logic controller is more stable than with a conventional PI controller, the speed almost has got to be constant while load torque is applied at a selected instant due to the instantaneous response of FLC. While the fuzzy controller gives better control performance than the PI controller, it shows high overshoot, rise time, and settling time within the speed of the motor. The opposite drawback of the study is, it doesn't discuss the applying of variable speed and regenerative speed to controlling the speed of the induction motor.
4. In [14], an optimization method in fractional-order control of electrical drives has also been discussed. This paper presents a comparative study on the optimization methods in FOPI controller design for induction motors. Choosing the most effective optimization algorithm for tuning fractional-order controllers is crucial to get the simplest system response. During this study genetic algorithm, local search, nonlinear sequential quadratic programming, and particle swarm optimization algorithms are used to tune a fractional PI controller for induction motors. The gap here is optimization is completed just for PI but in PID the derivative has its task in controlling an induction motor called improving transient response through high-frequency compensation by differentiation.
5. In [15], a comparative study on speed control of indirect vector control of induction motor using fuzzy logic and PI controller was presented and shows fuzzy logic controller can replace the PI controller. But Finding rules are time-consuming and not accurate.
6. In [16], a simple method of IM control is to vary the provision stator voltage at supply frequency. This method of speed control has poor dynamic and static performance. This method also features a high slip power loss. An efficient method of speed control for IM is to alter the stator frequency since the speed is near synchronous speed the operating slip is tiny, and slip power loss within the rotor circuit is tiny. However, this method of speed control requires a frequency converter, which is rear.

7. In [17], Slip current control, v/f control uses frequency converter. In v/f control, both voltage and frequency varied kept their ratio constant. Both v/f control and slip current control fail to produce satisfactory transient performance. These control strategies are called scalar control during which only the magnitude and frequency of stator current and voltage are controlled and thanks to this, it has poor dynamic and static performance.

2.3 Electric Motor Varieties

Various kinds of electric motors are classified and differentiated based on structure and power sources as shown in Figure 2.1. Generally, electric motors are categorized into three types: DC motor, AC motor, and other types of motors [18]. Therefore IM motor is categorized under AC motor types.

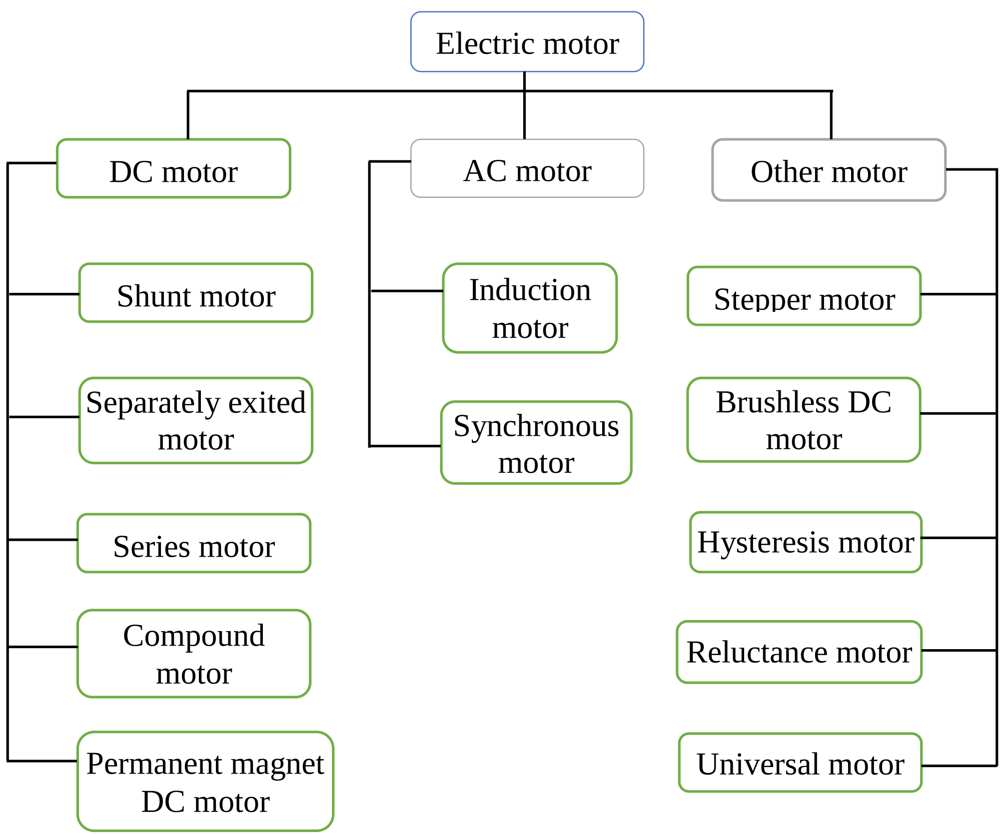


Figure 2. 1 Motor classification

2.3.1 Variable Frequency Drives

Devices providing speed control have a variety of names, including:

- Inverter drives

- Adjustable speed drives
- Variable speed drive (VSDs)
- Variable frequency drives (VFDs)
- Vector control drives.

The name often varies with the kind of motor paired with the device. For instance, the term VFD is often related to three-phase AC induction motors. Three-phase induction motors are often paired with variable speed drives. Nowadays, an AC motor is preferred to DC motors, specifically, an induction motor because of its low cost, less maintenance, lower weight, high efficiency, improved ruggedness, and reliability. All these features make wide use of induction motors in many areas of industrial application [10]. While variable speed control devices are not all identical, the primary principles of operation remain the same. This section of the thesis refers to variable frequency control devices generically as "VFDs" and provides a broad description of their features, efficiency, and application. Any notable differentiation in the construction or operation of a variable frequency control device is addressed in subsequent.

Variable loads occur frequently in the commercial and residential sectors. By using a variable frequency drive, the motor speed can be adjusted to match the system's actual operating requirements.

The synchronous speed N_s of an ac motor is related to supply frequency f and poles P by the equation which is used for scalar control mechanism is as follows.

$$N_s = \frac{120f}{p} \tag{2.1}$$

Where N_s is the speed in rpm, f is the frequency of the applied excitation in Hz, and P is the number of the poles. The rotor attempts to run at this speed, also known as the synchronous speed, but never quite reaches this speed, rotating instead at a speed N_r due to slip. The difference between the source angular frequency (ω_s) and the angular frequency of the rotor electrical speed (ω_r) is termed the angular slip frequency (ω_{sl}) while the ratio of the slip speed to the stator speed is the actual slip (S) [1].

$$S = \frac{N_s - N_r}{N_s} = \frac{\omega_s - \omega_r}{\omega_s} = \frac{\omega_{sl}}{\omega_s} \tag{2.2}$$

Note that slip can also be defined in terms of N_r and N_s as shown in Equation 2.2 where N_r is the rotor speed in rpm.

2.4 Structure of the Three Phase Induction Motor

Actually, every induction motor has three main parts; rotor, stator, and enclosure. The stator and also the rotor do the work and also the enclosure protects the stator and rotor. Most induction motors are the rotary type with basically a stationary stator and a rotating rotor. The stationary stator features a cylindrical core that's housed inside a metal frame. The stator magnetic core is made by stacking thin electrical steel laminations with uniformly spaced slots stamped within the inner circumference to accommodate the three distributed stator windings. The stator windings are formed by connecting coils of copper or aluminum conductors that are insulated from the slot walls. The rotor consists of a cylindrical laminated iron core with uniformly spaced peripheral slots to accommodate the rotor windings. AC induction motor is represented by a stator and a concentric rotor and a narrow air gap between the smooth surfaces of the rotor and the stator. The connection between the stator and the rotor is formed by the flux, which implies there is no mechanical connection between them aside from the bearings [1].

There are two different types of induction motors depend on the rotor;

1. Wound rotor induction motor

This type of IM features a three-phase winding within the rotor, similar to the stator winding. The rotor winding terminals are connected to the three slip rings which turn with the rotor. The slip rings/brushes have external resistors to be connected serial with the winding. The external resistors are mainly used during start-up under normal running conditions the windings are short-circuited externally.

2. Squirrel cage IM

This type of IM consists of a series of conducting bars laid into slots carved in the face of the rotor and shorted at either end by the big shorting rings. The construction of the slip ring motor is complicated because it consists of a slip ring and brushes whereas the construction of the squirrel cage induction motor is simple [6]. The maintenance cost of the wound IM is high as compared to the squirrel cage motor because the slip ring motor consists of brushes and rings. The slip ring motor has a brush transferring the power whereas the squirrel cage motor is brushless.

The copper loss in the wound motor is high as compared to a squirrel cage motor. The starting current of the wound rotor motor is low because it is controlled by a resistance circuit whereas it is high in the squirrel cage motor. The wound rotor motor is mostly used in the place where high starting torque is required like a hoist, crane, etc. the squirrel cage motor is used in drilling machines, lath machines, etc.[1].

In general, because of the above reviewed- literature merit, this thesis work focuses on the speed control of three-phase squirrel cage motor in closed-loop drives system control. The squirrel cage induction motor is shown in Figure 2.2.

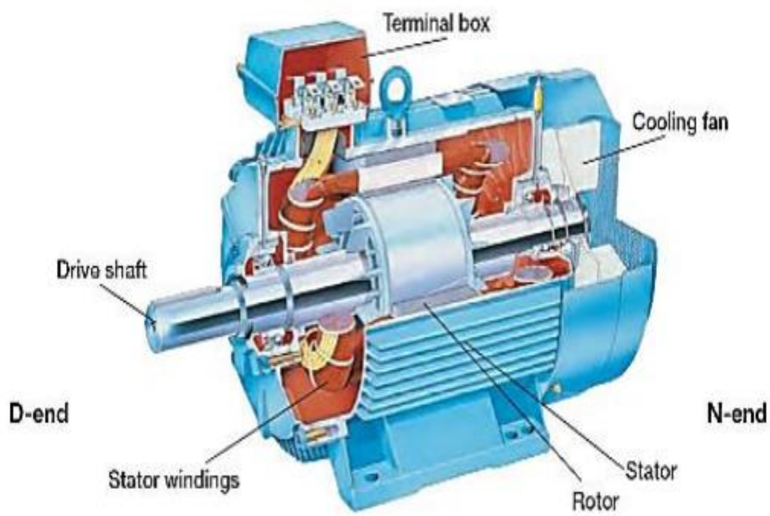


Figure 2. 2 Squirrel Cage Induction Motor [L1]

The working rule of the IM relies on the generated rotating magnetic field during which a three-phase induction motor is connected to the acceptable AC voltage source, it produces stator current, and every one of the three coils for every phase is fed with an AC. The current induced in each phase generate magnetic field intensity and this causes a flow of magnetic flux. The rotating magnetic field interacts with a collection of conductors arranged on the rotor and is short-circuited at the ends with two rings. This interaction between the magnetic field and also the conductor induces a current within the bars. Current flow within the conductors of the rotor through the short-circuiting rings at the end. This current, in turn, produces a magnetic field. There's a difference between revolving field speed and rotor speed then the revolving field induces a voltage within the rotor winding and this difference is termed because of the slip speed [1]. The induced voltage is present in a rotor current that generates flux in the counter direction by the stator windings. The interaction between the rotor magnetic field and the squirrel cage bars induces torque and causes rotation.

2.4.1 Control Techniques for IM

Induction motors are the most widely used electrical motors because of their reliability, low cost, and robustness. However, induction motors do not inherently have the potential of variable speed operation. Due to this reason, earlier dc motors were used in most of the electrical drives but the recent developments in speed control methods of the induction motor lead to their large-scale use in most electrical drives. There is a different method of speed control of the induction motor. From those methods, the general classification of the induction motor control methods is presented in Figure 2.3 [19].

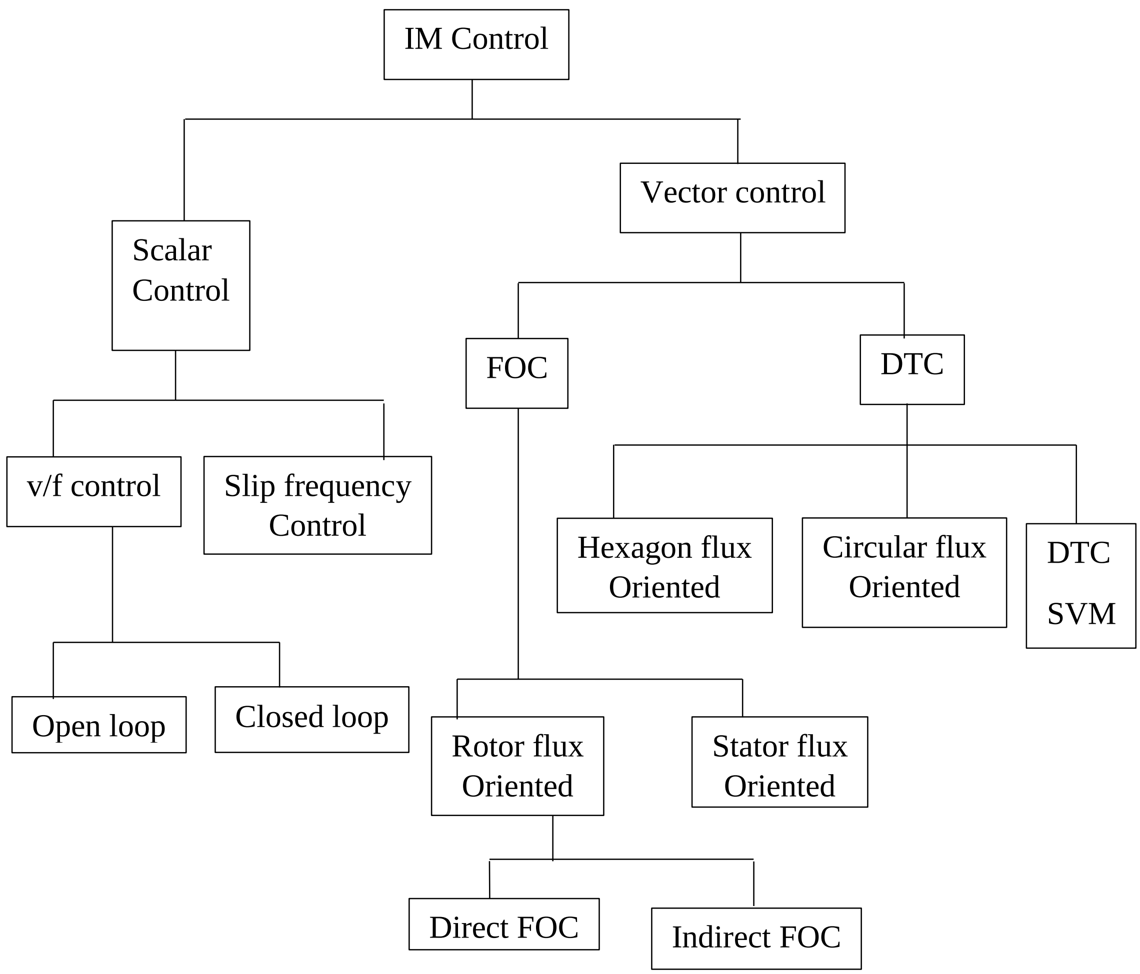


Figure 2. 3 Classification of induction motor control methods

The two superior control techniques are

1. Scalar control: - This method involves controlling the magnitude of voltage or frequency of supply fed to the induction motor. The V/f control method is a control method based on the principle of keeping air gap flux constant by controlling stator voltage and frequency so that the V/f ratio is kept constant. Along with V/f, another control technique is the slip frequency control.

This control method keeps the slip frequency constant and controls the speed by controlling the current and hence the magnitude flux of the motor. In this scheme, the torque command is given to the drive as either a current command or a slip command. The current/slip relationship is then utilized to calculate the slip or current command. The scalar control is somewhat simple to implement, but the inherent coupling effect (i.e. both torque and flux are functions of voltage or current and frequency) gives a sluggish response, and the system is prone to instability because of a high-order system effect [7]. The foregoing problems can be solved by vector or field-oriented control.

2. Vector control or field-oriented control:- here induction motor can be controlled like a separately excited dc motor, brought a renaissance in the high-performance control of ac drives. Vector control is a very widely used control method for high-performance induction motor drive applications and it is applicable to both induction and synchronous motors. The goal of the FOC (also called vector control) on synchronous and the asynchronous machine is to be able to separately control the torque producing and magnetizing flux components [20]. It implies independent decoupled control of flux and torque components of stator current through a coordinated change in supply voltage amplitude, phase, and frequency.

So, as a conclusion high- performance IM drives use vector control (field-oriented control (FOC).

2.5 Introduction to Fractional Order Modeling

2.5.1 Background of Fractional Calculus

The idea of fractional calculus has been known since the regular calculus, with the first reference probably being associated with Leibniz and L'Hospital in 1695 where half order derivative was mentioned [21]. Basically, Fractional calculus deals with fractional integration as well as differentiation.

The mathematical modeling and simulation of systems and processes, based on the description of their properties in terms of fractional derivatives, naturally leads to differential equations of fractional order and to the necessity to solve such equations. Recent findings support the notion that fractional-order calculus should be employed where more accurate modeling and robust control are concerned. Specifically, fractional-order calculus found its way into complex mathematical and physical problems [22].

2.5.2 Definition and Their Laplace Transform of Fractional Order Calculus

The most frequently used definitions are Riemann Liouville's (RL) definition and Grunwald-Letnikov's (GL) definition [23].

Fractional calculus is a generalization of integration and differentiation to non-integer order fundamental operator aD_t^{α} where a and t are the limits of the operation and α is any real number [24]. The continuous integrodifferential operator is defined as [21].

$$aD_t^{\alpha} = \begin{cases} \frac{d^{\alpha}}{dt^{\alpha}}, & \alpha > 0 \\ 1, & \alpha = 0 \\ \int_a^t (d\mathcal{T})^{-\alpha}, & \alpha < 0 \end{cases} \tag{2.3}$$

The Laplace transform is an essential tool in dynamic system and control engineering. In the section below main definitions of fractional order derivatives are presented [21],[24],[22],[25].

1. Riemann-Liouville definition of fractional order derivative

$$aD_t^{\alpha} f(t) = \frac{1}{\Gamma(m-\alpha)} \frac{d^m}{dt^m} \int_a^t (t-\tau)^{m-\alpha-1} f(\tau)\, d\tau \tag{2.4}$$

Where $m - 1 < \alpha \leq m \in N$ and $\alpha \in R$ is a fractional order of differentiation of function f(t) and Γ(.) Represents gamma function.

The Laplace transform of RL fractional order differentiation for the causal system is given as follows

$$\mathscr{L}\left[0D_t^{\alpha} f(t)\right] = S^{\alpha} F(S) - \sum_{k=0}^{n-1} s^k\, 0D_t^{\alpha-k-1} f(0) \tag{2.5}$$

f(0) are the initial conditions, and $n - 1 < \alpha \leq n \in N$.

2. Caputo's definition of fractional order derivative

$$aD_t^{\alpha} f(t) = \frac{1}{\Gamma(m-\alpha)} \int_a^t f^{(m)}(\tau)(t-\tau)^{m-\alpha-1}\, d\tau \tag{2.6}$$

Where $m - 1 < \alpha \leq m \in N$ and $\alpha \in R$ is a fractional order of the differentiation of function f(t).

The Laplace transform of Caputo fractional operator for the causal system is given as

$$\mathscr{L}[0D_t^{\alpha}f(t)]=S^{\alpha}F(S)-\sum_{k=0}^{n-1}s^{\alpha-k-1}f^{(k)}(0) \tag{2.7}$$

Where $n-1<\alpha\leq n\in N$.

$$\Gamma(x)=\int_0^{\infty}t^{x-1}e^{-t}dt$$

$$\Gamma(x)=(x-1)! \tag{2.8}$$

3. Grünwald-Letnikov definition fractional order derivative

$$aD_t^{\alpha}f(t)=\lim_{h\to 0}\frac{1}{h^{\alpha}}\sum_{j=0}^{[\frac{t-a}{h}]}(-1)^j\binom{\alpha}{j}f(t-jh) \tag{2.9}$$

Where [·] means the integer part, h is the step size

Laplace transform of the Grünwald-Letnikov fractional operator for the causal system is given by

$$\mathscr{L}[0D_t^{\alpha}f(t)]=s^{\alpha}F(s) \tag{2.10}$$

$$\binom{a}{j}=\frac{\alpha!}{j!(\alpha-j)!}=\frac{\Gamma(\alpha+1)}{\Gamma(j+1)\Gamma(\alpha-j+1)} \tag{2.11}$$

Rieman-Liouville and Caputo definitions are very close. The only variation is the order of initial conditions and this conditions of integer order which makes them easier to interpret.

2.6 Reference Frame Theory

Based on the reference frame speed, there are possibly four commonly used reference frames that are used to transform a three-phase (ABC) variable into two-dimensional variables [26].

1. **Stationary reference frame:** In this case, the dq axis does not rotate. So, the reference frame speed is zero (ω =0). It is best suited for studying stator variables only. The stator d axis variables are identical to the stator phase A variable.
2. **Rotor reference frame:** The reference frame speed is equal to the rotor speed ($\omega=\omega_r$). Since in this reference frame the d axis of the reference frame is moving at the same relative speed as the rotor phase A winding and coincident with its axis, it is best suited when the analysis is confined to rotor variables as the rotor d axis variable is identical to phase-rotor variables.

3. **Synchronous reference frame:** When the reference frame is rotating at synchronous speed, both the stator and rotor are rotating at different speeds. The reference frame speed is equal to synchronous speed (ω = ω_e). A synchronously rotating reference frame is suitable when an analog computer is employed because both stator and rotor dq quantities become steady DC quantities.
4. **Arbitrary reference frame:** In this reference frame speed is unspecified ω and the dq axis can rotate at an arbitrary speed, there is no relative speed between the four coils d_s, q_s, d_r, q_r.

2.6.1 Vector Transformation

A change of variables that transform three-phase stationery (abc) variables into two-dimensional (dq) variables may be expressed as in equation (2.12) below [2],[26],[27],[28].

$$f_{qdos} = K_s f_{abcs} \tag{2.12}$$

Where

$$f_{abcs}{}^T = f_{as} \quad f_{bs} \quad f_{cs}$$

$$\left[f_{qdos} \right]^T = \left[f_{qs} \quad f_{ds} \quad f_{os} \right]$$

In the above equation, f represents voltage, current, flux, or electric charge.

$$K_s = \frac{2}{3}\begin{bmatrix} \cos(\theta) & \cos(\theta - \frac{2\pi}{3}) & \cos(\theta + \frac{2\pi}{3}) \\ \sin(\theta) & \sin(\theta - \frac{2\pi}{3}) & \sin(\theta + \frac{2\pi}{3}) \\ \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \end{bmatrix}$$

K_s = a vector transformation matrix preserving amplitude.

2.6.2 Reference Frame Transformation

The transformation of the physical variables of an AC machine using reference frame theory could make the analysis easy by transforming the time-varying differential equations to the time-invariant differential equations[29]. The electrical variables such as voltage, current, and flux in a, b, and c phases of a three-phase system can be transformed to the variables in d, q, and n or 0 (direct, quadrature, and neutral) orthogonal axes, where the magnetic couplings between axes are zero. Usually, the d axis, which means the direct axis, is the axis where the main flux directs and the q axis, which means the quadrature axis, lies 90° ahead of the d axis spatially concerning the positive rotational direction of a rotating MMF. Also, the n or 0 axes, which means the neutral axis and sometimes called a zero-sequence axis [29].

The equivalence between three-phase and two-phase is based on the quantity of magnetomotive force produced in two-phase and three-phase windings along with equal current magnitudes. Assuming that each of the three-phase winding has N_s turns per phase and equal currents magnitudes, the two-phase winding will have 3*N_s/2, turns per phase for MMF equality.

The transformations are usually based on the following assumptions:

- Space harmonics of the flux linkage distribution are neglected.
- Rotor flux is assumed to be concentrated across d-axis and zero flux along q-axis.
- Machine core losses are negligible.
- Saturation is neglected.
- Rotor temperature alters the flux, but the variation with time is assumed to be negligible.
- The neutral point is isolated.
- There are no field current dynamics.

The rotor reference frame is chosen because rotor position determines independently the stator voltages & currents, induced EMFs, and torque.

So, to avoid the difficulty in MATLAB implementation, reference frame transformation can be simply employed in two steps; that is

1. abc –to-αβ it is called Clark transformation
2. αβ-to-dq which is called Park transformation

2.6.3 Clark and Park transformations

The Clarke and Park transformation is a transformation of coordinates that designed to transform a multiphase stationary coordinate system into two axes stationary and rotating coordinate system respectively. This transformation is used to reduce the complexity of the system and easier the control method. The stationary two-phase variables of Clarke's transformation are denoted as α and β [26]. As shown in Figure 2.4 the α axis coincides the phase a-axis and the β axis lags the α axis by 90°.

1. Clarke Transform (abc-to-αβ)

Converts balanced three-phase quantities into balanced two-phase time-varying stationary two-dimensional αβ signal.

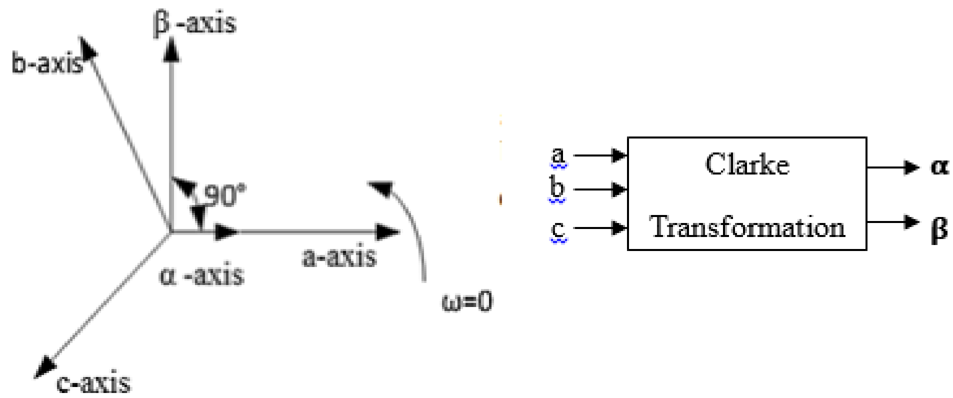


Figure 2. 4 Clark transformation [26]

$$I_a = I * \cos(\omega t) \tag{2.13}$$

$$I_b = I * \cos(\omega t - \frac{2\pi}{3}) \tag{2.14}$$

$$I_b = I * \cos(\omega t - \frac{4\pi}{3}) \tag{2.15}$$

This macro implements the following equations:

$$I_\alpha = I_a \tag{2.16}$$

$$I_\beta = (2I_b + I_a) / \sqrt{3} \tag{2.17}$$

Which result in

$$I_\alpha \approx I * \cos(\omega t) \tag{2.18}$$

$$I_\beta \approx I * \sin(\omega t) \tag{2.19}$$

2. Park Transform (αβ-to-dq)

This transformation converts time-varying vectors in the balanced 2-phase orthogonal stationary system into two dimensional time-invariant orthogonal rotating reference frame.

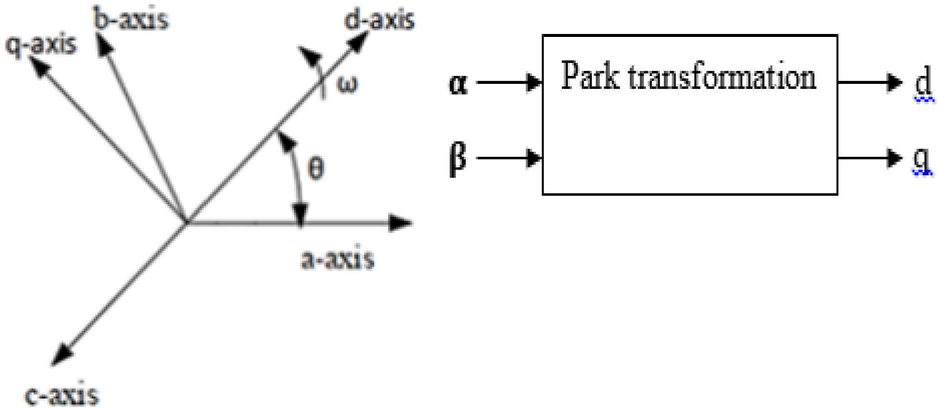


Figure 2. 5 Park transformation[26]

It Implements the following equations:

$$I_D = I_\alpha * \cos(\theta) + I_\beta * \sin(\theta) \tag{2.20}$$

$$I_Q = -I_\alpha * \sin(\theta) + I_\beta * \cos(\theta) \tag{2.21}$$

The instantaneous input quantities are defined by the following equations:

$$I_\alpha = I * \cos(\omega t) \tag{2.22}$$

$$I_\beta = I * \sin(\omega t) \tag{2.23}$$

3. Inverse Park Transform

This transformation projects vectors in orthogonal rotating reference frame into two phase orthogonal stationary frame. It Implements the following equations:

$$I_d = I_D * \cos(\theta) - I_Q * \sin(\theta) \tag{2.24}$$

$$I_q = I_D * \sin(\theta) + I_Q * \cos(\theta) \tag{2.25}$$

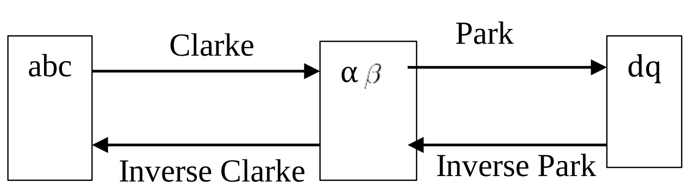


Figure 2. 6 Reference transformation block

The three-phase stator voltages of an induction machine underbalanced condition can be expressed as,

$$V_a = \sqrt{2}V_{rms}\sin(\omega t) \tag{2.26}$$

$$V_b = \sqrt{2}V_{rms}\sin(\omega t - \frac{2\pi}{3}) \tag{2.27}$$

$$V_c = \sqrt{2}V_{rms}\sin(\omega t + \frac{2\pi}{3}) \tag{2.28}$$

Here, V_a, V_b *and* V_c is the three-line voltage and When q - axis is aligning on a - axis, the phase difference between the two axis are equal to zero, i.e. $\theta = 0^0$. The relationship between α**β** and abc (Clarke transformation) is as follows

$$\begin{bmatrix} V_\alpha \\ V_\beta \end{bmatrix} = \frac{2}{3}\begin{bmatrix} 1 & -\frac{1}{2} & -\frac{1}{2} \\ 0 & \frac{\sqrt{3}}{2} & -\frac{\sqrt{3}}{2} \end{bmatrix}\begin{bmatrix} V_a \\ V_b \\ V_c \end{bmatrix} \tag{2.29}$$

So, Relationship between α**β** and dq are:-

$$\begin{bmatrix} V_d \\ V_q \end{bmatrix} = \begin{bmatrix} \cos(\theta) & \sin(\theta) \\ -\sin(\theta) & \cos(\theta) \end{bmatrix}\begin{bmatrix} V_\alpha \\ V_\beta \end{bmatrix} \tag{2.30}$$

The instantaneous values of the stator and rotor currents in the three-phase system are ultimately calculated using the following transformation [30].

$$\begin{bmatrix} i_\alpha \\ i_\beta \end{bmatrix} = \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix}\begin{bmatrix} i_d \\ i_q \end{bmatrix} \tag{2.31}$$

$$\begin{bmatrix} i_a \\ i_b \\ i_c \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ -\frac{1}{2} & -\frac{\sqrt{3}}{2} \\ -\frac{1}{2} & -\frac{\sqrt{3}}{2} \end{bmatrix}\begin{bmatrix} i_\alpha \\ i_\beta \end{bmatrix} \tag{2.32}$$

Where θ is, the angle between the synchronously rotating references frame d axis and the stationary frame α axis. The angular velocity ω and the angular displacement θ of arbitrary reference frame are related as:

$$\omega = \frac{d\theta}{dt} \tag{2.33}$$

Thus,

$$\theta = \int \omega dt \tag{2.34}$$

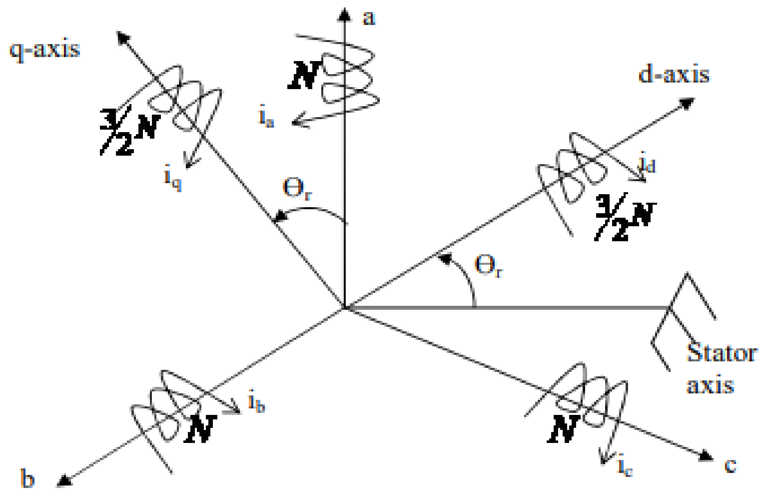


Figure 2. 7 Stationary frame abc to DQ axis transformation [31]

Let the magneto motive force mmf = f = NI

This gives us:

$$f_q = \frac{3*i_q}{2}N = Ni_a\cos(\theta) + Ni_b\cos(\theta - \frac{2\pi}{3}) + Ni_c\cos(\theta + \frac{2\pi}{3}) \tag{2.35}$$

$$f_d = \frac{3*i_d}{2}N = Ni_a\sin(\theta) + Ni_b\sin(\theta - \frac{2\pi}{3}) + Ni_c\sin(\theta + \frac{2\pi}{3}) \tag{2.36}$$

When we removing N from both sides of the above equation we obtain the following relationship:

$$i_q = \frac{2}{3}\left(\cos(\theta)i_a + \cos(\theta - \frac{2\pi}{3})i_b + \cos(\theta + \frac{2\pi}{3})i_c\right) \tag{2.37}$$

$$i_d = \frac{2}{3}\left(\sin(\theta)i_a + \sin(\theta - \frac{2\pi}{3})i_b + \sin(\theta + \frac{2\pi}{3})i_c\right) \tag{2.38}$$

The above equations (2.37) and (2.38) can be rewritten in a matrix form, like as follows:

$$\begin{bmatrix} i_q \\ i_d \end{bmatrix} = \frac{2}{3}\begin{bmatrix} \cos(\theta) & \cos(\theta - \frac{2\pi}{3}) & \cos(\theta + \frac{2\pi}{3}) \\ \sin(\theta) & \sin(\theta - \frac{2\pi}{3}) & \sin(\theta + \frac{2\pi}{3}) \end{bmatrix}\begin{bmatrix} i_a \\ i_b \\ i_c \end{bmatrix} \tag{2.39}$$

Equation (2.39) will be expressed in a compact form by

$$\mathrm{i_{dq}} = \left[\mathrm{T_{abc}}\right]\mathrm{i_{abc}} \tag{2.40}$$

Where

$$[T_{abc}] = \frac{2}{3}\begin{bmatrix} \cos(\theta) & \cos(\theta - \frac{2\pi}{3}) & \cos(\theta + \frac{2\pi}{3}) \\ \sin(\theta) & \sin(\theta - \frac{2\pi}{3}) & \sin(\theta + \frac{2\pi}{3}) \end{bmatrix} \tag{2.41}$$

And transformation from dq to abc or inverse can be expressed as :-

$$i_{qd} = \begin{bmatrix} i_q & i_d \end{bmatrix}^T \tag{2.42}$$

$$i_{abc} = \begin{bmatrix} i_a & i_b & i_c \end{bmatrix}^T \tag{2.43}$$

The compact inverse transformation of the above equation in matrix form as:

$$\begin{bmatrix} i_a \\ i_b \\ i_c \end{bmatrix} = \begin{bmatrix} \cos(\theta) & \sin(\theta) \\ \cos(\theta - \frac{2\pi}{3}) & \sin(\theta - \frac{2\pi}{3}) \\ \cos(\theta + \frac{2\pi}{3}) & \sin(\theta + \frac{2\pi}{3}) \end{bmatrix} \begin{bmatrix} i_q \\ i_d \end{bmatrix} \tag{2.44}$$

Like of the above three phase to two phase compact form two phase to three phase written as:

$$i_{abc} = \left[T_{abc}\right]^{-1} i_{qd} \tag{2.45}$$

Where

$$\left[T_{abc}\right]^{-1} = \begin{bmatrix} \cos(\theta) & \sin(\theta) \\ \cos(\theta - \frac{2\pi}{3}) & \sin(\theta - \frac{2\pi}{3}) \\ \cos(\theta + \frac{2\pi}{3}) & \sin(\theta + \frac{2\pi}{3}) \end{bmatrix} \tag{2.46}$$

When q - axis is aligning on a - axis, the phase difference between the two axes are equal to zero, i.e. $\theta = 0^0$, due to this condition, the relationship between dq and abc is changed to equation (2.47).

$$\begin{bmatrix} i_q^s \\ i_d^s \end{bmatrix} = \frac{2}{3}\begin{bmatrix} 1 & -\frac{1}{2} & -\frac{1}{2} \\ 0 & -\frac{\sqrt{3}}{2} & \frac{\sqrt{3}}{2} \end{bmatrix} \begin{bmatrix} i_a \\ i_a \\ i_a \end{bmatrix} \tag{2.47}$$

Where

$$\left[T_{abc}\right] = \frac{2}{3}\begin{bmatrix} 1 & -\frac{1}{2} & -\frac{1}{2} \\ 0 & -\frac{\sqrt{3}}{2} & \frac{\sqrt{3}}{2} \end{bmatrix} \tag{2.48}$$

Inverse transformation from two phase to three phase looks like as follows:

$$\begin{bmatrix} i_a \\ i_a \\ i_a \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ -\frac{1}{2} & -\frac{\sqrt{3}}{2} \\ -\frac{1}{2} & \frac{\sqrt{3}}{2} \end{bmatrix} \begin{bmatrix} i_q^s \\ i_d^s \end{bmatrix} \tag{2.49}$$

Where

$$\left[T_{abc}\right]^{-1} = \begin{bmatrix} 1 & 0 \\ -\frac{1}{2} & -\frac{\sqrt{3}}{2} \\ -\frac{1}{2} & \frac{\sqrt{3}}{2} \end{bmatrix} \tag{2.50}$$

These transformations are used both for current and voltage transformation. So in a similar manner with voltage transformation equations as follows [30]:

$$\begin{bmatrix} v_a \\ v_b \\ v_c \end{bmatrix} = \begin{bmatrix} \cos(\theta) & \sin(\theta) \\ \cos(\theta - \frac{2\pi}{3}) & \sin(\theta - \frac{2\pi}{3}) \\ \cos(\theta + \frac{2\pi}{3}) & \sin(\theta + \frac{2\pi}{3}) \end{bmatrix} \begin{bmatrix} v_q \\ v_d \end{bmatrix} \tag{2.51}$$

In addition to this, three-phase to two-phase voltage transformation like current transformation looks like:

$$\begin{bmatrix} v_q \\ v_d \end{bmatrix} = \frac{2}{3}\begin{bmatrix} \cos(\theta) & \cos(\theta - \frac{2\pi}{3}) & \cos(\theta + \frac{2\pi}{3}) \\ \sin(\theta) & \sin(\theta - \frac{2\pi}{3}) & \sin(\theta + \frac{2\pi}{3}) \end{bmatrix} \begin{bmatrix} v_a \\ v_b \\ v_c \end{bmatrix} \tag{2.52}$$

CHAPTER THREE

METHODOLOGY

3.1 Introduction

This chapter mainly focuses on the methodology that has been followed in this thesis work. The materials, data collection, analysis method, mathematical modeling of IM, and design of controller and inverter switching techniques for IM drive are explained. In addition to this, FOPID and ANFIS controller techniques for speed control of the motor, field-oriented control algorithm, and space vector modulation are presented.

3.2 Materials

In this thesis work, software such as MATLAB/2019b, Microsoft office 2016, and Math Type 6.0 equation editor is used. MATLAB is the computing software that consists of technical toolboxes and Simulink. Microsoft office is used for writing the thesis documentation. Math Type 6.0 is used for writing mathematical formulas and equations in Microsoft offices word and PowerPoint presentation tools.

3.3 Methods

To attain the general and specific objectives of the study stated in the previous chapter, the following methods and techniques will be followed.

- ❖ Gathering & survey related works from various techniques which are literature, reading books, internet, YouTube, journals, publications, and documents on related areas.
- ❖ Modeling of IM using reference frame transformation.
- ❖ Investigate fractional-order mathematical model of induction motor.
- ❖ Design the space vector modulation.
- ❖ Design GA-based fractional order PID and ANFIS controllers for vector control of induction motor.
- ❖ Performance evaluation and discussion based on MATLAB/Simulink simulation results.
- ❖ Conclude the result and recommend the remaining task for future researchers.

In general, the method followed throughout the research study can be summarized in the following flow chart of Figure 3.1

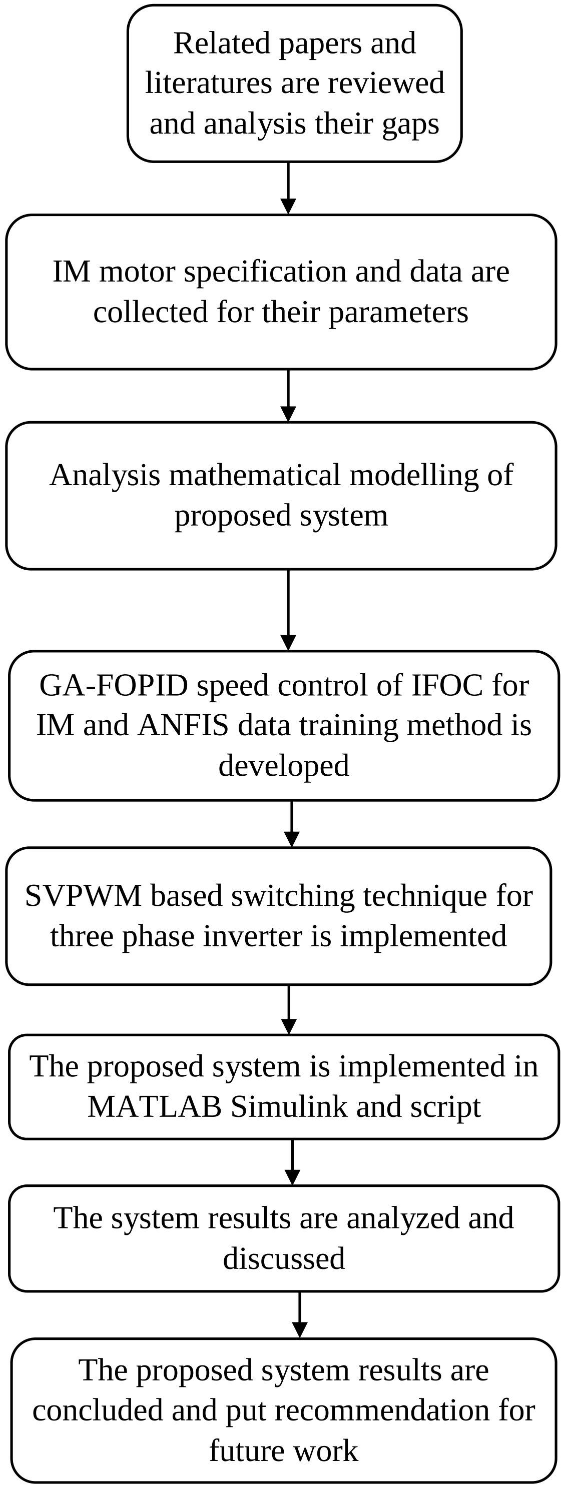


Figure 3. 1 Block diagram of the research methodology

The overall schematic diagram of speed control of indirect vector controlled induction motor is shown below.

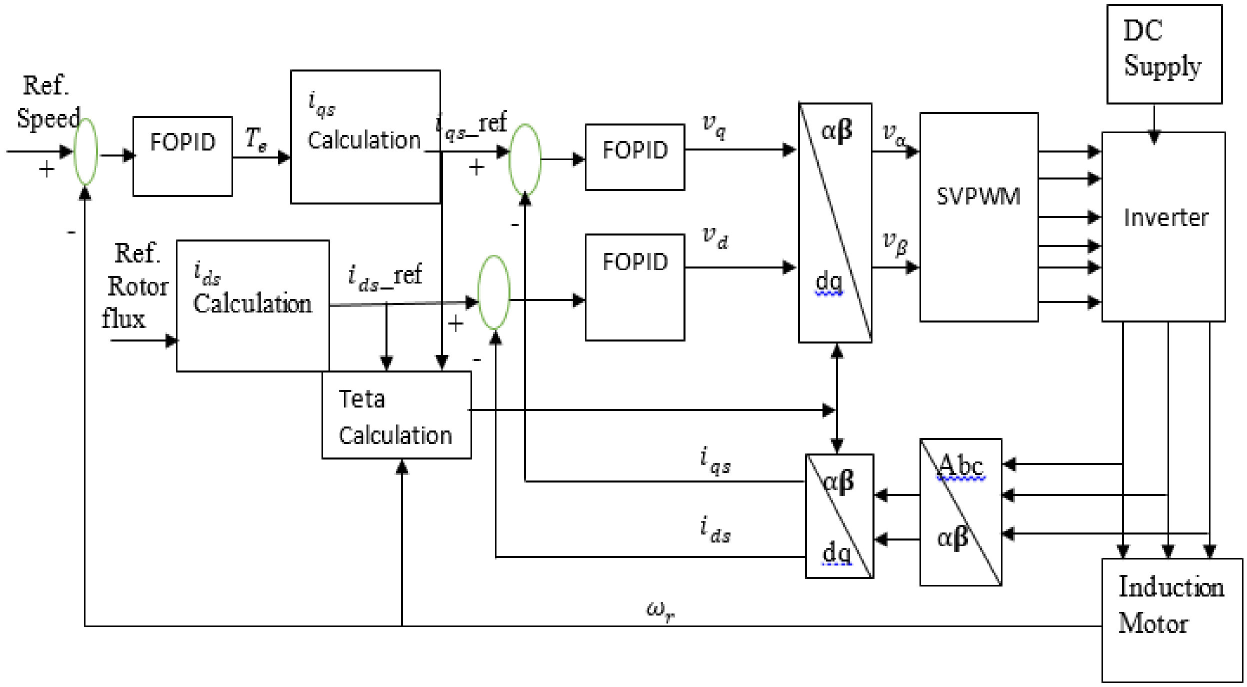


Figure 3. 2 Complete diagrams of indirect vector control

3.4 The Proposed Block Diagram of IM Drive

The design of speed control of vector-controlled induction motor based on trained data through ANFIS and FOPID speed control is shown in Figure 3.3. I saw from the figure, the DC source gives power to the whole system and the three-phase inverter is used to convert the DC voltage to the AC voltage because the load is the three-phase induction motor. The vector control of the induction machine is separately excited in which the torque and flux are controlled independently by using a FOPID controller. The torque and flux commands are regulated by using the FOPID controller into consideration of the current commands to develop the required voltage Commands. These voltages are transformed into a stationary reference frame by using inverse park transformation and fed to SVPWM. The space vector pulse width modulation generates a gate signal to feed into the three-phase inverter. The speed of the induction motor is controlled by using the FOPID controller by comparing the estimated speed of the motor to the set value or reference speed.

3.5 Field Oriented Control Design

Actually vector control algorithm is mainly focuses on two fundamental ideas [2],[26],[28],[32].

1. Separating stator currents to the flux and torque producing currents

An IM can be modeled most easily (and controlled easily) using two quadrature currents rather than the familiar three-phase currents actually applied to the motor. These two currents are called direct (i_d) and quadrature (i_q) are responsible for producing flux and torque respectively in the motor. By definition, the i_q current is in phase with the stator flux, and i_d is at right angles

2. Reference frames theory.

The idea of a reference frame is to transform a quantity that is sinusoidal in one reference frame, to a constant value in another reference frame, which is rotating at the same frequency. Once a sinusoidal quantity is transformed to a constant value by careful choice of reference frame, it becomes possible to control that quantity easily.

The field-orientated control consists of controlling the stator currents represented by a vector. Field-orientated controlled machines need two constants as input references: the torque component (aligned with the q co-ordinate) and the flux component (aligned with d co-ordinate).

Rotor flux field-oriented control also classified into two types [16],[30],[33].

1. Direct field-oriented control, and
2. Indirect field-oriented control

3.5.1 Direct Field Oriented Control

It depends on the generation of unit vector signals from the stator voltage and current signals. The stator flux component can be directly computed from stator quantity. This type of field-oriented control is an optimum choice for medium and high-speed applications.

In the case of direct vector control, the motor flux is measured by directly measure the air gap fluxes using Hall Effect sensors so it is problematic in terms of cost and installation problem [7]. In the direct field-oriented control, the slip frequency of the drive is not controlled. The motor is self-controlled by using the unit vector to control the frequency and phase. The transient response is fast because torque control does not affect flux. As shown in Figure 3.4 the direct vector control method depends on the generation of unit vector signals from the stator or air-gap flux signals. The air-gap signals can be measured directly or estimated from the stator voltage and current signals. The stator flux components can be directly computed from stator quantities. In these systems, the rotor speed is not required for obtaining rotor field angle information [34],[35].

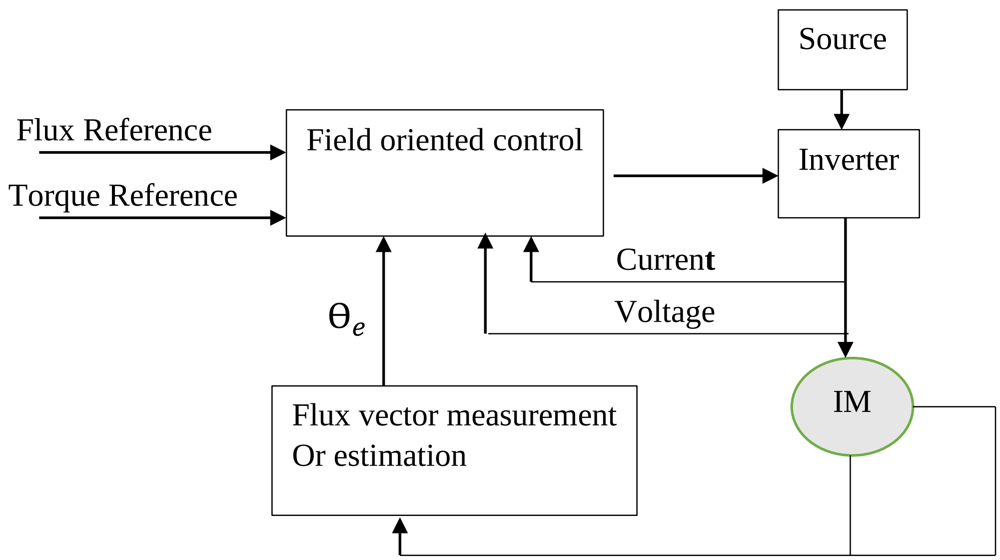


Figure 3. 3 Direct field-oriented control simplified block diagram [36]

Since in this thesis, indirect field-oriented control is used and the above figure is only used for comparison purposes and no application on this thesis work.

3.5.2 Indirect Field Oriented Control

In indirect field-oriented control, the rotor field angle is obtained indirectly by integrating the summation of the rotor speed and slip speed. So, this thesis work mainly focuses on indirect field-oriented control which is shown in figure 3.5 below.

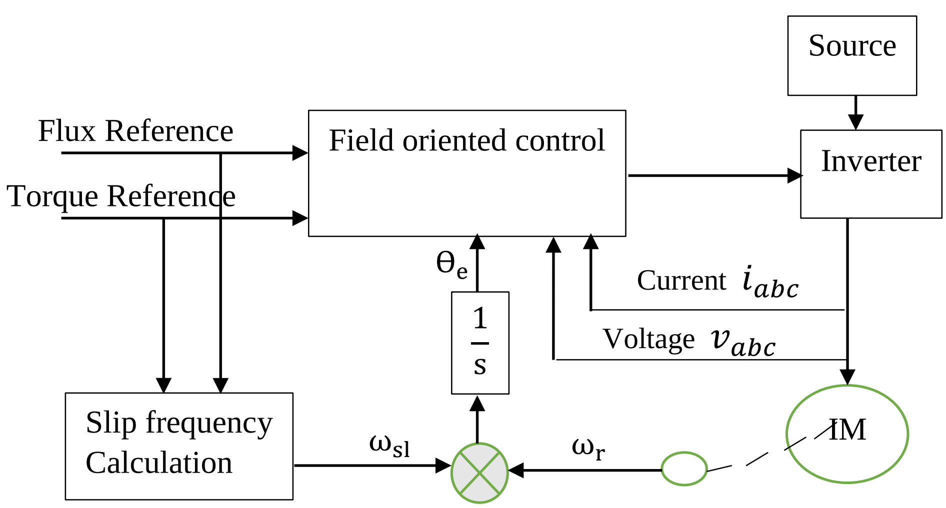


Figure 3. 4 Indirect field-oriented control simplified block diagram [36]

In this thesis work, an indirect field-oriented control is proposed in which the rotor flux position is not directly measured rather it is estimated from stator current.

Reference current calculation for IFOC the quadrature current in the rotor flux oriented synchronous reference frame given by equation (3.1) below.

$$i_{qs}^{*} = \frac{4L_r}{3L_m\lambda_r^{'}} \tag{3.1}$$

Assuming constant rotor flux for fast response time the direct stator current is given by equation (3.2) below.

$$i_{ds}^{*} = \frac{1}{L_m}\lambda_r^{'} \tag{3.2}$$

Where L_m is mutual inductance, L_r is rotor inductance, and λ_r is rotor flux linkage.

Indirect field orientation (IFO) is the difference from the direct flux orientation based on the calculation of the slip speed ω_{sl} [7]. The angular position, Θ_e, of the rotor flux vector is expressed as

$$\theta_{field} = \theta_e = \int \omega_e = \int (\omega_{sl} + \omega_r)dt \tag{3.3}$$

$$\theta_{field} = \int (\frac{L_m R_r}{L_r \lambda_r} i_{qs} + \frac{P}{2}\omega_m)dt \tag{3.4}$$

3.6 Vector Controlled Induction Motor Dynamic Modeling

The voltage equation in two-phase expressed as follows based on figure 3.6 and figure 3.7 [50].

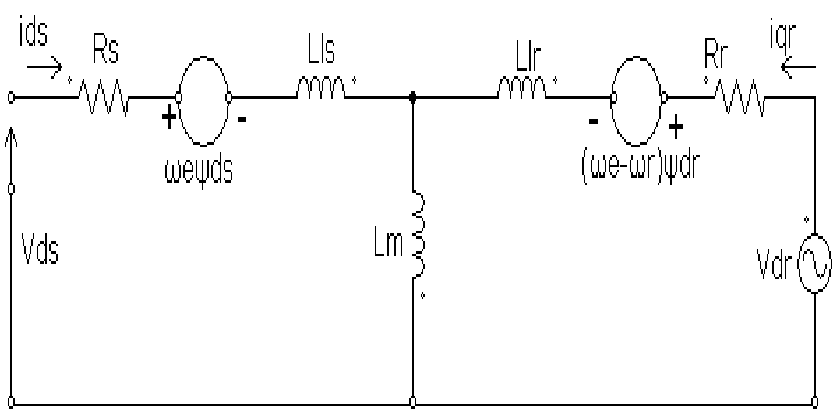


Figure 3. 5 Equivalent circuit of IM in D-Axis

The motor equations can be obtained in a synchronous reference frame by considering all the parameters are in the stator reference frame [37]

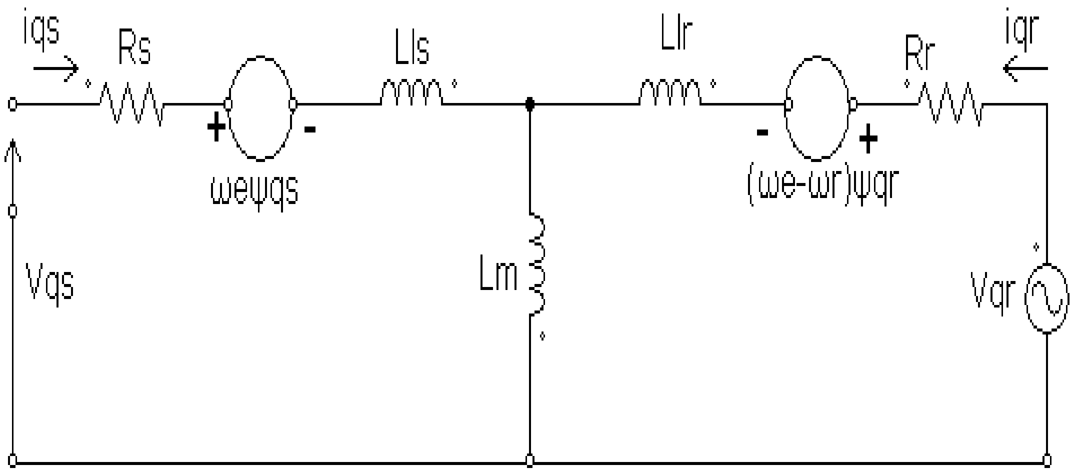


Figure 3. 6 Equivalent circuit of IM in Q-Axis

From Figure 3.5 in the indirect field-oriented control, the rotor flux angle is being measured indirectly, instead of using air-gap flux sensors. IFOC estimates the rotor flux by computing the slip speed (ω_{sl}). In this method, the rotor field angle and thus the unit vectors are indirectly obtained by summation of the rotor speed and slip frequency.

The stator and rotor voltage equations of induction motor in the field orientation synchronous the reference frame can be written as in [38].

$$V_{ds} = R_s i_{ds} - \omega_e \sigma i_{qs} + \frac{d}{dt}(\sigma i_{ds} L_s) - \frac{L_m}{L_r}\frac{d}{dt}[\omega_e \varphi_{qr} - \frac{d}{dt}(\varphi_{dr})] \quad (3.5)$$

$$V_{qs} = R_s i_{qs} - \omega_e \sigma i_{ds} + \frac{d}{dt}(\sigma i_{qs} L_s) + \frac{L_m}{L_r}\frac{d}{dt}[\omega_e \varphi_{dr} - \frac{d}{dt}(\varphi_{qr})] \quad (3.6)$$

$$V_{dr} = R_r i_{dr} + \frac{d}{dt}(\varphi_{dr}) - [\omega_e - \omega_r]\varphi_{qr} = 0 \quad (3.7)$$

$$V_{qr} = R_r i_{qr} + \frac{d}{dt}(\varphi_{qr}) + [\omega_e - \omega_r]\varphi_{dr} = 0 \quad (3.8)$$

Where, $\omega_{sl}= \omega_e$ - ω_r

The rotor flux linkage equations can be written as:

$$\varphi_{dr} = L_r i_{dr} + L_m i_{ds} \quad (3.9)$$

$$\varphi_{qr} = L_r i_{qr} + L_m i_{qs} \quad (3.10)$$

These equations may be rewritten as:

$$i_{dr} = \frac{1}{L_r}\varphi_{dr} - \frac{L_m}{L_r}i_{ds} \tag{3.11}$$

$$i_{qr} = \frac{1}{L_r}\varphi_{qr} - \frac{L_m}{L_r}i_{qs} \tag{3.12}$$

For perfect decoupling control, the total rotor flux needs to be aligned with the d-axis so that the q-axis rotor flux is set zero and d-axis rotor flux is maintained constant then equations written as: φ_{qr} = 0, this implies $\frac{d}{dt}(\varphi_{qr})$ = 0 and $\varphi_{dr} = \varphi_r$ Substitute into the previous equations to implement the indirect vector control strategy[7].

$$\theta_e = \theta_{sl} + \theta_r \tag{3.13}$$

$$\varphi_r = L_m i_{ds} \tag{3.14}$$

$$\frac{R_r}{L_r}\varphi_{qr} - \frac{R_r L_m}{L_r}i_{qs} + \frac{d}{dt}(\varphi_{qr}) + \omega_{sl}\varphi_{dr} = 0 \tag{3.15}$$

The required value of slip speed can be calculated by the following equation

$$\omega_{sl} = \frac{L_m R_r}{\varphi_r L_r}i_{qs}\ or\ \omega_{sl} = \frac{L_m i_{qs}}{\tau_r \varphi_r} \tag{3.16}$$

Where, τ_r is the rotor time constant calculated by: $\tau_r = \frac{L_r}{R_r}$

The quadrature stator current component reference i_{qs}^* is calculated by

$$I_{qs}^* = \left(\frac{2}{3}\right)\left(\frac{2}{P}\right)\left(\frac{L_r}{L_m}\right)\left(\frac{T_e^*}{\varphi_r}\right) \tag{3.17}$$

The direct stator current component reference I_{ds}^* is obtained from reference rotor flux input λ_r^*.

$$I_{ds}^* = \frac{\lambda_r^*}{L_m} \tag{3.18}$$

Where, λ_r^* is reference nominal flux linkage.

3.6.1 Fundamental Equation of AC Motor

For deducing the general equations of the AC induction motor (with “squirrel cage” rotor), let us make the following assumptions:

- Neglect the copper losses and the slots in the machine;

- Spatial distribution of fluxes and amper turns wave are considered sinusoidal
- All the losses due to wiring, saturation, and slot effects can be neglected
- Stator and rotor permeability is assumed to be infinite.

1. Induction machine modelling in the synchronous frame [39],[40],[41];

$$v_{ds}^{e} = R_s i_{ds}^{e} - \omega_e \varphi_{ds}^{e} + \frac{d\varphi_{ds}^{e}}{dt} \tag{3.19}$$

$$v_{qs}^{e} = R_s i_{qs}^{e} - \omega_e \varphi_{qs}^{e} + \frac{d\varphi_{qs}^{e}}{dt} \tag{3.20}$$

Where v_{ds}^{e}, v_{qs}^{e} is stator voltages along d,q axis respectively.

$$v_{dr}^{e} = R_r i_{dr}^{e} - (\omega_e - \omega_r)\varphi_{dr}^{e} + \frac{d\varphi_{dr}^{e}}{dt} \tag{3.21}$$

$$v_{qr}^{e} = R_r i_{qr}^{e} + (\omega_e - \omega_r)\varphi_{qr}^{e} + \frac{d\varphi_{qr}^{e}}{dt} \tag{3.22}$$

Where v_{dr}^{e}, v_{qr}^{e} is stator voltages along with d,q axis respectively.

Stator d^e - q^e axis flux equations

$$\Psi_{ds}^{e} = (L_{ls} + L_m) i_{ds}^{e} + L_m i_{dr}^{e} = L_s i_{ds}^{e} + L_m i_{dr}^{e} \tag{3.23}$$

$$\Psi_{qs}^{e} = (L_{ls} + L_m) i_{qs}^{e} + L_m i_{qr}^{e} = L_s i_{qs}^{e} + L_m i_{qr}^{e} \tag{3.24}$$

Rotor d^e - q^e axis flux equations

$$\Psi_{dr}^{e} = (L_{lr} + L_m) i_{dr}^{e} + L_m i_{ds}^{e} = L_r i_{dr}^{e} + L_m i_{ds}^{e} \tag{3.25}$$

$$\Psi_{qr}^{e} = (L_{lr} + L_m) i_{qr}^{e} + L_m i_{qs}^{e} = L_r i_{qr}^{e} + L_m i_{qs}^{e} \tag{3.26}$$

$$\left.\begin{aligned} \Psi_{ds} &= L_s i_{ds} + L_m i_{dr} \\ \Psi_{qs} &= L_s i_{qs} + L_m i_{qr} \\ \Psi_{dr} &= L_r i_{dr} + L_m i_{ds} \\ \Psi_{qr} &= L_s i_{qr} + L_m i_{qs} \end{aligned}\right\} \tag{3.27}$$

Stator and rotor linkage flux in the synchronous reference frame in Matrix form is given as follows.

$$\begin{bmatrix} \varphi_{ds}^{e} \\ \varphi_{qs}^{e} \\ \varphi_{dr}^{e} \\ \varphi_{qr}^{e} \end{bmatrix} = \begin{bmatrix} L_s & 0 & L_m & 0 \\ 0 & L_s & 0 & L_m \\ L_m & 0 & L_r & 0 \\ 0 & L_m & 0 & L_r \end{bmatrix} \begin{bmatrix} i_{ds}^{e} \\ i_{qs}^{e} \\ i_{dr}^{e} \\ i_{qr}^{e} \end{bmatrix} \tag{3.28}$$

The torque developed by the motor in terms of q-axis current and flux linkage, using the 'rms' convention is

$$T_e = \frac{3}{2}\frac{p}{2}\frac{L_m}{L_r}(\varphi_{rd} i_{sq} - \varphi_{rq} i_{sd}) \tag{3.29}$$

The electromagnetic torque of the mechanical system is given by;

$$T_e = J\frac{dw_r}{dt} + fw_r + T_L \tag{3.30}$$

Where, ω_r is rotor speed in RPM and J-is moment of inertia and f-friction coefficient.

2. Induction machine modeling in the Stationary reference frame [39],[40],[41]. Let us choose first a standstill co-ordinate system ω_e = 0 (where ω_e is synchronous speed), which means that the common coordinate system is a stationary one. Replacing the value ω_e = 0 in the general Equations of the motor given by (3.19), (3.20), (3.21), and (3.22), the following vector voltage equations conclude:

$$V_{ds}^{s} = R_s i_{ds}^{s} + \frac{d\varphi_{ds}^{s}}{dt} \tag{3.31}$$

$$V_{qs}^{s} = R_s i_{qs}^{s} + \frac{d\varphi_{qs}^{s}}{dt} \tag{3.32}$$

$$V_{dr}^{s} = R_s i_{dr}^{s} + \frac{d\varphi_{dr}^{s}}{dt} + w_r \varphi_{qr}^{s} = 0 \tag{3.33}$$

$$V_{qr}^{s} = R_s i_{qr}^{s} + \frac{d\varphi_{qr}^{s}}{dt} - w_r \varphi_{dr}^{s} = 0 \tag{3.34}$$

The stator and rotor voltage equations and the electromagnetic torque equations can be represented in matrix form as given below separately;

$$\begin{bmatrix} \varphi_{ds} \\ \varphi_{qs} \end{bmatrix} = \begin{bmatrix} L_s & 0 \\ 0 & L_s \end{bmatrix} \begin{bmatrix} i_{ds} \\ i_{qs} \end{bmatrix} + \begin{bmatrix} L_m & 0 \\ 0 & L_m \end{bmatrix} \begin{bmatrix} i_{dr} \\ i_{qr} \end{bmatrix} \tag{3.35}$$

$$\begin{bmatrix}\varphi_{dr}\\ \varphi_{qr}\end{bmatrix}=\begin{bmatrix}L_r & 0\\ 0 & L_r\end{bmatrix}\begin{bmatrix}i_{dr}\\ i_{qr}\end{bmatrix}+\begin{bmatrix}L_m & 0\\ 0 & L_m\end{bmatrix}\begin{bmatrix}i_{ds}\\ i_{qs}\end{bmatrix} \tag{3.36}$$

So, neglecting the magnetic saturation of the circuit and iron losses, the flux linkage Equations (3.19) and (3.20) of the circuit can be represented in matrix form as given below separately: -

$$\begin{bmatrix}v_{ds}\\ v_{qs}\end{bmatrix}=\begin{bmatrix}R_s & 0\\ 0 & R_s\end{bmatrix}\begin{bmatrix}i_{ds}\\ i_{qs}\end{bmatrix}+\frac{d}{dt}\begin{bmatrix}\varphi_{ds}\\ \varphi_{qs}\end{bmatrix}+\begin{bmatrix}0 & -w_e\\ w_e & 0\end{bmatrix}\begin{bmatrix}\varphi_{ds}\\ \varphi_{qs}\end{bmatrix} \tag{3.37}$$

$$\begin{bmatrix}v_{dr}\\ v_{qr}\end{bmatrix}=\begin{bmatrix}R_r & 0\\ 0 & R_r\end{bmatrix}\begin{bmatrix}i_{dr}\\ i_{qr}\end{bmatrix}+\frac{d}{dt}\begin{bmatrix}\varphi_{dr}\\ \varphi_{qr}\end{bmatrix}+\begin{bmatrix}0 & -(w_e-w_r)\\ (w_e-w_r) & 0\end{bmatrix}\begin{bmatrix}\varphi_{dr}\\ \varphi_{qr}\end{bmatrix} \tag{3.38}$$

$$T_e=\frac{3}{2}\frac{P}{2}\frac{L_m}{L_r}\begin{bmatrix}\varphi_{dr} & \varphi_{qr}\end{bmatrix}\begin{bmatrix}i_{ds}\\ -i_{qs}\end{bmatrix} \tag{3.39}$$

Solving Equation (3.36) for the rotor currents $i_{dr}^{s}$and i_{qr}^{s} putting these values in Equation (3.35), as follows;

$$\begin{bmatrix}\varphi_{ds}\\ \varphi_{qs}\end{bmatrix}=\begin{bmatrix}\sigma L_s & 0\\ 0 & \sigma L_s\end{bmatrix}\begin{bmatrix}i_{ds}\\ i_{qs}\end{bmatrix}+\begin{bmatrix}\frac{L_m}{L_r} & 0\\ 0 & \frac{L_m}{L_r}\end{bmatrix}\begin{bmatrix}\varphi_{dr}\\ \varphi_{qr}\end{bmatrix} \tag{3.40}$$

Where the total leakage factor σ is defined as $\sigma=1-\dfrac{L_m^2}{L_sL_r}$

Putting the values of rotor d- and q-axis voltage as zero and the values of rotor d and q axis currents from Equation (3.36) in (3.37), as follows;

$$\frac{d}{dt}\begin{bmatrix}\varphi_{dr}\\ \varphi_{qr}\end{bmatrix}=\frac{L_m}{T_r}\begin{bmatrix}1 & 0\\ 0 & 1\end{bmatrix}\begin{bmatrix}i_{ds}\\ i_{qs}\end{bmatrix}-\frac{1}{T_r}\begin{bmatrix}1 & 0\\ 0 & 1\end{bmatrix}\begin{bmatrix}\varphi_{dr}\\ \varphi_{qr}\end{bmatrix}+(\omega_e-\omega_r)\begin{bmatrix}0 & 1\\ -1 & 0\end{bmatrix}\begin{bmatrix}\varphi_{dr}\\ \varphi_{qr}\end{bmatrix} \tag{3.41}$$

Again, substituting, the values of stator fluxes from Equation (3.40) in (3.37), as follows;

$$\frac{d}{dt}\begin{bmatrix}i_{ds}\\ i_{qs}\end{bmatrix}=\begin{bmatrix}-R_1 & w_e\\ w_e & -R_1\end{bmatrix}\begin{bmatrix}i_{ds}\\ i_{qs}\end{bmatrix}+\begin{bmatrix}R_2 & R_3w_r\\ -R_3w_r & R_2\end{bmatrix}\begin{bmatrix}\varphi_{dr}\\ \varphi_{qr}\end{bmatrix}+\begin{bmatrix}\frac{1}{\sigma L_s} & 0\\ 0 & \frac{1}{\sigma L_s}\end{bmatrix}\begin{bmatrix}v_{ds}\\ v_{qs}\end{bmatrix} \tag{3.42}$$

So, the mathematical model or the state-space model of the induction motor in terms of stator currents and rotor flux linkages as the state variables are obtained by combining Equation (3.41) and (3.42), as follows;

$$\frac{d}{dt}\begin{bmatrix} i_{ds} \\ i_{qs} \\ \varphi_{dr} \\ \varphi_{qr} \end{bmatrix} = \begin{bmatrix} -R_1 & w_e & R_2 & R_3 w_r \\ -w_e & -R_1 & R_3 w_r & R_2 \\ R_5 & 0 & -R_4 & w_{sl} \\ 0 & R_5 & -w_{sl} & -R_4 \end{bmatrix}\begin{bmatrix} i_{ds} \\ i_{qs} \\ \varphi_{dr} \\ \varphi_{qr} \end{bmatrix}\begin{bmatrix} \frac{1}{\sigma L_s} & 0 \\ 0 & \frac{1}{\sigma L_s} \\ 0 & 0 \\ 0 & 0 \end{bmatrix}\begin{bmatrix} v_{ds} \\ v_{qs} \end{bmatrix} \tag{3.43}$$

Where,

$$R_1 = \frac{1}{\sigma L_s}(R_s + R_r \frac{L_m^2}{L_r^2}), R_2 = \frac{1}{\sigma L_s}\frac{R_r L_m}{L_r^2}, R_3 = \frac{1}{\sigma L_s}\frac{L_m}{L_r}, R_4 = \frac{1}{T_r}$$

$$R_5 = \frac{R_r L_m}{L_r}, w_{sl} = w_e - w_e = slip\,speed.$$

3.7 Space Vector Pulse Width Modulation

3.7.1 Classification of Invertors

Depending on the kind of DC source supplying the inverter, they will be classified as voltage source inverters (VSI) or current source inverters (CSI). Typically, of the three-phase bridge configuration, with DC-link connected between the rectifier and the inverter. The DC link could be a simple inductive, capacitive, or inductive-capacitive low-pass filter since neither the voltage across a capacitor nor this through an inductor can change instantaneously. A capacitive-output DC link is employed for a VSI and an inductive output link is utilized in CSI. Three-phase inverters, supplying voltages and currents of adjustable frequency and magnitude to the stator, are a very important element of adjustable speed drive systems employing IM. Inverters with semiconductor power switches are DC to AC power converters [7].

3.7.2 Principle of Space Vector PWM

PWM has six switches (1- 6) and each of these is represented with an IGBT switching device as in figure 3.8. A, B, and C represent the output for the phase-shifted sinusoidal signals. Depending on the switching combination the inverter produces different outputs, creating the two-level signal.

The biggest difference from other pulse width modulation methods is that the SVPWM uses a vector as a reference. This gives the advantage of a better overview of the system.

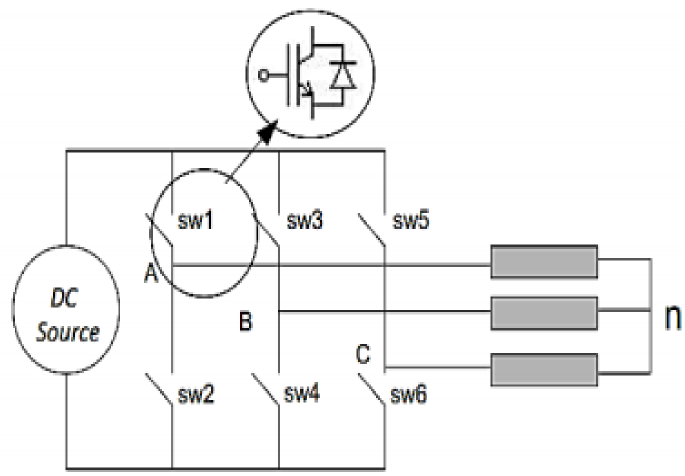


Figure 3.8 Three-phase inverters, with a load and neutral point [42]

The eight switching vectors of the three upper power switches, output line to neutral voltage (phase voltage), and output line-to-line voltages in terms of DC-link V_{dc}, are given in Table 3.1.

Table 3. 1 Switching vectors, phase voltages, and output line to line voltages

Voltage Vectors	Switching Vectors			Line to neutral Voltage			Line to line Voltage			V_α	V_β
	A	B	C	V_{an}	V_{bn}	V_{cn}	V_{an}	V_{bn}	V_{cn}		
V_0	0	0	0	0	0	0	0	0	0	0	0
V_1	1	0	0	2/3	-1/3	-1/3	1	0	-1	2/3	0
V_2	1	1	0	1/3	-1/3	-1/3	1	0	-1	1/3	$1/\sqrt{3}$
V_3	0	1	0	-1/3	2/3	-1/3	-1	1	0	-1/3	$1/\sqrt{3}$
V_4	0	1	1	-2/3	1/3	-1/3	-1	0	1	-2/3	0
V_5	0	0	1	-1/3	-1/3	2/3	0	-1	1	-1/3	$-1/\sqrt{3}$
V_6	1	0	1	1/3	-2/3	1/3	1	-1	0	1/3	$-1/\sqrt{3}$
V_7	1	1	1	0	0	0	0	0	0	0	0

Principle of space vector PWM

- ❖ Treats the sinusoidal voltage (reference voltage) as a constant amplitude vector rotating at a constant frequency.

- This PWM technique approximates the reference voltage V_{ref} by a combination of the eight switching patterns (V_0 to V_7).
- Coordinate transformation (dq reference frame to the stationary α-β frame).
- The vectors (V_1 to V_6) divide the plane into six sectors (each sector: 60 degrees).
- V_{ref} is generated by two adjacent non-zero vectors and two zero vectors.

Table 3.2 shows that the switches can be ON or OFF, meaning 1 or 0. switching state '1' denotes that the upper switch in an inverter leg is on and the inverter terminal voltage (V_{an}, V_{bn}, V_{cn}) is positive ($+V_{DC}$) while '0' indicates that the inverter terminal voltage is zero due to the conduction of the lower switch in the inverter leg [43].

Table 3.2 Switching states for each phase leg

Switching States	A			B			C		
	S_1	S_2	V_{an}	S_3	S_4	V_{bn}	S_5	S_6	V_{cn}
0	ON	OFF	V_{DC}	ON	OFF	V_{DC}	ON	OFF	V_{DC}
1	OFF	ON	0	OFF	ON	0	OFF	ON	0

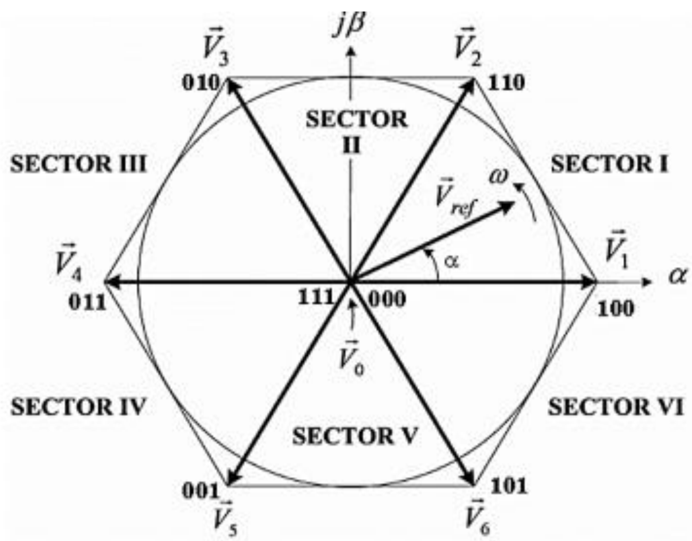


Figure 3.9 Space voltage vectors in different sectors [L1]

There are eight possible combinations of switching states in the two-level inverter as listed in Table 3.3. The switching state of [100] corresponds to the conduction of S_1, S_6, and S_2 in the inverter legs A, B, and C, respectively.

Among the eight switching states, [111] and [000] are active states and the others are zero state.

Table 3.3 Space vectors, switching states, and on-state switches

Space vector		Switching state (three phases)	On-state switch	Vector definition
Zero vector	V_0	[111]	S_1, S_3, S_5	0
		[000]	$S_4, S_{6,} S_2$	
Active vector	V_1	[100]	$S_{1,}\ S_6, S_2$	$V_1 = \frac{2}{3}V_d e^{j0}$
	V_2	[110]	S_1, S_3, S_2	$V_1 = \frac{2}{3}V_d e^{j\frac{\pi}{3}}$
	V_3	[010]	S_4, S_3, S_2	$V_1 = \frac{2}{3}V_d e^{j\frac{2\pi}{3}}$
	V_4	[011]	$S_4, S_{3,} S_5$	$V_1 = \frac{2}{3}V_d e^{j\frac{3\pi}{3}}$
	V_5	[001]	$S_{4,}\ S_{6,} S_5$	$V_1 = \frac{2}{3}V_d e^{j\frac{4\pi}{3}}$
	V_6	[101]	S_1, S_6, S_5	$V_1 = \frac{2}{3}V_d e^{j\frac{5\pi}{3}}$

Complex vector expression for these eight vectors can be given as:

$$V_k = \begin{cases} \frac{2}{3}V_{dc}e^{\frac{j(k-1)\pi}{3}} & if\ k = 1,2,3,4,5,6 \\ 0 & if\ k = 0,7 \end{cases} \tag{3.44}$$

Therefore, space vector PWM can be implemented by the following steps:

Step 1: Determine V_α, V_β, V_{ref}, and angle (α)

Using the coordinate transformation to a 2-Φ stationary reference frame, the V_α, V_β, V_{ref}, and angle (α) can be determined as follows:

$$V_\alpha = V_{an} - V_{bn}.\cos(60) - V_{cn}.\cos(60)$$

$$= V_{an} - \frac{1}{2}V_{bn} - \frac{1}{2}V_{cn} \tag{3.45}$$

$$V_\beta = 0 + V_{bn}.\cos(30) - V_{cn}.\cos(30)$$

$$= \frac{\sqrt{3}}{2}V_{bn} - \frac{\sqrt{3}}{2}V_{cn} \tag{3.46}$$

Therefore, the above equations can be summarized in matrix form as follows.

$$\begin{bmatrix} V_\alpha \\ V_\beta \end{bmatrix} = \begin{bmatrix} 1 & -\frac{1}{2} & -\frac{1}{2} \\ 0 & \frac{\sqrt{3}}{2} & -\frac{\sqrt{3}}{2} \end{bmatrix} \begin{bmatrix} V_{an} \\ V_{bn} \\ V_{cn} \end{bmatrix} \tag{3.47}$$

The reference space vector voltage crossing every sector is derived as:

$$\left|\bar{V}_{ref}\right| = \sqrt{V_\alpha^2 + V_\beta^2} \tag{3.48}$$

The angle in which the reference voltage vector is found is determined by equation (3.49).

$$\alpha = \tan^{-1}\left(\frac{V_\beta}{V_\alpha}\right) = \omega t = 2\pi ft \tag{3.49}$$

Where,

f is the fundamental frequency at which the reference voltage rotates.

Here the need for α is that to transform from one sector to another sector, this transformation must depend on the value of α.

From the above step 1 concluded that to implement the space vector PWM, the voltage equations in the abc reference frame can be transformed into the stationary dq reference frame that consists of the horizontal d-axis and vertical q-axis. So to do this apply to park and Clarke transformations in which three-phase parameters can be easily converted to comfortable two-phase parameters. These all are used to develop gate signals for the space vector pulse width modulation techniques.

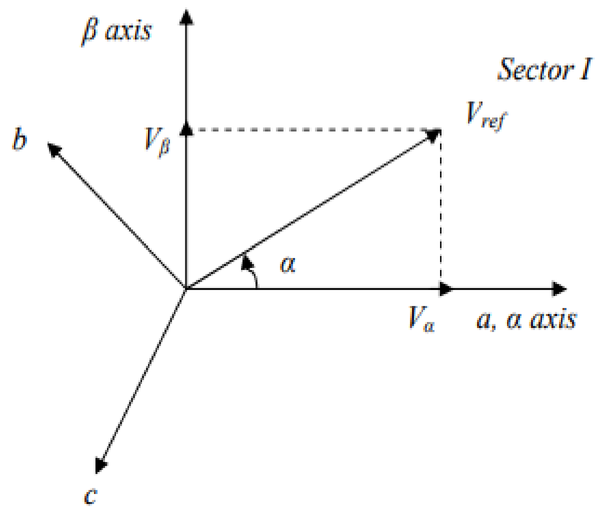


Figure 3.10 Voltage space vector and its components in (abc axis) [44]

$$V_\alpha = V_d \cos(\theta) - V_q \sin(\theta) \tag{3.50}$$

$$V_\beta = V_d \sin(\theta) + V_q \cos(\theta) \tag{3.51}$$

Having calculated V_α, V_β and V_{ref} and the reference angle in this step is taken. The next step is to calculate the duration time for each vector V_1-V_6.

Step 2: Determine time duration T_1, T_2, T_0

From figure 3.10, the switching time duration can be calculated as follows: Switching time duration at Sector 1:

$$\int_0^{Tz} \overline{V}_{ref}\, dt = \int_0^{T_1} \overline{V}_1 dt + \int_{T_1}^{T_1+T_2} \overline{V}_2 dt + \int_{T_1+T_2}^{Tz} \overline{V}_0 dt \tag{3.52}$$

$$\therefore Tz.\overline{V}_{ref} = (T_1.\overline{V}_1 + T_2.\overline{V}_2) \tag{3.53}$$

The average voltage for the first sector which is made by vectors V_0, V_1, V_2 and V_7 is given by equation (3.54). (Where, $0 \le \alpha \le 60$)

$$\Rightarrow T_z.\left|\overline{V}_{ref}\right|.\begin{bmatrix}\cos(\alpha)\\ \sin(\alpha)\end{bmatrix} = T_1.\frac{2}{3}.V_{dc}.\begin{bmatrix}1\\ 0\end{bmatrix} + T_2.\frac{2}{3}.V_{dc}.\begin{bmatrix}\cos(\frac{\pi}{3})\\ \sin(\frac{\pi}{3})\end{bmatrix} \tag{3.54}$$

$$\therefore T_1 = T_z.a.\frac{\sin(\frac{\pi}{3}-\alpha)}{\sin(\frac{\pi}{3})} \tag{3.55}$$

$$\therefore T_2 = T_z.a.\frac{\sin(\alpha)}{\sin(\frac{\pi}{3})} \tag{3.56}$$

$$T_0 = T_z - (T_1 + T_2) \tag{3.57}$$

Where,

$$T_z = \frac{1}{f_z} \, and \; a = \frac{\left|\bar{V}_{ref}\right|}{\frac{2}{3}V_{dc}}$$

T_1 and T_2 are the switching time durations of vectors V_1 and V_2 respectively.

T_0 is the time duration of the zero vector.

T_z is the time period for which one sector is applied.

Switching time duration at any sector is given by the following equations:

$$\begin{aligned} T_n &= \frac{\sqrt{3}.T_z.\left|\bar{V}_{ref}\right|}{V_{dc}}\left(\sin\left(\frac{\pi}{3}-\alpha+\frac{n-1}{3}\pi\right)\right) \\ &= \frac{\sqrt{3}.T_z.\left|\bar{V}_{ref}\right|}{V_{dc}}.\sin\left(\frac{n}{3}\pi-\alpha\right) \\ &= \frac{\sqrt{3}.T_z.\left|\bar{V}_{ref}\right|}{V_{dc}}\left(\sin\frac{n}{3}\pi.\cos\alpha-\cos\frac{n}{3}\pi.\sin\alpha\right) \end{aligned} \tag{3.58}$$

$$\begin{aligned} T_{n+1} &= \frac{\sqrt{3}.T_z.\left|\bar{V}_{ref}\right|}{V_{dc}}\left(\sin\left(\alpha-\frac{n-1}{3}\pi\right)\right) \\ &= \frac{\sqrt{3}.T_z.\left|\bar{V}_{ref}\right|}{V_{dc}}\left(-\cos\alpha.\sin\frac{n-1}{3}\pi+\sin\alpha.\cos\frac{n-1}{3}\pi\right) \end{aligned} \tag{3.59}$$

$$T_0 = T_z - (T_n + T_{n+1}) \tag{3.60}$$

Where,

n = 1 through 6(sector I to VI) and $0 \leq \alpha \leq 60$

The method used to approximate the desired stator reference voltage with only eight possible states of switches is to combine adjacent vectors of the reference voltage and to determine the time of application of each adjacent vector as shown in figure 3.11 for the first sector.

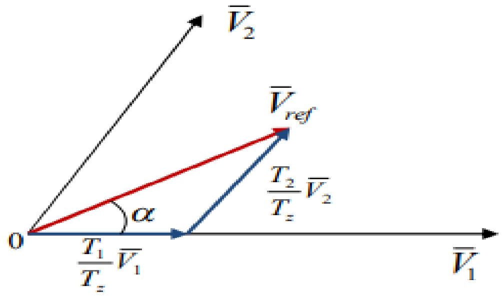


Figure 3.11 Reference voltage as a combination of adjacent vectors in sector I [38]

Step 2.1: Description of Duration Times for All Sectors

The general calculation to receive the duty times in the rest of the sectors is given by

$$T_1 = T_c * \alpha * \sin\left(\frac{\pi}{3} - \theta + \frac{n-1}{3} * \pi\right) = T_c * \alpha * \left[\sin\left(\frac{n}{3}\pi\right)\cos(\theta) - \cos(\frac{n}{3}\pi)\sin(\theta)\right] \quad (3.61)$$

$$T_2 = T_c * \alpha * \sin\left(\theta - \frac{n-1}{3} * \pi\right) = T_c * \alpha * \left[-\cos(\theta)\sin\left(\frac{n-1}{3}\pi\right) + \sin(\theta)\cos\left(\frac{n-1}{3}\pi\right)\right] \quad (3.62)$$

$$T_0 = T_c - T_1 - T_2 \quad (3.63)$$

Table 3.4 Duration time for each sector [43]

Sector	Duration times		
	T_1	T_2	T_0
1	$T_c * \alpha * \sin\left(\frac{\pi}{3} - \theta\right)$	$T_c * \alpha * \sin(\theta)$	$T_c - T_1 - T_2$
2	$T_c * \alpha * \sin\left(\frac{2\pi}{3} - \theta\right)$	$T_c * \alpha * \sin\left(\theta - \frac{\pi}{3}\right)$	$T_c - T_1 - T_2$
3	$T_c * \alpha * \sin(\pi - \theta)$	$T_c * \alpha * \sin\left(\theta - \frac{2\pi}{3}\right)$	$T_c - T_1 - T_2$
4	$T_c * \alpha * \sin\left(\frac{4\pi}{3} - \theta\right)$	$T_c * \alpha * \sin(\theta - \pi)$	$T_c - T_1 - T_2$
5	$T_c * \alpha * \sin\left(\frac{5\pi}{3} - \theta\right)$	$T_c * \alpha * \sin\left(\theta - \frac{4\pi}{3}\right)$	$T_c - T_1 - T_2$
6	$T_c * \alpha * \sin(2\pi - \theta)$	$T_c * \alpha * \sin\left(\theta - \frac{5\pi}{3}\right)$	$T_c - T_1 - T_2$

Step 3: Description of Switching Time

The switching time in each sector is summarized in the table below:

Table 3.5 Duty time for each sector [L1]

Sector	Upper switches	Lower switches
1	$S_1 = T_1 + T_2 + T_0/2$ $S_3 = T_2 + T_0/2$ $S_5 = T_0/2$	$S_4 = T_0/2$ $S_6 = T_1 + T_0/2$ $S_2 = T_1 + T_2 + T_0/2$
2	$S_1 = T_1 + T_0/2$ $S_3 = T_1 + T_2 + T_0/2$ $S_5 = T_0/2$	$S_4 = T_2 + T_0/2$ $S_6 = T_0/2$ $S_2 = T_1 + T_2 + T_0/2$
3	$S_1 = T_0/2$ $S_3 = T_1 + T_2 + T_0/2$ $S_5 = T_2 + T_0/2$	$S_4 = T_1 + T_2 + T_0/2$ $S_6 = T_0/2$ $S_2 = T_1 + T_0/2$
4	$S_1 = T_0/2$ $S_3 = T_1 + T_0/2$ $S_5 = T_1 + T_2 + T_0/2$	$S_4 = T_1 + T_2 + T_0/2$ $S_6 = T_2 + T_0/2$ $S_2 = T_0/2$
5	$S_1 = T_2 + T_0/2$ $S_3 = T_0/2$ $S_5 = T_1 + T_2 + T_0/2$	$S_4 = T_1 + T_0/2$ $S_6 = T_1 + T_2 + T_0/2$ $S_2 = T_0/2$
6	$S_1 = T_1 + T_2 + T_0/2$ $S_3 = T_0/2$ $S_5 = T_1 + T_0/2$	$S_4 = T_0/2$ $S_6 = T_1 + T_2 + T_0/2$ $S_2 = T_1 + T_0/2$

3.8 The Proposed System Block Diagram

The design of speed control of indirect field-oriented control of induction motor based on training data through ANFIS and FOPID speed control is shown in Figure 3.12. From the figure, speed error is given as input to ANFIS in which error is trained again and again based on the criteria RMSE value until getting minimum error and which is acts as gain and coefficient parameters for FOPID controller. The DC source gives power to the whole system and the three-phase inverter is used to convert the DC voltage to the AC voltage because the load is the three-phase induction motor. Load here is the torque which is acts as a disturbance to the system. The vector control of the induction machine is separately excited in which the torque and flux are controlled independently by using a FOPID controller. These voltages are transformed into a stationary reference frame by using inverse park transformation and fed to SVPWM. The space vector pulse width modulation generates a gate signal to feed into the three-phase inverter.

The speed of the IM is controlled by using the FOPID controller by comparing the estimated speed of the motor to the set value or reference speed.

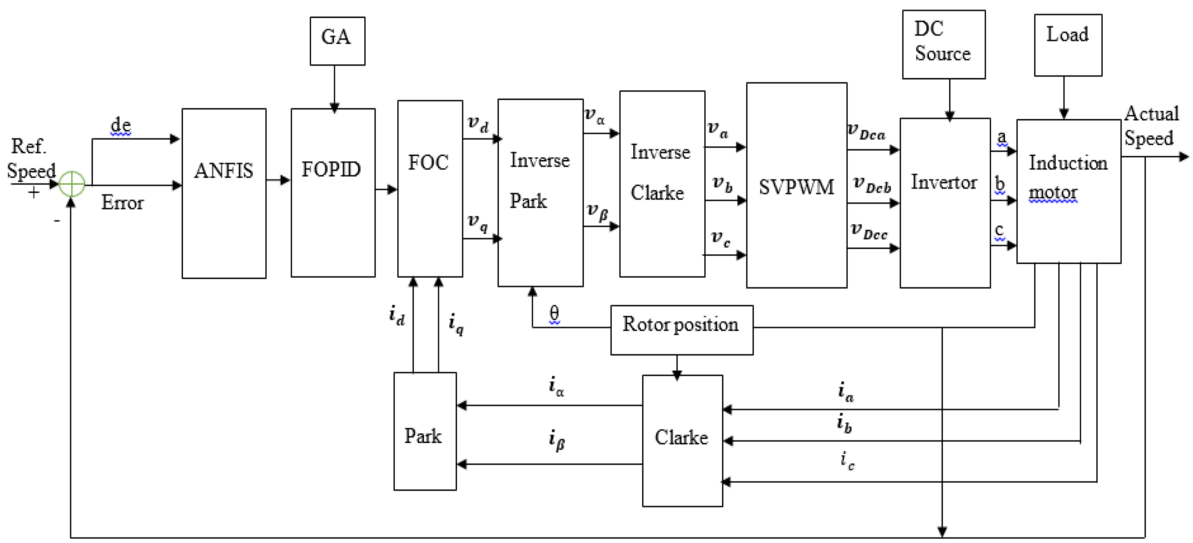


Figure 3.12 Block diagram of the proposed drive control system

3.9 Genetic Algorithm Optimization Technique

The most popular evolutionary algorithm for optimization is the Genetic Algorithm (GA), invented by John Holland at the University of Michigan in 1975 [45].

The basic element of GA is the chromosome. This contains the generic information for a given solution and is typically coded as a binary string. Given an optimization problem, the Genetic

Algorithm encodes the parameter designed into a finite bit string, and then runs iteratively randomly using the three operators but based on the fitness function evolution. Finally, the Genetic Algorithm finds and decodes the solution to the problem from the last pool of mature strings. Initially; a population of chromosomes created randomly represents several solutions to a given problem. A fitness function, which is in effect a performance index, is used to select the best solutions in the population to be parents to the offspring that will comprise the next generation. The fitter the parent, the greater the probability of selection. This emulates the evolutionary process of "survival of the fittest" [46].

The Genetic Algorithm Process Architecture is shown in flowchart Figure 3.13. As shown in the figure the process is started by taking a random population set from the total population and iteratively perform the calculation of fitness function or objective function until finding the best solution.

This step is repeated up to total iteration and when the iteration is finished it can display both best fit value as well as the best values for the five parameters to be optimized which are gains of PID controller in addition to the order of integrator and derivatives for FOPID. The fitness function is based on the finding of integral absolute time error which must be minimum as possible. So, the genetic algorithm uses three main types of rules at each step to create the next generation from the current population. This algorithm is repeated for many generations and finally stops when reaching individuals that represent the optimum solution in which error approximately equals the value of error tolerance.

- Selection rules select the individuals, called parents, which contribute to the population at the next generation.
- Crossover rules combine two parents to form children for the next generation.
- Mutation rules apply random changes to individual parents to form children.

To evaluate control performance, the fitness function (J) is based on the Integral Time Absolute Error (ITAE) criterion which has the advantage of providing lesser overshoot along with the less settling time [15]. Since the objective is to minimize the error between the desired output and the actual output.

$$J = \int_0^T t\,|e(t)|\,dt \qquad (3.64)$$

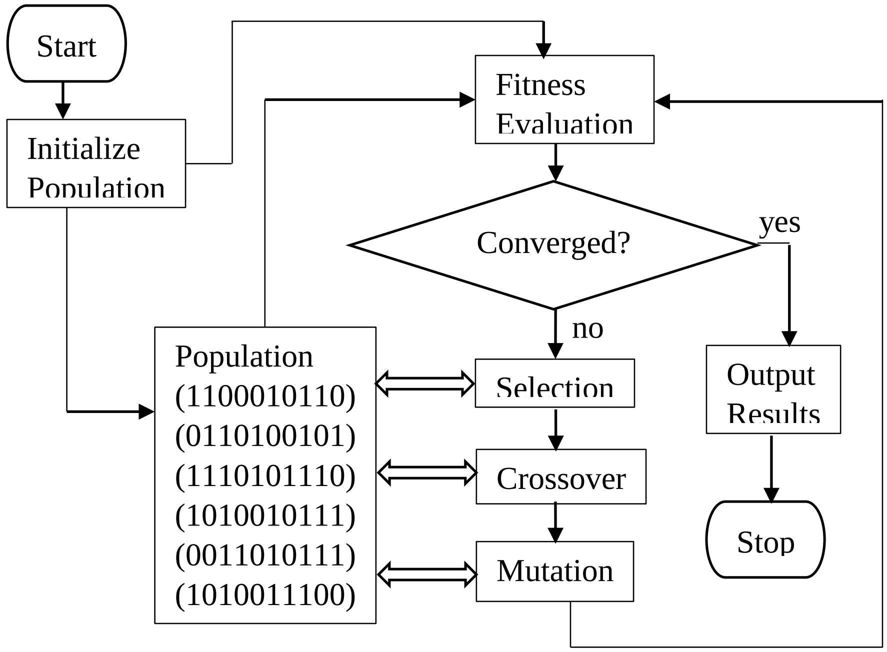


Figure 3.13 Genetic Algorithm Process Architecture

To implement the GA optimization first, the fitness function must be defined. The fitness function is the objective function that needs to be minimized.

In this thesis, the objective function contains the function that represents the integral time error between desired and actual output. Besides the fitness function, the number of variables to be optimized should also be specified. The variables to be optimized are PID speed controller's gains (k_p, k_i, k_d) and λ& μ in case of fractional order controller to ensure optimal control performance for the induction motor.

The block diagram for the entire system is given below.

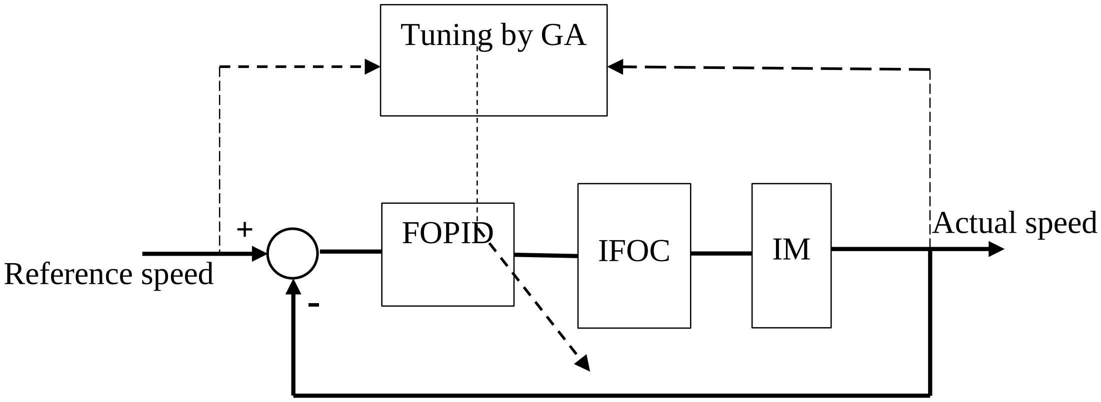


Figure 3.14 Structure of FOPID controller optimization using GA technique

Here the broken line on FOPID shows that the fitness function is calculated to get the best values for the parameters of FOPID. FOPID controllers are optimized by the Genetic Algorithm function ‘ga’ of MATLAB 2019b. The control systems are Simulink models. To call the Simulink model from MATLAB, the Simulink model must first be built, and then MATLAB programming functions are used to call the Simulink models. Parameters of the fractional-order PID controller consist of five variables, namely, kp1 =x1, ki1=x2, lambda1=x3, kd1=x4, and mu1 =x5 which are set in the ‘fractional PID Controller’ block as shown in figure 3.15 MATLAB codes for genetic algorithm optimization is written in appendix B.

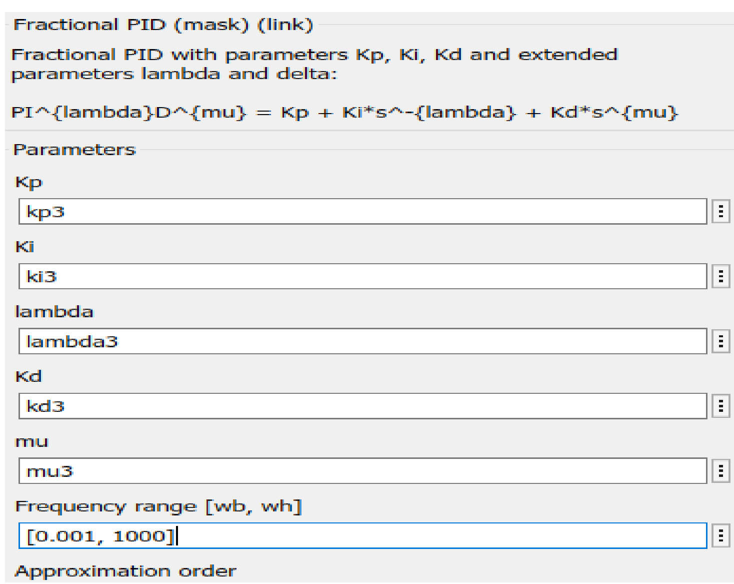


Figure 3.15 Configuration for parameters of FOPID controller

3.10 Controller Design

3.10.1 Conventional PID

A proportional–integral–derivative controller (PID controller or three-term controller) is a control loop mechanism employing feedback that is widely used in industrial control systems and a variety of other applications requiring continuously modulated control. A PID controller continuously calculates an error value[47].

A PID controller is a feedback control system, which calculates an error value as the difference between the reference value and a measured variable and applies a correction based on proportional, integral, and derivative terms continuously. The controller attempts to minimize the error signal by adjustment of a control variable to a new value determined by a weighted sum. The output of a PID controller, which is equal to the control input to the plant, is calculated in the S-domain from the feedback error as in equation (3.66)[48].

Increasing the proportional gain (K_p) will reduce the rise time but never eliminates the steady-state error. Introducing the integral gain (K_i) will help to reduce the steady-state error, but it makes the system very sluggish (and oscillatory), thereby making the transient response very poor. The effect of adding a derivative gain (K_d) is an increase in the stability of the system, reduction in the overshoot, and improvement in the transient response (but no effect on the steady-state error). The general effect of each controller parameter (K_p, K_i, K_d) independently on a closed-loop feedback system has been summarized in Table 3.6[48].

PID controller is today probably the most widespread type of controller in the industry. The PID controller has the advantage that the physical effect of each of its three gain terms is clearly visualized in the features of the transient response: that is the effect of the three gains $K_p, K_i, and\ K_d$is known. That is K_p is responsible for overshoot, K_i is responsible for the speed of response and steady-state error, and K_d is responsible for the damping effect. PID control is widely used in systems that employ output feedback control. In such systems, the controlled output is measured and fed back to a summing point where it is subtracted from the reference input. The difference between the reference and feedback corresponds to the control loop error (or servo error) and forms the input to the PID controller. The PID controller output is the parallel sum of three paths error, error integral, and error derivative multiplied by their corresponding gains. The user adjusts the relative weights of each path to optimize transient response performance[49].

The control output for a PID control algorithm is given by equation (3.65).

$$u(t) = K_p e(t) + K_i \int_0^t e(\tau) d\tau + K_d \frac{de(t)}{dt} \tag{3.65}$$

Taking Laplace transform of the equation in 3.65 yields the system in s-domain as follows

$$U(S) = E(S)\left(K_p + \frac{K_i}{S} + K_d S \right) \tag{3.66}$$

Lastly the transfer function of the PID controller is

$$G(S) = \frac{U(S)}{E(S)} = K_p + \frac{K_i}{S} + K_d S \tag{3.67}$$

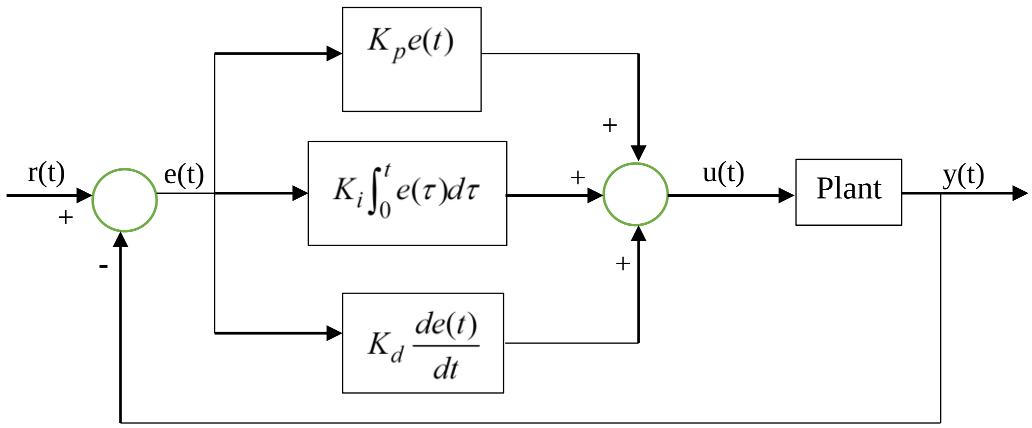


Figure 3.16 PID control structure

Table 3.6 effects of PID gains on transient state performances

Gains	Rise time	Overshoot	Settling time	Steady-state error
K_p	Decrease	Increase	Small change	Decrease
K_i	Decrease	Increase	Increase	Decrease
K_d	Small change	Decrease	Decrease	No change

Most of the previous work focuses on applying PI controller in control application area but based on the contributions that are explained in table 3.6 above this thesis work focuses on applying fractional order PID controller rather than using fractional-order PI controller to control seed of vector controlled three-phase induction motor.

Means derivative gain have their own task in the control area called minimization of the overshoot as well as settling times which acceptable strategies from the control perspective.

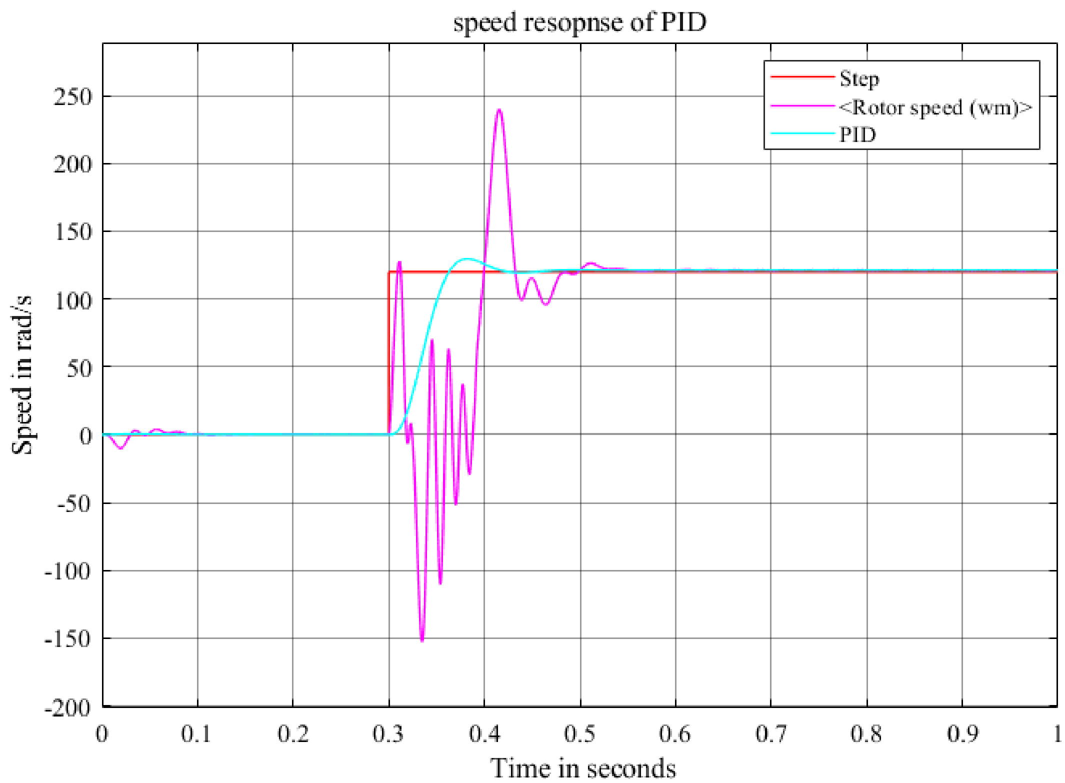


Figure 3.17 Speed response of PID for constant input at no load

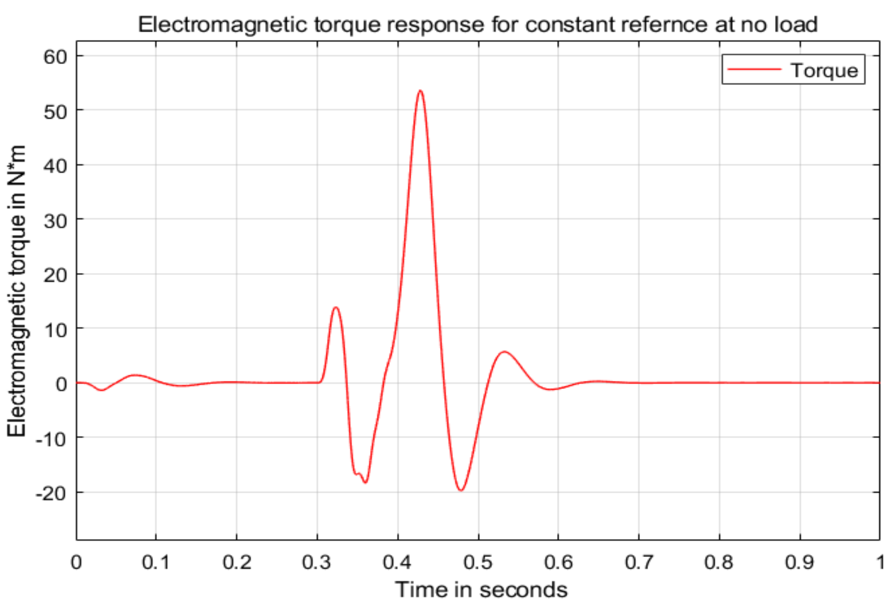


Figure 3.18 Electromagnetic torque response

Let us analyze figure 3.17 and figure 3.18

1. As we can see from the figure rotor speed has a small steady-state error but initially, it has high oscillation which affects motor life.

2. It has also a high maximum overshoot, high settling time which is not an accepted condition in the control law.

So based on this drawback conventional PID is not good to control the speed of indirect field oriented control of three-phase induction motor and applying GA-based PID control in which the parameters of PID controller is updated until getting the best solution so that the speed of induction motor mimic to set point.

3.10.2 GA-based PID

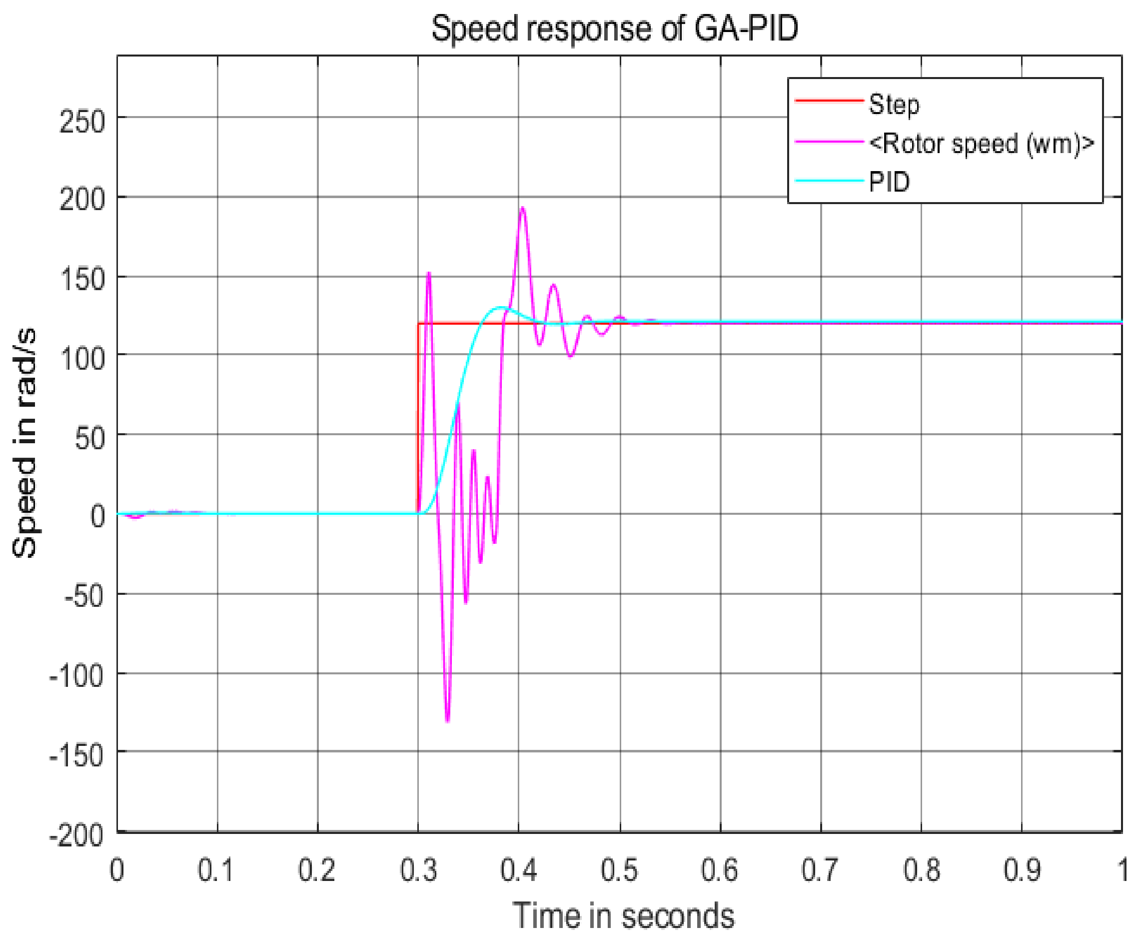


Figure 3.19 Speed response of GA-PID for constant reference at no load

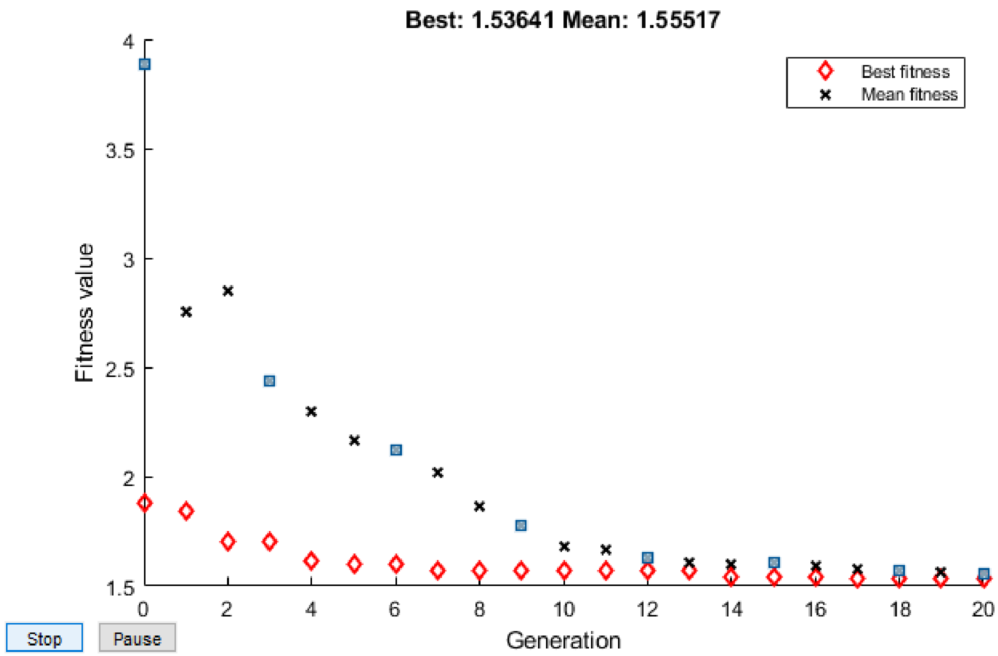


Figure 3.20 Best evaluated outputs of fitness function of PID controller

Table 3.7 Effects of gains on the system performance

Before GA		After GA	Fval
Kp	1	0.4737	1.53641
Ki	1	0.001	
Kd	1	0.2657	

I concluded from the above figures that it violates basic principles as follows

1. There is two oscillation level after applying 120 rad/s at a t=0.3se .this results in more vibration on the motor.

2. As we know IM motors have high starting torque and it must come back to zero immediately after overshoot. But here the value after overshoot is 1.99 N*m which is different from zero.

3.10.3 Review of FOPID Controller

Fractional Order Control (FOC) means controlled systems and/or controllers described by fractional-order differential equations [50]. Recently, there are increasing interest to enhance the performance of ordinary PID controllers by using the concept of fractional calculus, where the orders of derivatives and integrals are non-integer. The idea of the fractional-order controller was first proposed by A. Oustaloup through Commande Robuste d'Ordre Non-Entier (CRONE) controller in 1991 [51]. Later on, Igor Podlubny had initiated the most common form of fractional order PID in the form of $PI^{\lambda}D^{\mu}$ in 1999 involving an integrator of order λ and differentiator of order μ, where the values of λ and μ lie between 0 and 1 [52]. Clearly, depending on the values of the orders λ and μ, numerous choices for the controller's type can be made. He also demonstrated that the response of this type of controller is better as compared to the classical PID controller. One of the most important advantages of the FOPID controller is the better control of dynamical systems, which are described by fractional-order mathematical models. Another advantage lies in the fact that the FOPID controllers are less sensitive to changes of parameters of a controlled system. This is due to the two extra degrees of freedom to better adjust the dynamical properties of a fractional-order control system. However, up till now, there is no systematic way to set the value for λ and μ [53],[54]. The fractional integrodifferential equation of the FOPID controller is given by [52],[54]:

$$u(t) = K_p e(t) + K_I D_t^{-\lambda} e(t) + K_D D_t^{\mu} e(t) \tag{3.68}$$

The transfer function of the FOPID controller is obtained through Laplace transform as follows:

$$G_{FOPID}(S) = \frac{U(S)}{E(S)} = K_P + \frac{K_I}{S^{\lambda}} + K_D S^{\mu} \tag{3.69}$$

Where E(s) is an error and U(s) is the controller's output. It is obvious that the fractional order PID controller not only needs to design three parametersK_p, K_i , and K_d, but also design two orders λ and μ of integral and derivative controllers. Figure 3.21 shows the block diagram configuration of the FOPID controller.

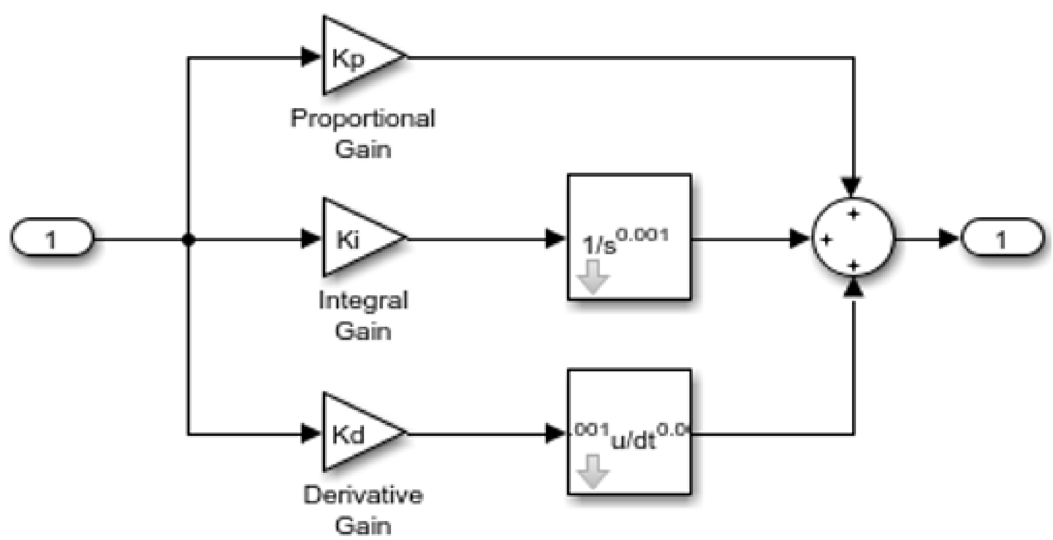


Figure 3.21 Simulink block diagram of the FOPID controller

In Equation (3.69), S^{λ} and S^{μ} have fractional orders which are not directly compatible with the MATLAB toolbox and it becomes difficult to realize the hardware of the FOPID controller. In

MATLAB fractional order PID controller is implemented using FOMCOM toolbox [55]. As shown in Figure 3.22, the fractional-order PID controller generalizes the ordinary integer-order PID controller and expands it from point to plane. Point (0, 0) corresponds to P controller, point (0, 1) corresponds to ordinary PD controller, point (1, 0) corresponds to an ordinary PI controller, and point (1, 1) corresponds to ordinary PID controller, whereas the shaded portion between four corners represents the FOPID controllers. All these classical types of PID controllers are special cases of the FOPID controller when the values of λ and μ are integer values of 0 or 1 [52],[54].

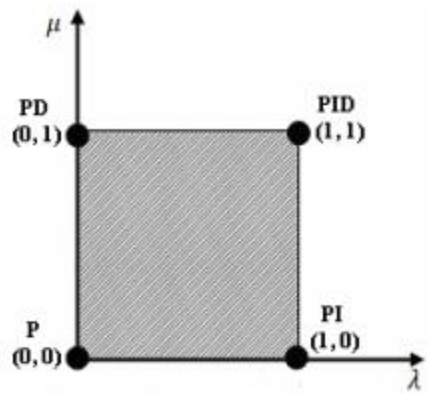


Figure 3.22 Pictorial representation of the fractional-order PID controller [52]

The FOPID control loop is represented by the block diagram as shown in Figure 3.23, where Gp(s) and Gc(s) are the plant and the FOPID controller models, respectively. The FOPID receives the error signal E(s) and produces the control signal U(s) to control the output signal C(s) and regulate the disturbance signal D(s), referring to the reference input R(s).

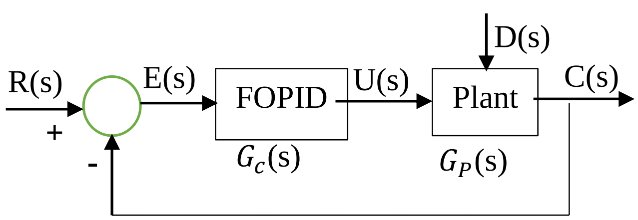


Figure 3.23 FOPID control loop

In the present work, a new method is proposed to find the optimum control parameters λ, K_i , K_p , K_d and μ. These parameters are computed such that the controlled system exhibits some desired time-domain performances. There are several performance criteria that can be used for this purpose. Some of these are integral of absolute error (IAE), integral of squared-error (ISE), or integral of time-weighted-squared-error (ITSE) [56].

$$IAE = \int_0^\infty \left| y_{ref}(t) - y(t) \right| dt = \int_0^\infty \left| e(t) \right| dt \tag{3.70}$$

$$ISE = \int_0^\infty e^2(t)\, dt \tag{3.71}$$

$$ITSE = \int_0^\infty te^2(t)\, dt \tag{3.72}$$

$$ITAE = \int_0^\infty te(t)\, dt \tag{3.73}$$

Where e(t) is the error between the desired profile (y_{ref}f (t)) and the actual one (y(t)). Although the ITSE performance criterion can overcome the largest time it takes to perform the entire task, it cannot ensure a desirable stability margin [55]. Among the listed fitness function above, in this thesis work, ITAE is used due simple it results in small overshoot and oscillation in which minimized error between actual and reference speed is obtained.

Table 3.8 Parameters for GA optimization

GA property	Value or Description
Population size	30
Number of generation	20
Selection function	Tournament selection
Crossover function	Intermediate
Mutation function	Uniform mutation
Cross over probability	0.6
Mutation probability	0.1
Initial range	Upper : UB= [5 5 5 1 1] Lower :LB= [0.1 0.1 0.1 0.1 0.1]
Tolerance	10^{-5}

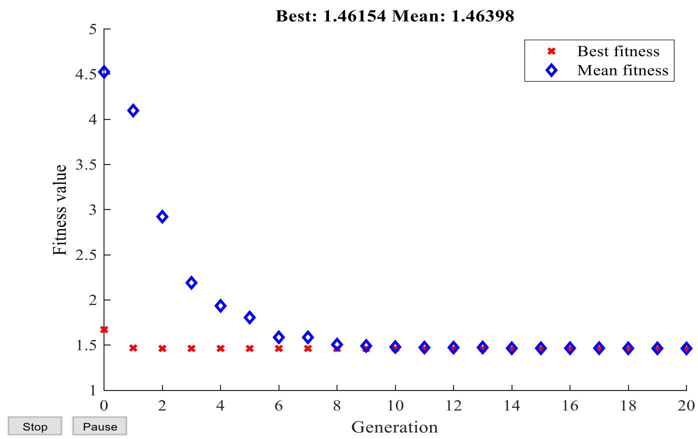


Figure 3.24 Best evaluated outputs of fitness function of FOPID controller

Here best value is 1.46154 but I need the value around 10^{-5} based on the error tolerance to minimize speed error to zero.

The parameters of FOPID speed controllers obtained according to the procedure of optimization by the technique of GA are given below in table 3.9.

Table 3.9 FOPID controller initial parameters

Parameters	Before GA	After GA	Fval
k_p	0.05	0.001	1.46154
k_i	5	1.8339	
k_d	1	0.8622	
λ	0.5	0.5377	
μ	0.5	0.3188	

3.10.4 Adaptive Neuro-Fuzzy Inference System

ANFIS has the learning ability of neural networks to realize the fuzzy logic inference system, are gained popularity in the control of nonlinear systems [57]. Some architectures for neuro-fuzzy controllers are proposed for example by Jang [58], Abraham [59], and so on. In this section, the basics of adaptive networks, called ANFIS, are introduced in detail and it works as a fuzzy controller [60]. Neuro-fuzzy control combines the mapping and learning ability of an artificial neural network with the linguistic and fuzzy inference advantages of fuzzy logic that can self-modify their membership function to achieve the desired performance [61]. Thus, a neuro-fuzzy controller has the potential to outperform conventional ANN or fuzzy logic controllers.

Three types of inference systems.

1: Tsukamoto Fuzzy Inference System

In this type overall output is calculated as the weighted average of each rule's crisp output, which is represented by the rule's firing strength and output membership functions. The rules firing strength can be determined by the product or minimum of the output of input membership functions. The output membership functions used must be monotonically non-decreasing [62].

2: Mamdani Fuzzy Inference System

These type standard fuzzy sets are used in both the antecedents and consequents of the rules, and a defuzzification procedure is utilized to obtain a crisp value from the output. The overall fuzzy output is calculated by applying the "max" function to the fuzzy outputs of each rule. For final crisp output several approaches might be used like: center of the area, the bisector of area, mean of maxima, maximum criterion, etc. [63].

3: Sugeno Fuzzy Inference System

In this type, if-then rules are used [64]. The output of each rule is a linear combination of input variables plus a constant term, and the final output is the weighted average of each rule's output.

ANFIS controller employs a Tagaki Sugno Kang (TSK) fuzzy inference system [65]. The basic ANFIS architecture is shown in figure 3.25. Square nodes in the ANFIS structure denote parameter sets of the membership functions of the TSK fuzzy system. Circular nodes are static (non-modifiable) and perform operations such as product or max/min calculations. A hybrid learning rule is used to accelerate parameter adaptation. This uses sequential least-squares in the forward pass to identify consequent parameters, and backward pass to establish the premise parameters [65].

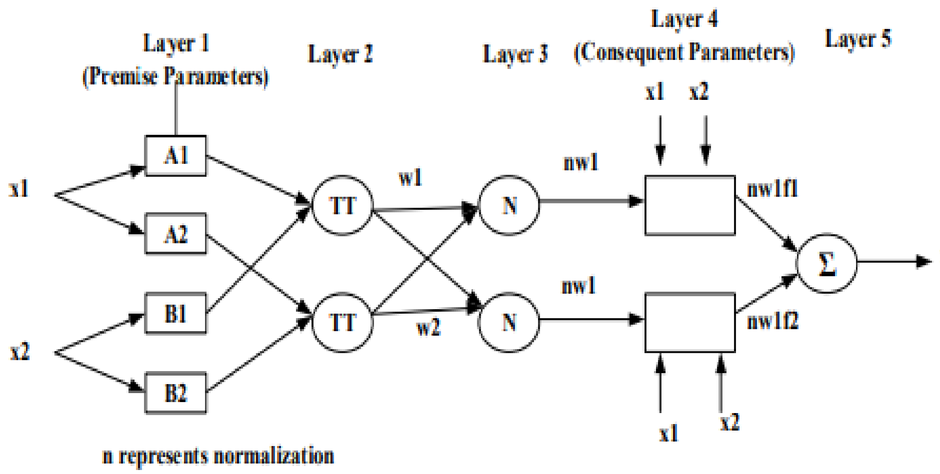


Figure 3. 25 ANFIS structure [65]

3.10.4.1 Basic Principles of ANFIS

Figure 3.25 shows the ANFIS structure and it is composed of five functional blocks (rule base, database, a decision-making unit, a fuzzification interface, and a defuzzification interface) which are generated using five network layers [57]:

Layer 1: This layer is composed of several computing nodes whose activation functions are fuzzy logic membership functions.

Layer 2: This layer chooses the minimum value of the inputs.

Layer 3: This layer normalizes each input to the others (The i^{th}node output is the ith input divided by the sum of all the other inputs).

Layer 4: This layer's i^{th} node output is a linear function of the third layer's i^{th} node output and the ANFIS input signals.

Layer 5: This layer sums all the incoming signals. The ANFIS structure can be tuned automatically by a least square estimation (for output membership functions) and a backpropagation algorithm (for output and input membership functions) [66].

If the fuzzy inference system has inputs x_1 and x_2 and output f as shown in figure 3.25 then first order TSK rule base becomes

Rule 1: if x_1 is A_1 and x_2 is B_1 then $f_1= p_1x_1+q_1x_2+r_1$

Rule 2: if x_1 is A_2 and x_2 is B_2 then $f_2= p_2x_1+q_2x_2+r_2$

For n membership functions Rule n: if x_1 is A_n and x_2 is B_n then $f_n= p_nx_1+q_nx_2+r_n$

Where

$A_1 \ldots A_N$, $B_1 \ldots B_N$ membership functions and $p_1 \ldots p_n$, $q_1 \ldots q_n$ and $r_1 \ldots r_n$ are constants within the consequent functions. The membership type can be any type of membership function which are available in the fuzzy logic toolbox. For this thesis, the typical membership function is the generalized triangular membership function.

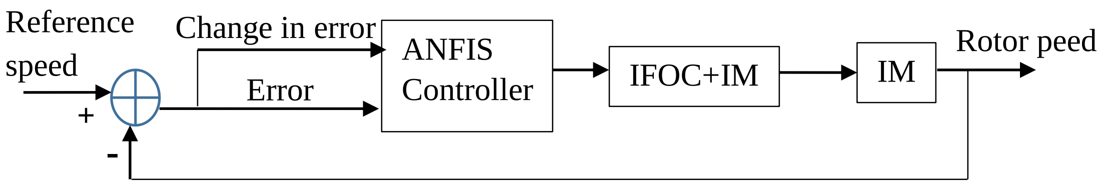


Figure 3.26 ANFIS control scheme for IFOIM

By using limited input-output training data pairs, ANFIS tunes the membership functions and other associated parameters by backpropagation gradient descent and the least square type method. Such methodologies make the ANFIS modeling more systematic and less reliant on expert knowledge. Every node in layer one and layer four have an adaptive node with updated parameters to achieve a desired input-output mapping. These parameters are updated according to given training data and gradient-based learning procedure while the nodes in layers two and three are fixed with no parameters [67].

In the learning process of ANFIS, the premise parameters (on layer one) and consequent parameters (on layer four) should be tuned to optimum mathematical relation between inputs and outputs. ANFIS uses a two-pass learning cycle. In the forward pass, the algorithm uses the least-squares method to identify the consequent parameters. In the backward pass, the errors are propagated backward and the premise parameters are updated by gradient descent. An initial fuzzy inference system is first created and improved through the learning process. ANFIS tunes the initial fuzzy inference system by a hybrid algorithm which, combining the gradient descent backpropagation and least square optimization techniques. At each epoch, the error is calculated.

3.10.4.2 Design of ANFIS System

The Neuro-Fuzzy Designer includes four distinct areas to support a typical workflow. This application helps to perform the following tasks:

1. Loading.
2. Generating FIS Structure.
3. Training the FIS.
4. Validating the Trained FIS.

The Neuro-Fuzzy Designer Structure is given in figure 3.27 below which is obtained simply by writing a simple command called “anfisedit” on the command window of mat lab.

Loading, Plotting and Clearing the Data

To train a FIS, it must begin by loading a Training data set that contains the desired input/output data of the system to be modeled. Any data set that is to be loaded must be an array with the data arranged as column vectors, and the output data in the last column.

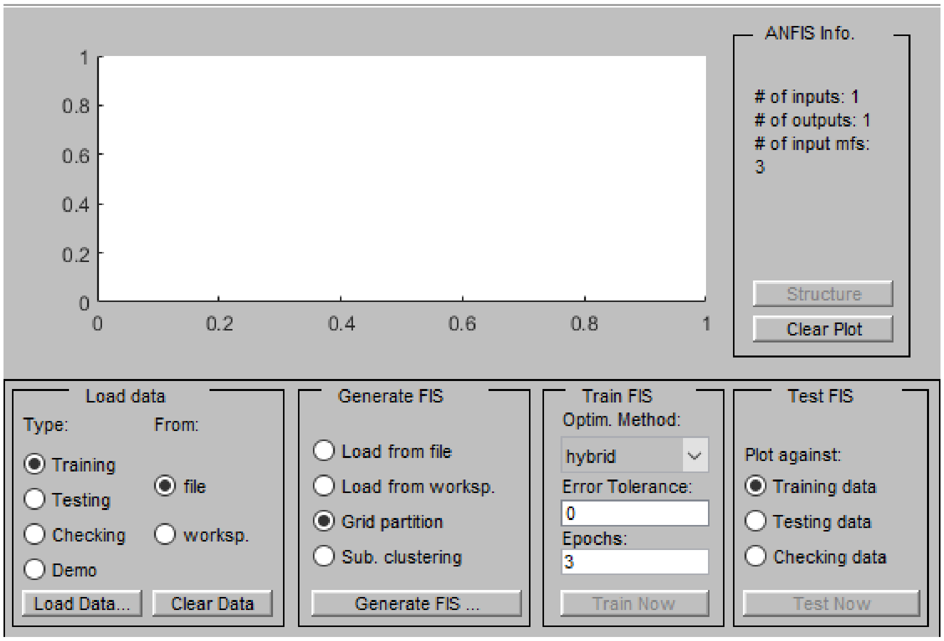


Figure 3.27 The Neuro-Fuzzy Designer Structure

To load a data set using the Load Data portion of the designer:

1. Specify the data type.
2. Select the data from a file or the MATLAB workshop.
3. Click Load Data.

After the data is loaded, it displays in the plot. The training, testing, and checking data are annotated in blue as circles, diamonds, and pluses respectively.

To clear a specific data set from the designer:

1. In the Load Data area, select the data type.
2. Click Clear Data.

This action also removes the corresponding data from the plot.

The following figure 3.28 shows the Input-output relation of the ANFIS controller.

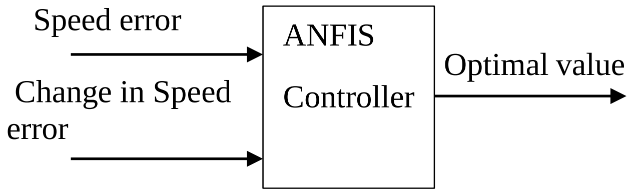


Figure 3.28 Input-output of ANFIS controller

To generate an adaptive neuro-fuzzy system, one thousand data have been considered from the FOPID controller output. Out of these data, 3/4th of total data have been considered as training data set, and the remaining 1/4th of total data have been considered as testing data. The linear membership function type is selected for output, a total of 49 rules are generated, and the number of generations considered is 60. The plot of membership functions for loading, training, surface, and rule views are shown in appendix B.

Fuzzy rules

The principle of designing fuzzy rules is that the output of the controller can make the IM drive system output response dynamic and static performance optimal.

For the system under study, the universe of discourse for both inputs e(t) and de(t) is normalized to the range -1 to 1 as the range of the universe of discourse for the membership functions is selected to be from -1 to 1, and the linguistic labels(fuzzy sets) are defined as{NB (Negative Big), NM(Negative medium), NS (Negative small), ZE (Zero), PS (Positive small), PM (Positive medium), and PB (Positive Big) } and are referred to in the rules bases as {NB,NM,NS,ZE,PS,PM,PB }. The linguistic labels of the outputs in the range 0 to 1 are {Zero, Medium Small, Small, Medium, Big, Medium big, very big} and referred to in the rule bases as {Z, MS, S, M, B, MB, VB}. Formulation of the fuzzy rules is according to the simulation analysis of the PMSM drive system. The rule base is expressed as IF (antecedent)-THEN (consequent) rules shown in equation (4.41).

If e(t) is NB and de(t) is NB THEN K_p, K_i, K_d, μ, and λ are ZE (4.41)

Where

NB is the linguistic label of input, and ZE is the linguistic label of output.

Tables 3.10 show the control rules for determining K_p, K_i, K_d, μ, and λ that are used to tune the FOPID controller gains.

Table 3.10 Rule bases

de/e	NB	NM	NS	ZE	PS	PM	PB
NB	ZE	ZE	MS	MS	S	S	M
NM	ZE	MS	MS	S	S	M	B
NS	MS	MS	S	S	M	B	B
ZE	MS	S	S	M	B	B	MB
PS	S	S	M	B	B	MB	MB
PM	S	M	B	B	MB	MB	VB
PB	M	B	B	MB	MB	VB	VB

Table 3.11 Specifications of the developed ANFIS

Parameters	Value or description
Number of inputs	2
Number of outputs	1
Number of rules	49
Number of epochs	50
Number of training data pairs	1000
Number of checking data pairs	250
Number of output MF	49
Type of output MF	Linear
Type of input MF	Triangular
Number of input MF	7
Aggregation method	Max
Implication method	Prod
And method	Min
Or method	Max
Structure of FIS	Sugeno
Optimization method	Hybrid

CHAPTER FOUR

RESULTS AND DISCUSSIONS

4.1 Simulink Modeling

In this chapter analysis of ANFIS based fractional order PID speed controllers for indirect vector controlled induction motor that gives good performance for the overall drive system in terms of motor parameter variation, load variation, and reference speed variation will be discussed. The analysis begins with the simulation of induction motor without ANFIS based controller especially by using FOPID and validation of induction motor model was checked. Then, analysis with controller will be followed by using ANFIS based FOPID. The good software package which is used to model, simulate, and analyze dynamic systems in this thesis work is MATLAB/Simulink. This is due to, Simulink has the advantage of being capable of complex dynamic system simulations, graphical environment with visual real-time programming, and broad selections of toolboxes. The mathematical equations presented in the previous chapters are used to model the three-phase vector-controlled induction motor in MATLAB/Simulink R2019b environment. The overall system Simulink block includes different sub-functional blocks such as IM model block, fractional order PID controller blocks, coordinate transformation blocks, IFOC block, and SVPWM inverter block. So by simulating every subsystem to obtain expected outputs as well as to satisfy both general and specific objectives which are discussed in chapter one.

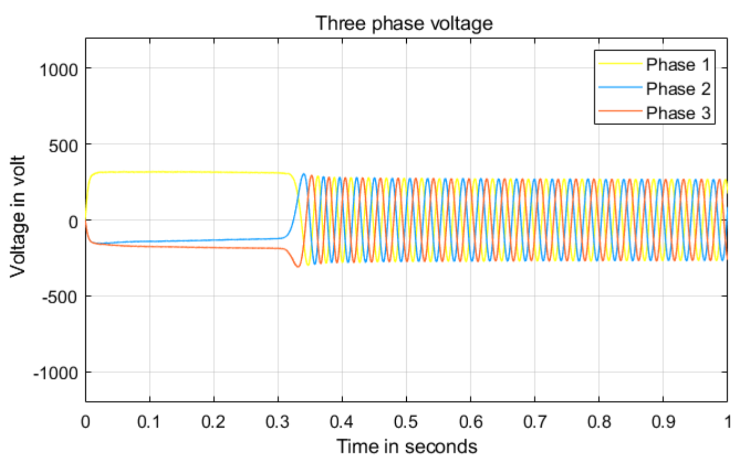


Figure 4. 1 Three-phase sinusoidal wave for transformation at constant reference speed

4.2 MATLAB Simulation Result of SVPWM

Space vector PWM inverter is designed by using subsystem in MATLAB Simulink. It comprises a PWM generator, IGBT voltage source inverter, gates, relational operators, and DC voltage source. Space Vector Pulse Width Modulation (SVPWM) method is one of the advanced, computation-intensive PWM methods and possibly the best among all the PWM techniques.

4.2.1 Switching Pattern of SVPWM

Figure 4.2 below shows, the equivalent switching pattern of space-vector modulation. The waveforms of each phase are 120° apart from each other. The waveforms show that the second lags to the waveform of the first phase by 120° and the third phase lead to the waveform of the first phase by 120°. The SVPWM based on a carrier with a reduced switching ratio is the better implementation of SVPWM because, when the switching ratio has reduced the heat generated in switches is also reduced at the same time which improves the lifetime of switches and the efficiency of the inverter. This pattern is compared with high- frequency carrier signal to produce a gating signal to switch on and off all six switches in the inverter in an appropriate sequence.

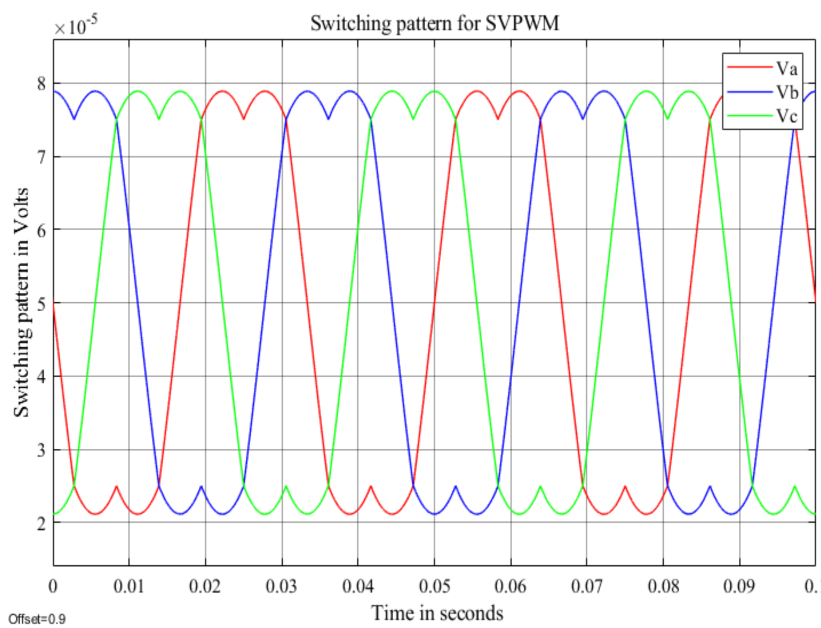


Figure 4. 2 Three-phase voltage for switching PWM

4.2.2 Generated Gate Signal

The main goal of generating an output gate signal based on the SVPWM techniques is to switch the IGBT of the inverter to generate the required three-phase AC voltage to drive the induction motor as shown in the below Figure 4.3. The signal that generated from the driver circuit is out of phase in one leg as shown in Figure 4.4 in different color those pulses is out of phase in on leg because to prevent short circuit which is the cause for damage of the inverter. Therefore, the switches are on and off sequentially based on the pattern of sector vectors to generate three-phase sinusoidal waves.

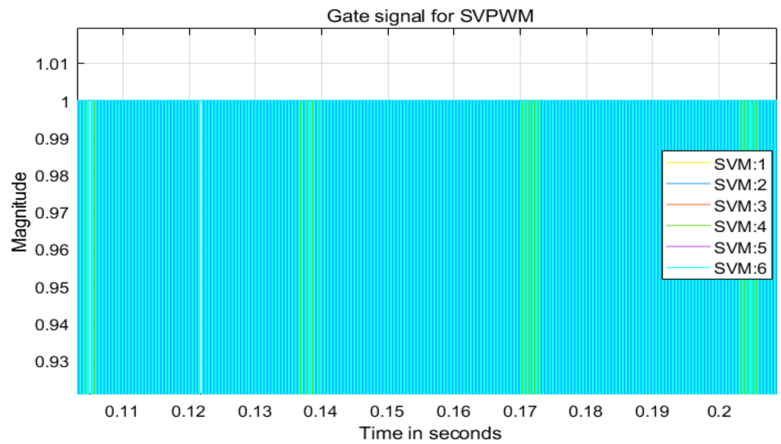


Figure 4. 3 Gate signal for each IGBT

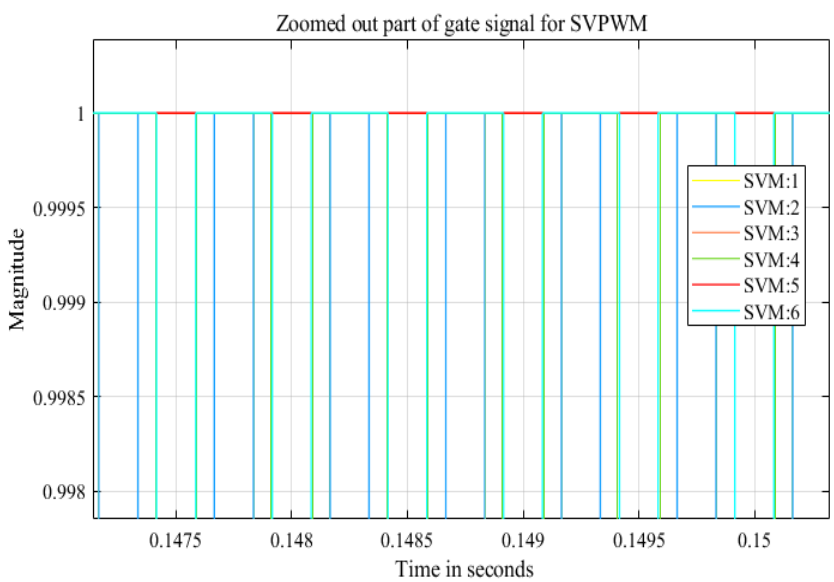


Figure 4. 4 Zoomed out part of gate signal for some IGBT

4.3 Analyzing the proposed system at constant reference input and no load

4.3.1 Three-phase Stator Current Waveform

As shown in figure 4.5, the three-phase stator currents are generated by the three-phase voltage source inverter. This three-phase inverter is controlled by a space vector pulse width modulation block for appropriate stator current generation. These three-phase currents should be equal magnitude and 120° phase shift with each other for appropriate rotating flux generation. Hence, the system can feed the appropriate stator voltage to the Induction Motor.

If the voltage applied to the motor is applied with appropriate magnitude and frequency, the speed of the motor can track the reference value.

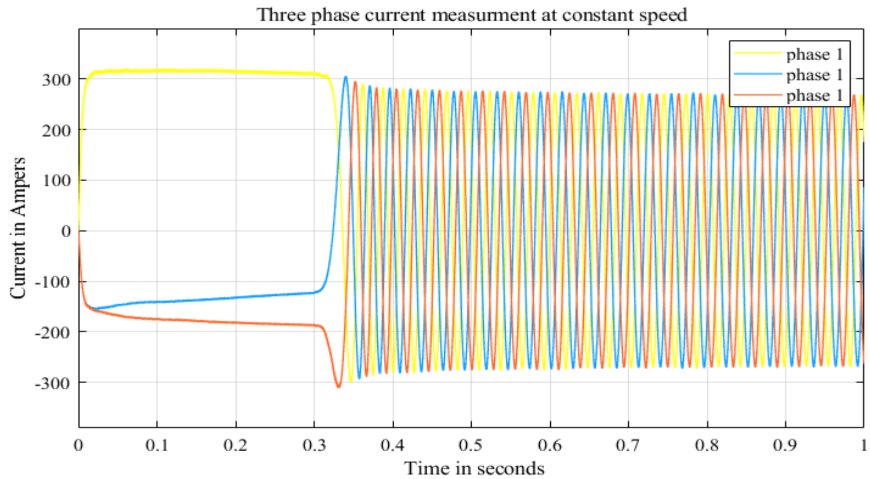


Figure 4. 5 Three-phase stator current at constant reference speed

4.3.2 Transformation of stator current from abc to dq

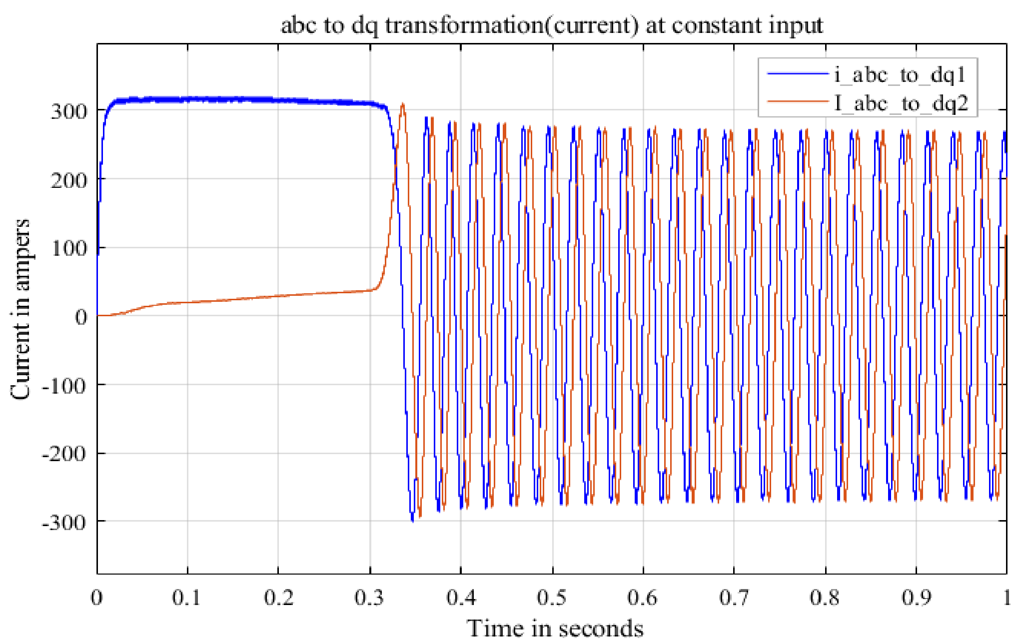


Figure 4. 6 Stator current transformation from abc to dq at constant reference speed

As shown in figure 4.6, contains both Clarke and park transformation in which the transformation from abc to alpha-beta is found in actual design. So the simulated output shows that transformation can follow the wave of three-phase voltage.

4.4 Set Point Tracking Capability For Constant Reference Speed At No Load

As we know, the performance of the estimated speed must converge with the actual speed value, especially during the transient response. Depending on this idea the performance of a proposed GA-FOPID controller convergence of the estimated rotor speed to the actual speed.

4.4.1 Speed Response For Using GA-FOPID

To analyze GA optimized FOPID speed controller MATLAB simulation results of the IM motor will be analyzed and the Simulink block is found in the appendix A. The simulation results are carried out at 120rad/sec reference speed. The controllers are evaluated based on performance criteria such as rise time, settling time, steady-state error, and peak overshoot. Figure 4.7 shows the speed versus time response of GA- FOPID speed controlled IM drive at no-load condition. Here response for the FOPID is good, but it doesn't try to track the estimated speed to set point.

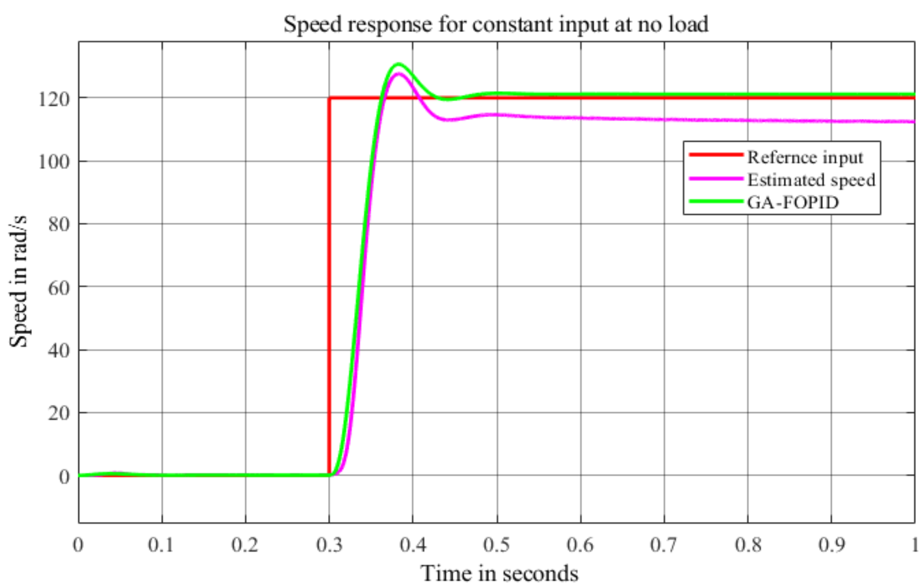


Figure 4. 7 Speed response of GA-FOPID control system for constant reference speed

4.4.2 The Error Signal Between Actual and Reference Speed

Figure 4.8 shows the speed error response between actual GA-FOPID controllers and reference speed. As the figure shows speed error of the GA-FOPID controller settles to its minimum value faster due to it is an online optimization because FOPID parameters are set using the GA optimization technique.

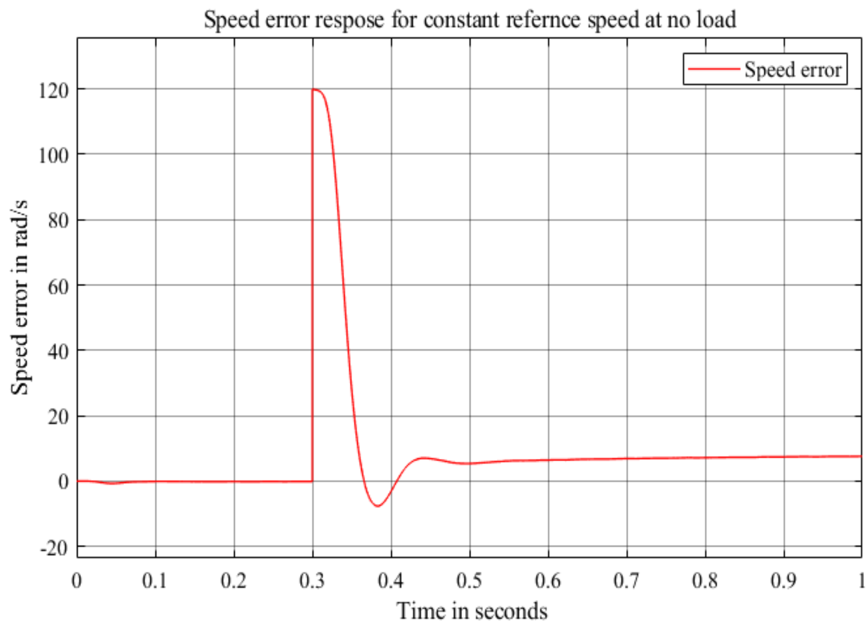


Figure 4. 8 Error signal between the actual and estimated speed at constant reference speed

As shown in figure 4.7 and figure 4.8, the maximum overshoot at t=0.3 is due to speed change and immediately comes back to zero that is the expected result.

4.4.3 Estimated Rotor Angle and Flux Response

The angle of rotor flux to the stationary frame d-axis is estimated with the proposed flux linkages in the rotating reference frame are found by using voltage and current flux estimation and then the rotor flux angle improved by using an ANFIS by training speed error. Figure 4.9 shows the estimated rotor flux angle in rad which is used to calculate sectors of SVPWM.

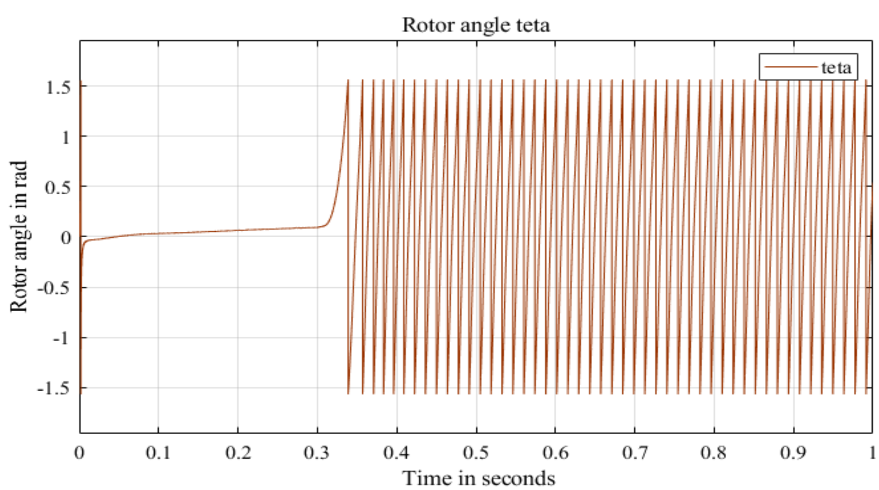


Figure 4. 9 Estimated rotor flux angle at constant reference speed

The estimated rotor flux with a reference speed of 120 rad/sec is shown in Figure 4.10. Both the dq-axis fluxes are 90-degree out of phase.

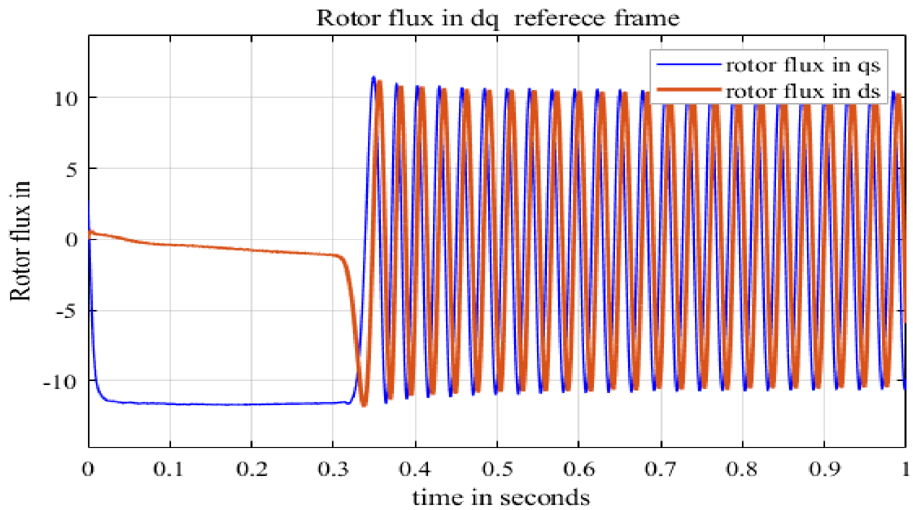


Figure 4. 10 Estimated rotor flux in dq-axis reference frame at constant reference speed

4.4.4 Torque Response

Electromagnetic torque response in figure 4.11 shows the same form as quadrature axis stator current since, i_{qs} is a torque-producing component of stator current. The electromagnetic torque of GA optimized FOPID controller settles in less time. Once the machine reaches the set speed of 120 rad/s, the average torque of the machine becomes nearly zero. Even though the peak electromagnetic torque at initial is high in the FOPID controller, it oscillates around zero fast.

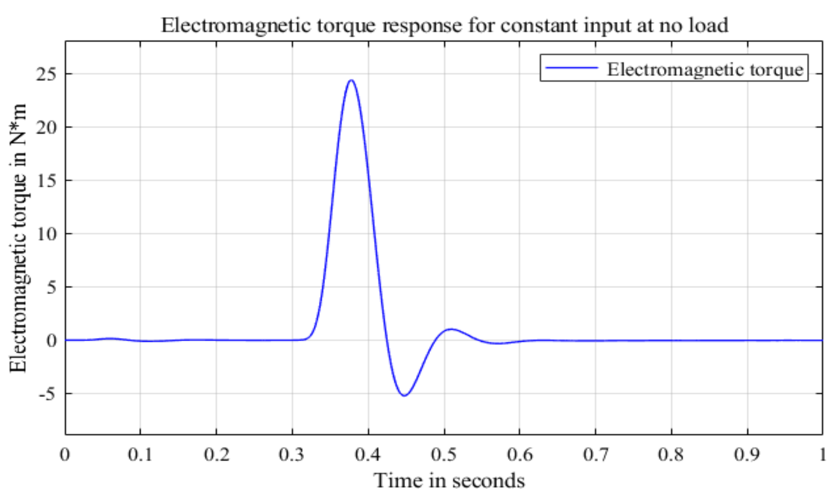


Figure 4. 11 Electromagnetic torque response for constant step input at no load

Figure 4.12 shows that zero torque is applied to the proposed system to see transient responses of the GA-FOPID controller for constant reference step input speed 120rad/s.

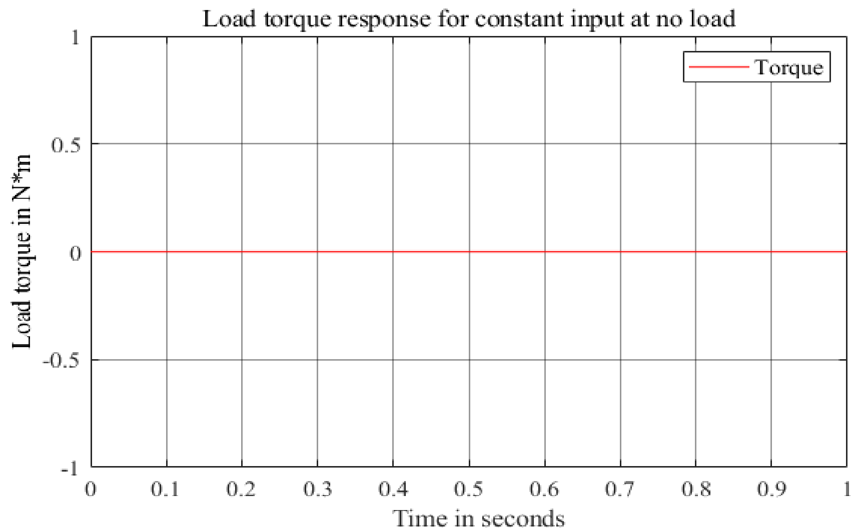


Figure 4. 12 Load torque response for constant step input at no load

4.5 Tracking Capacity of Controller For Variable Reference Speed At No Load

4.5.1 Stator Current And abc To dq Transformation Response

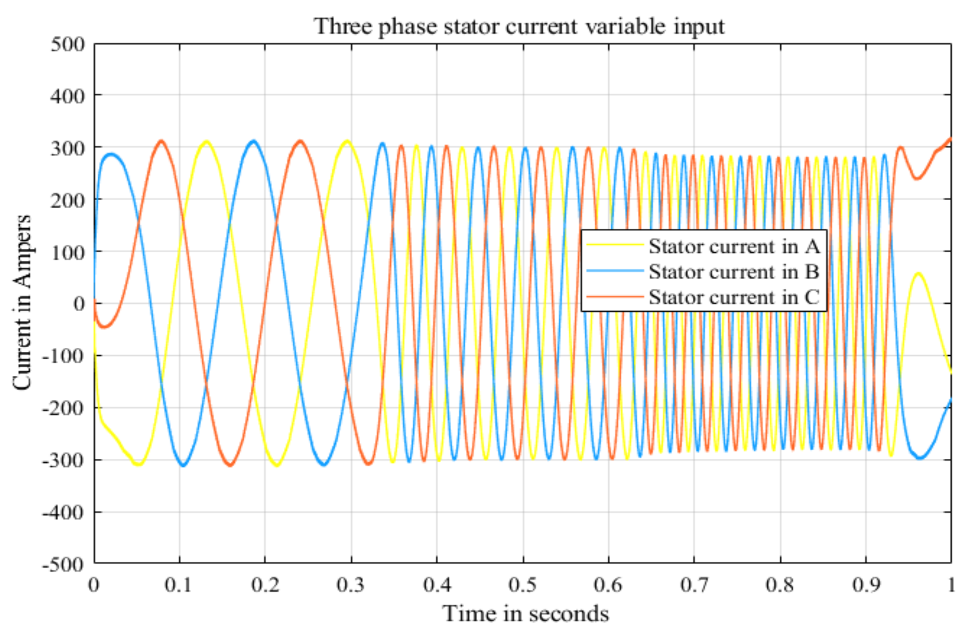


Figure 4. 13 Three-phase stator current waveform for variable input at no load

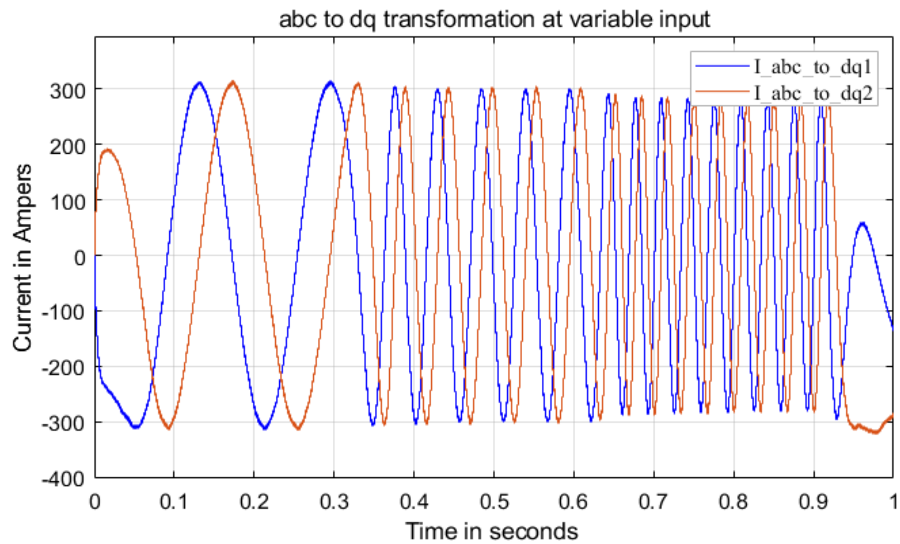


Figure 4. 14 Stator current transformation from abc to dq at variable reference

As shown in figure 4.13 and figure 4.14, the waveform shows some difference when we compare it to the same wave as in the case of constant reference input. This variation is due to the application of variable reference input. That means when input varies, the curve of both transformation waves varies immediately.

4.5.2 Estimated Rotor Flux Angle

Figure 4.15 below shows the estimated rotor flux angle at variable reference speed input. When the speed is large/minimum the frequency of the estimated rotor angle is large/minimum respectively. That is why the difference happens when we can compare it with the flux of constant reference input.

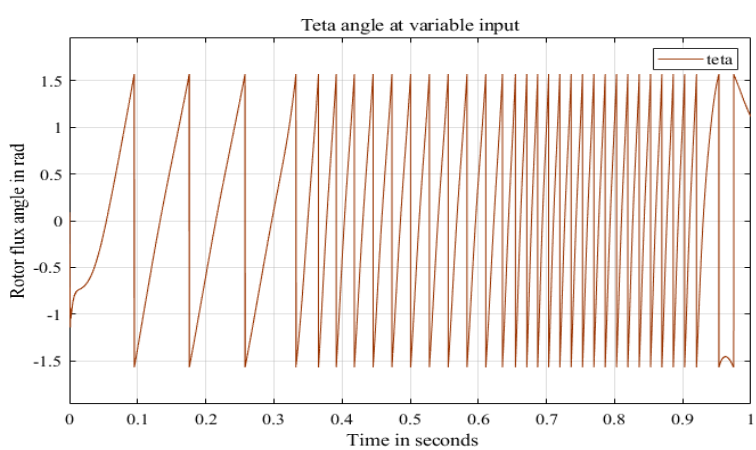


Figure 4. 15 Rotor flux angle response for variable input at no load

4.5.3 Speed Response Using GA-FOPID

The simulation results are carried out for different reference speeds at no load condition using the same parameters of the induction motor, the tracking performance of the estimator can be examined by changing the speed of the system. As shown in Figure 4.16 the proposed estimator tracks the pulse with different speed reference inputs. This shows the tracking performance of the estimator and the actual speed to the reference speed can be examined by changing the reference of the system. Reference speeds that are taken for simulation are 20, 60, 100, 0 in rad/sec applied sequentially at 0.3 sec time interval. The simulation results are carried out for different reference speeds at no-load conditions. As shown in Figure 4.16 the estimated and the actual GA optimized FOPID controller speed follow the reference speed with poor accuracy. From the simulation result here, the GA-FOPID controller shows that is better in overshoot, settling time, rise time, and steady-state error as before but it does not converge the estimated speed to reference speed.

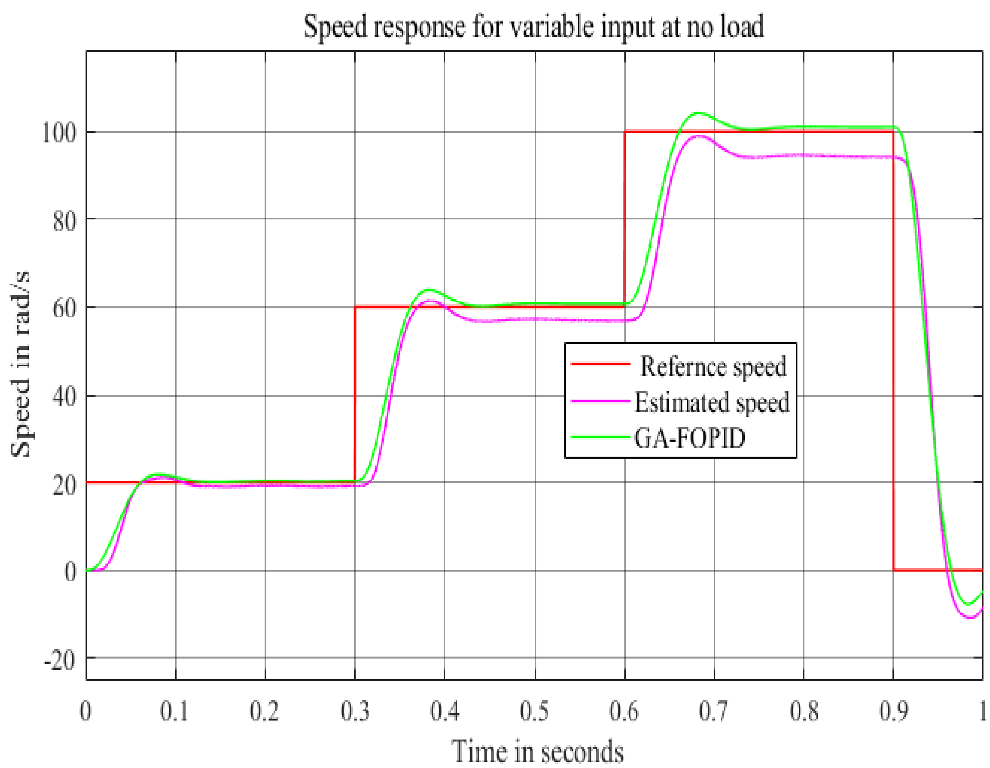


Figure 4. 16 GA optimized FOPID speed response for variable input at no load

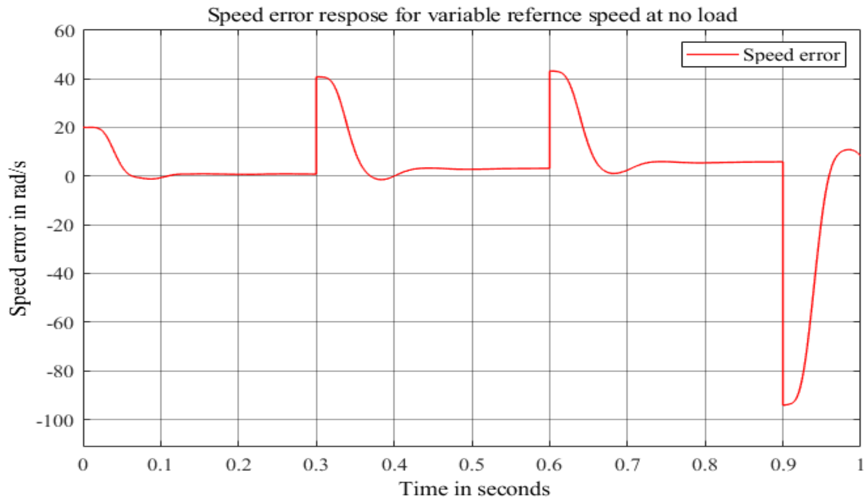


Figure 4. 17 Error signal between the actual and estimated speed of variable input

Figure 4.17 shows the error speed between actual rotor speed and variable input speed in which at each step of an applied variable input, approximately zero speed error is obtained. But the high overshoot on each starting step is due to applied speed and immediately comes to zero with the help of the proposed system controllers.

4.5.4 Electromagnetic Torque Response

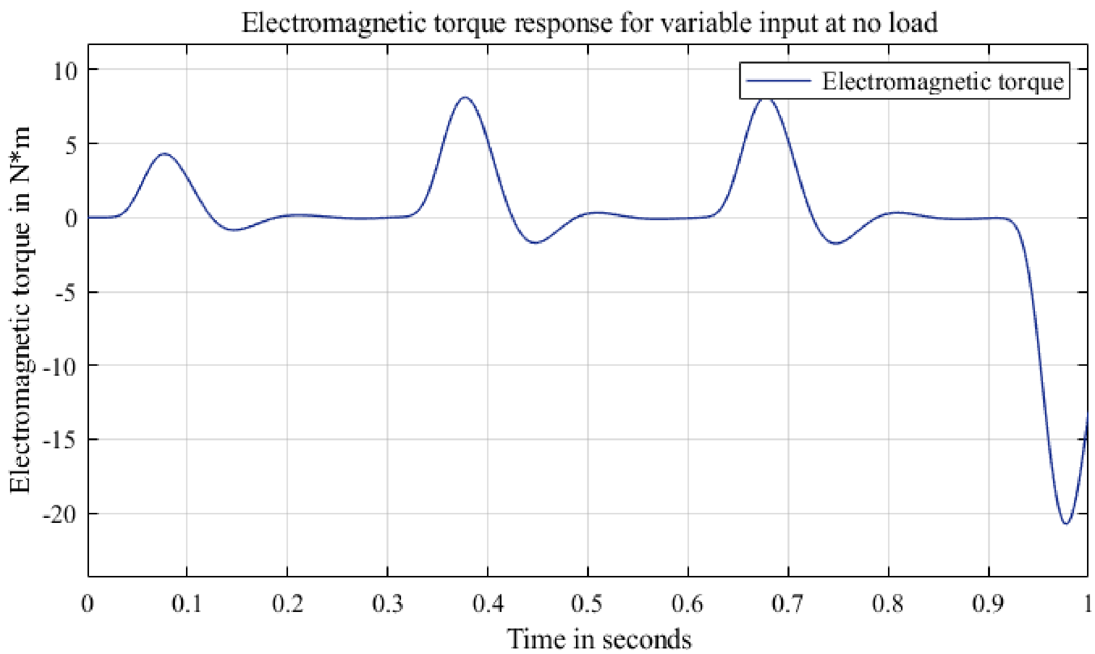


Figure 4. 18 Electromagnetic torque for variable reference input at no load

Figure 4.18 shows the electromagnetic torque for IM drive using GA-FOPID controllers. For electromagnetic torque response, it has the same form as quadrature axis stator current response in which the torque response of controllers settles fast near to zero immediately after maximum overshot at the speeded area.

4.6 Effect of Motor Parameter Variation

As we know that control of induction motors for high performance is a challenging problem because they exhibit significant non-linearity and it is known that uncertainties of motor parameters and unknown external disturbances can degrade the performance of the drive. To see the effect of motor parameter variation on the performance of IM drive, assume that the parameters such as inductance and poles are remaining constant, and it is evident that the speed response of the system is not affected by the variation. The resistance of IM changes because of temperature variation caused by machine losses. As the resistance value varies by temperature, the performance of controllers can be deteriorating. Let’s see the variation in resistance rotor windings in which double rotor resistance from 0.351 to 0.702 and reduce by half again to 0.1755. Figure 4.19 below shows the speed responses of IM drive due to rotor resistance variations without load at a constant reference speed of 120rad/sec. So, the changes in rising time, settling time, and steady-state error and peak overshoot especially high amount of steady-state error is produced is high compared to the same response in case of actual resistance of selected IM.

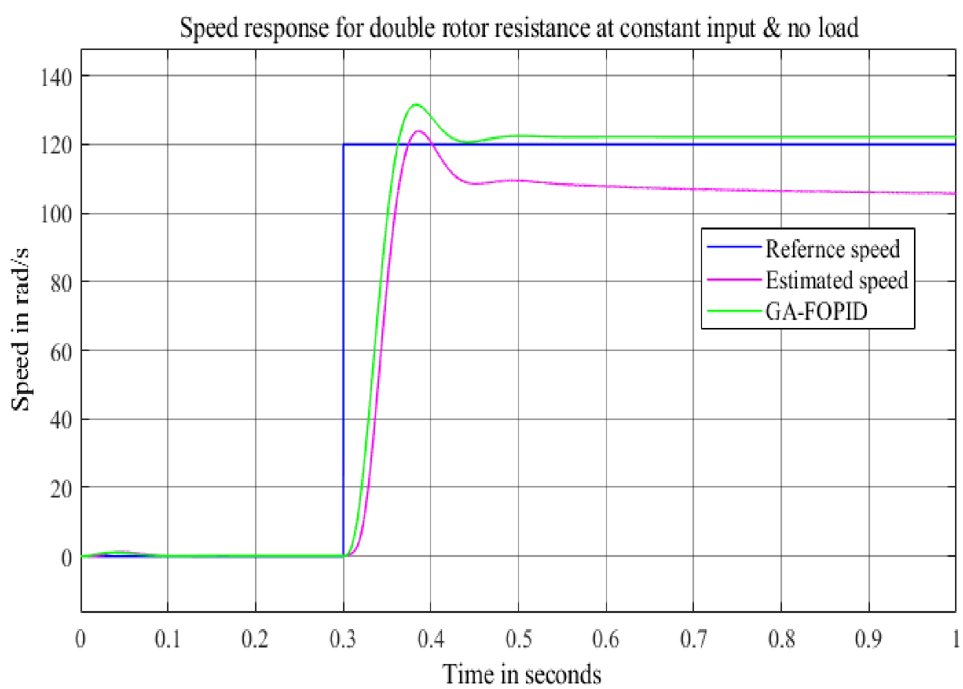
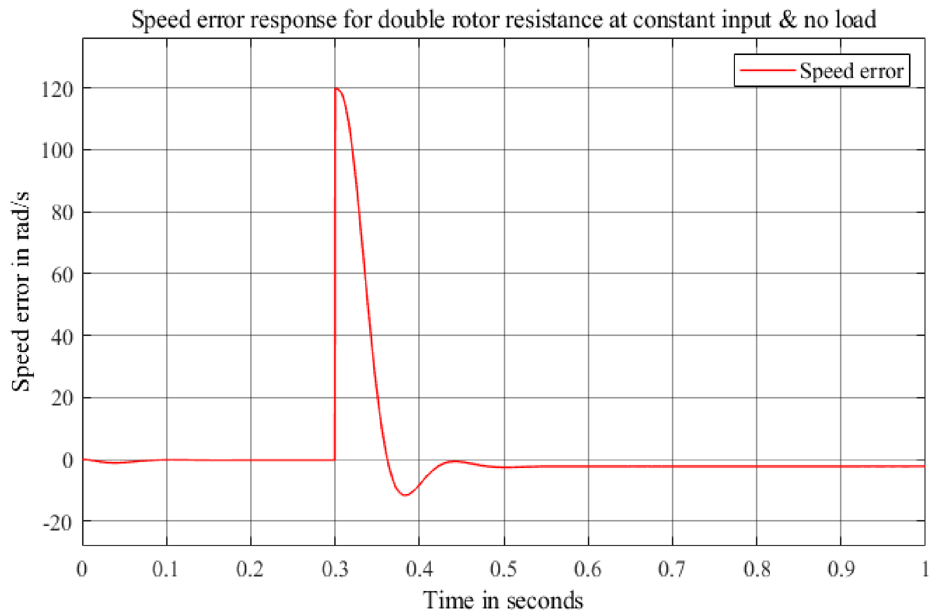
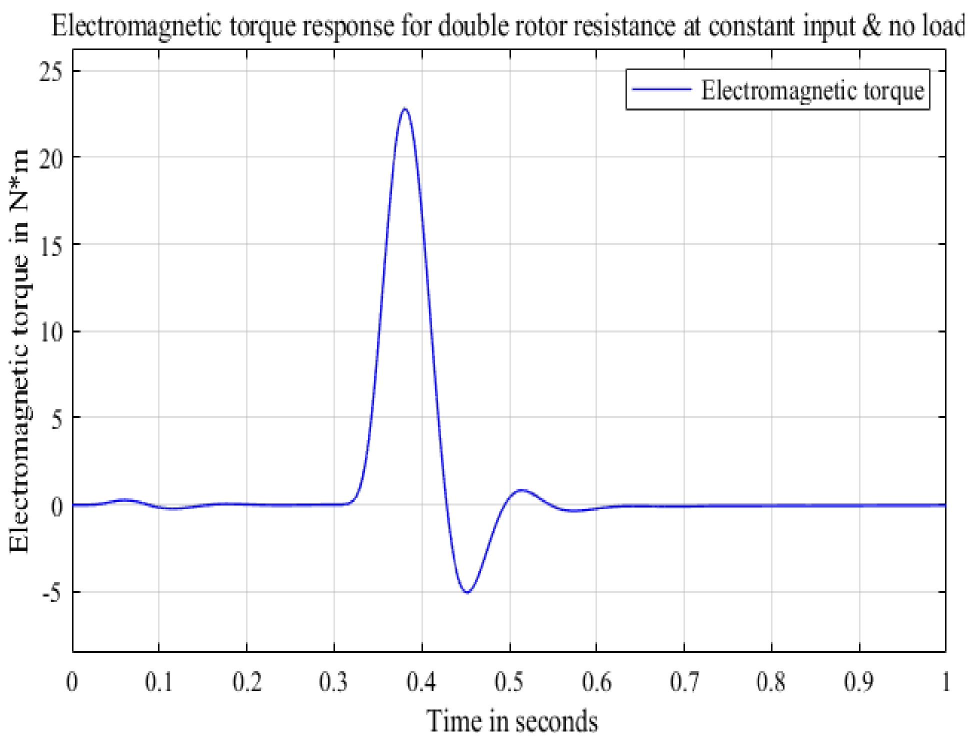


Figure 4. 19 Speed response for doubling rotor resistance and no load

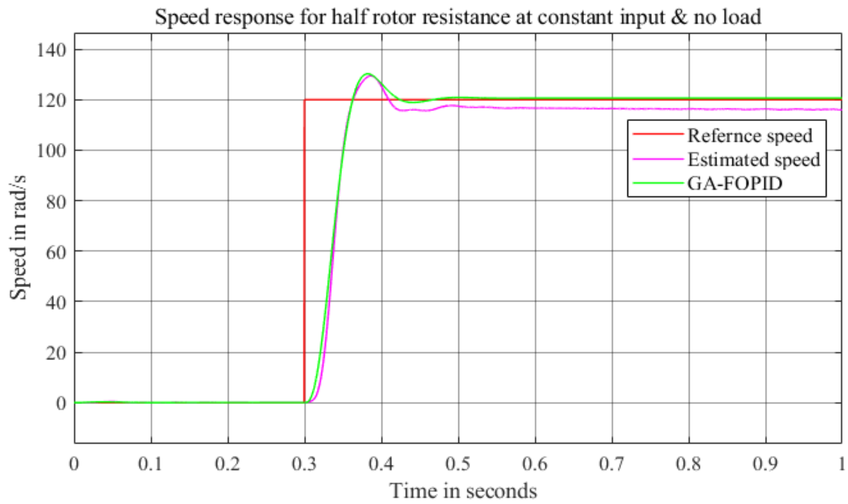


a)

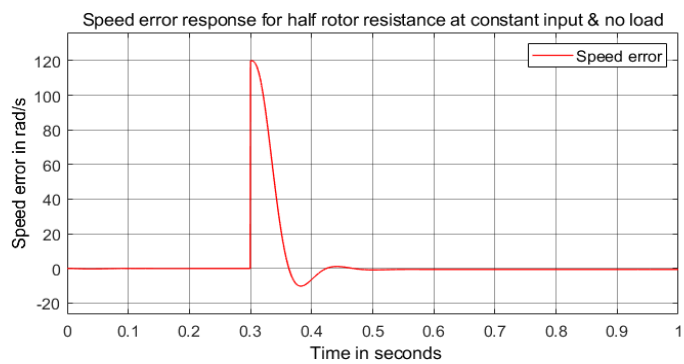


b)

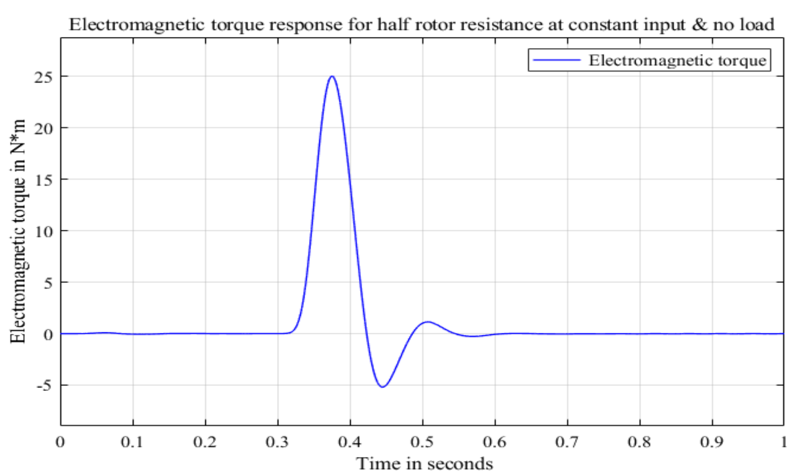
Figure 4. 20 a) Speed error b) Torque response for doubling rotor resistance



a)



b)



c)

Figure 4. 21 a) Speed b) Speed error c) Torque response for half rotor resistance

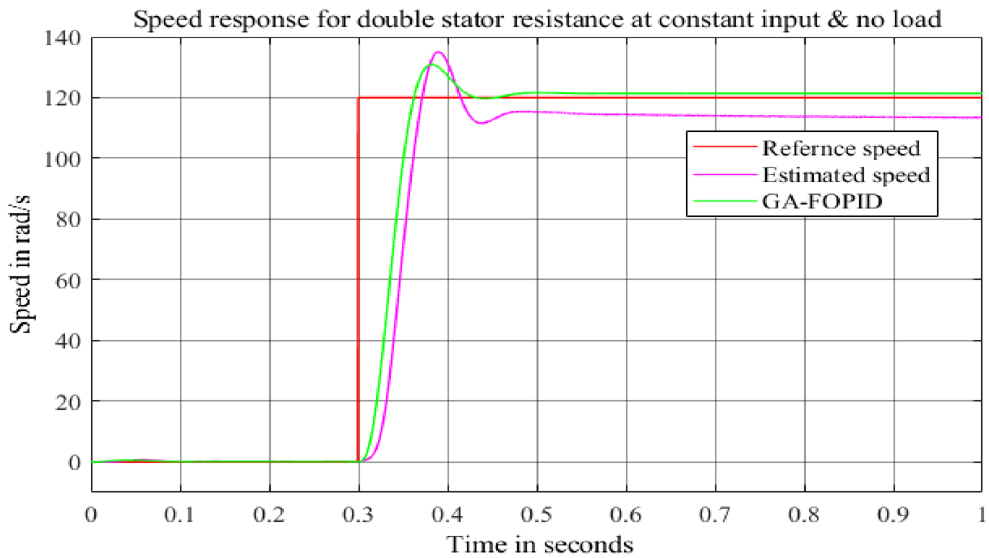


Figure 4. 22 Speed response for double stator resistance at no load and constant input

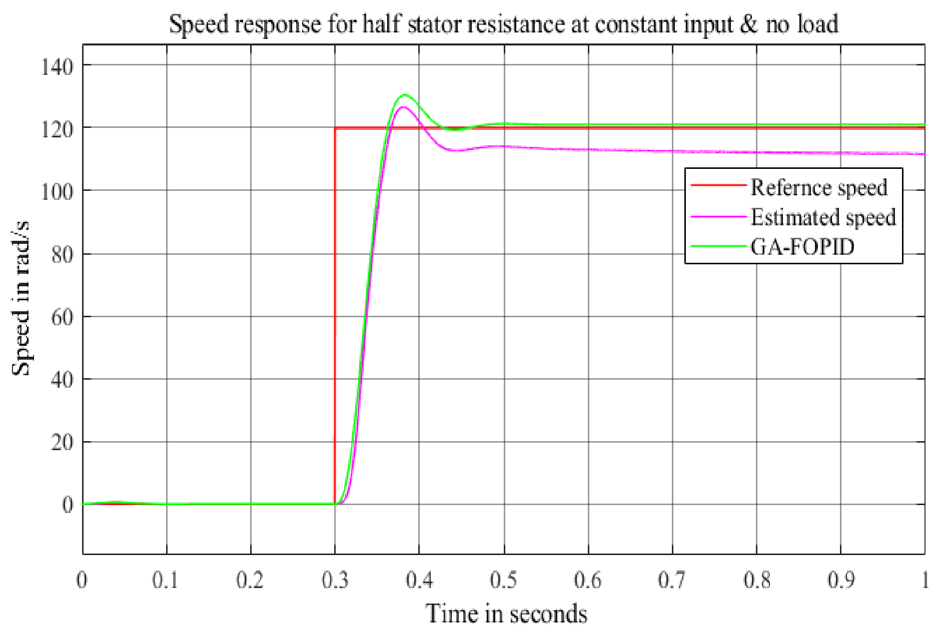
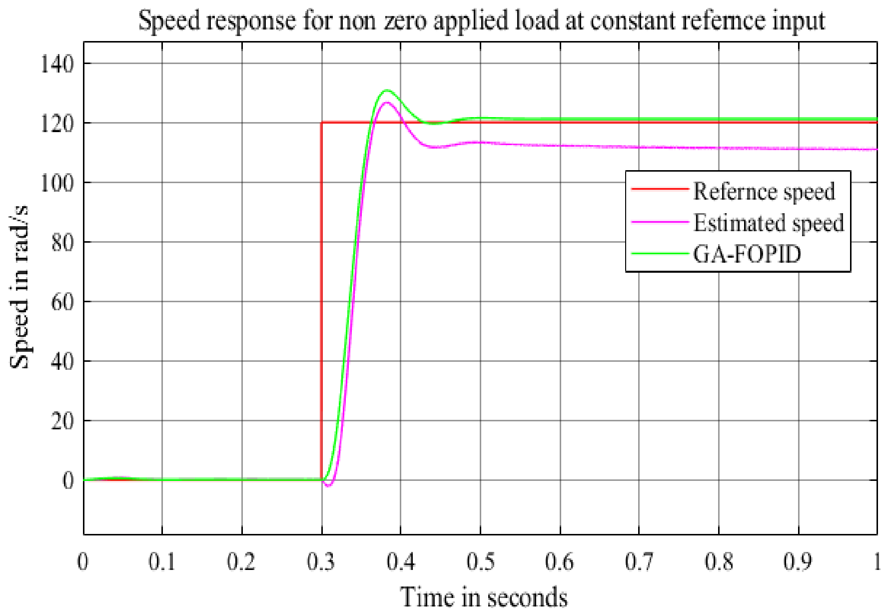


Figure 4. 23 Speed response for half stator resistance at no load and constant input

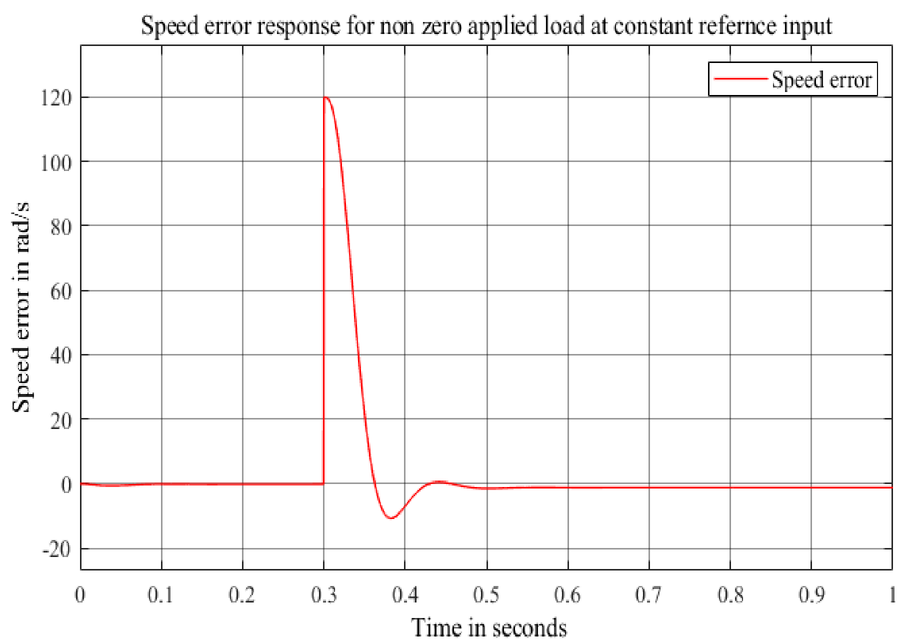
From the simulation result on motor parameter variation especially stator and rotor resistance, the electromagnetic torque, error, and speed response of GA-FOPID controllers at different values of rotor resistance without load and the tracking abilities are better in the case of reducing stator and rotor resistances. So speed response is more sensitive to the variation in motor parameters like the resistance of both stator and rotor windings

4.7 Effect of Load Torque Variation

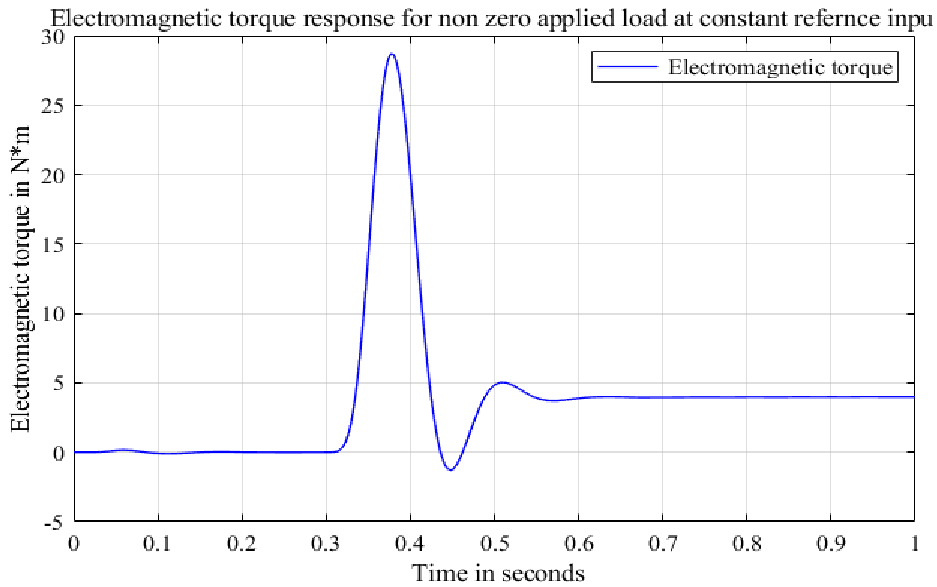
In order to testify the robustness of the controlled system, apply torque rather than no load. So in this thesis, apply 4 N*m torque to the load at a time of 0.3 seconds at 120rad/sec reference speed for a total simulation time of 1 second are considered and after 0.3 seconds it will come back to no-load condition. So we will see the effect on the speed response in this interval.



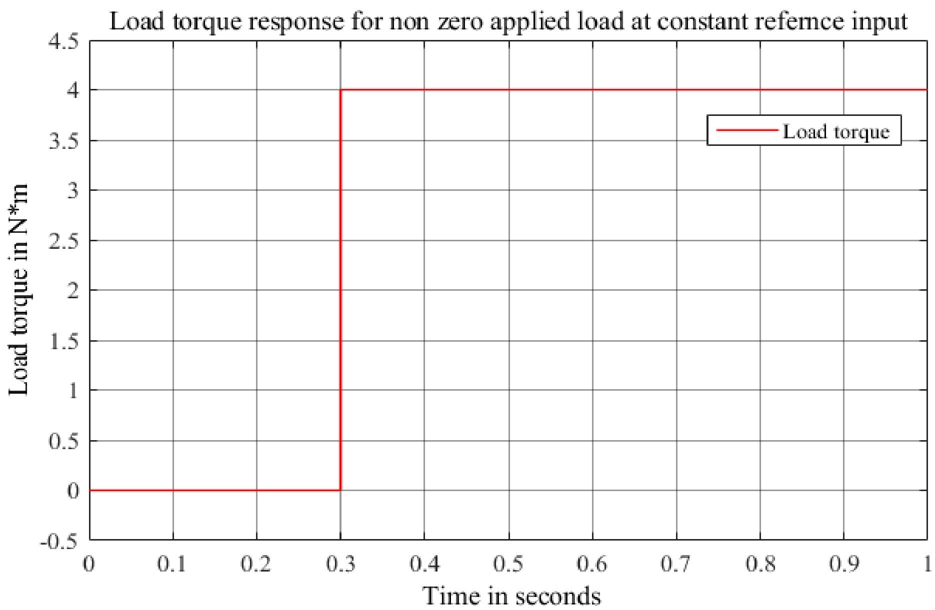
a)



b)



c)



d)

Figure 4. 24 a) Speed b) Speed error c) Torque d) Load torque with load torque variation

In the simulation, IM drive experiences sudden changes in the load torque at t=0.3sec,

Figure 4.24 a) shows the speed response of the GA-FOPID control system with load torque variation. As inferred from the figure GA-FOPID control system is more sensitive to load torque variations. Figures 4.24b and Figures 4.24c show the speed error and torque response with load torque variation respectively and they are more sensitive to parameter variation.

4.8 ANFIS Based Genetic Algorithm optimized FOPID Controller Response

The GA-FOPID speed controller is performed and it has good speed dynamic performance in all aspects constant reference input, variable reference input, motor parameter variation, and shifting IM from no load to loaded condition. But, in the case of load change conditions, the motor speed cannot track the desired speed. As a result, the steady-state error is increased when the load is increased. To reduce this effect to some extent neural network containing ANFIS is used based on the GA-FOPID result. ANFIS trains the error between setpoint and motor output and it helps to get the least amount of fitness function. Finally minimized error is given to the FOPID to tune the parameters through GA. The result of the ANFIS based GA-FOPID speed controller shows that it has better performance than GA-FOPID speed controller results as shown in Table 4.1.

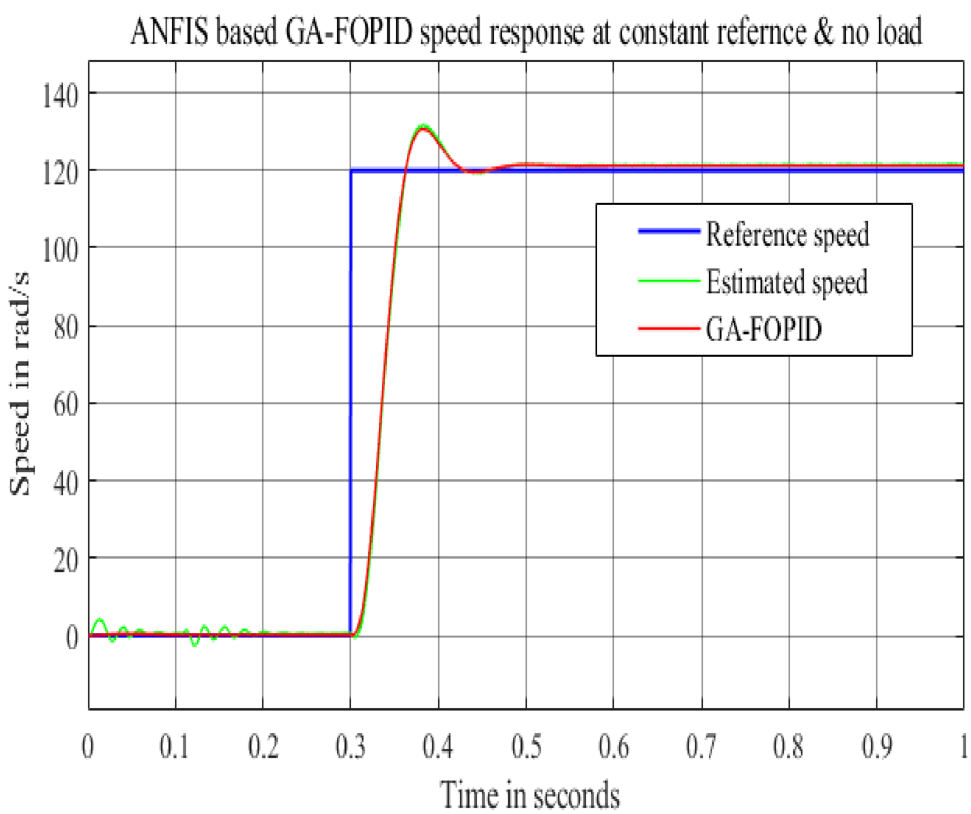


Figure 4. 25 ANFIS based GA optimized FOPID speed response for constant input

Figure 4.25 show that the response is more accurate when compared with the responses discussed in the last sections

Figure 4. 26 ANFIS based GA-FOPID controller speed error response

As shown in figure 4.26, ANFIS based GA-FOPID controller has less error compared to conventional PID controller the actual speed tracks the reference speed with a minimum acceptable error which is an acceptable result.

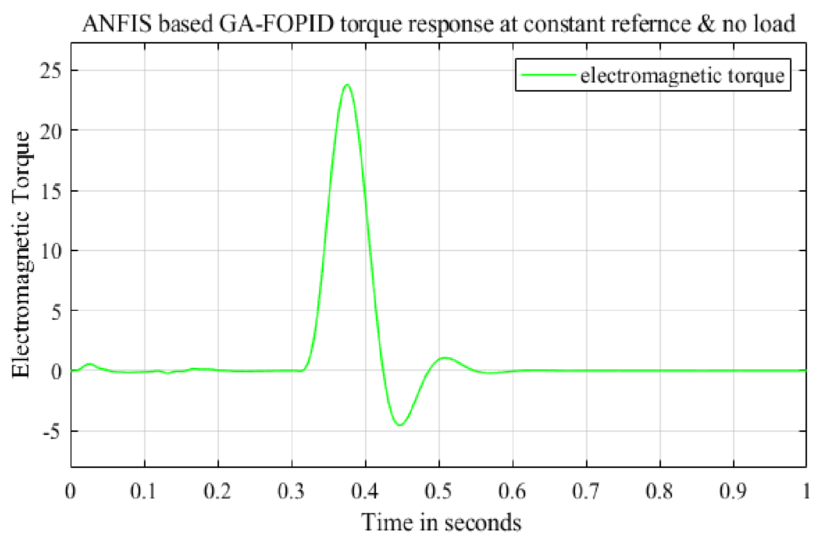


Figure 4. 27 ANFIS based GA optimized FOPID torque response for constant input at no load

Figure 4.27 shows the electromagnetic torque responses of IM drive using the ANFIS based GA-FOPID and the simulation result shows all controllers can track the load torque. It is seen that the quadrature axis stator current increases and initially the speed decreases with increasing load and the electromagnetic torque follows the load torque. But, the speed settles back to the reference speed. This result is as expected. In this study, different speed controllers are performed. These are the GA-FOPID speed controller, and the proposed ANFIS based GA-FOPID speed controllers. For comparison purposes, the simulation is performed in constant set-point conditions. From control system theory, the system is said to have better dynamic performance if it has a lower rise time, lower percent overshoot, lower settling time, and lower peak time.

Finally, the results show that ANFIS based GA-FOPID controller has improved system performance like system error, overshoot, and rising time, as well as Electromagnetic torque, rotor speed, and stator phase currents of the system, have good transient and steady-state response rather than other controllers. It is summarized in table 4.1in which the GA-FOPID control alone has bigger overshoot, rise time, and steady-state error compared to ANFIS based GA optimized fractional order PID controller. The ANFIS based GA-FOPID controller shows a faster response in comparison with FOPID speed controllers. ANFIS based GA-FOPID tries to speed up the performance of the IM drive. Further, it can also be observed that using the ANFIS based GA-FOPID control, the system stabilizes in very little time compared to the FOPID controllers because of the training process of the ANFIS involved and the proper selection of the rule. Table 4.1 shows the detailed speed dynamic performance parameters of each controller's result. GA-FOPID speed controller has poor performance than others especially in the case of steady-state error and overshoot. But the percentage of changes are varying from controller to controller and ANFIS based GA-FOPID controller has a low rising time compared to GA-FOPID controllers.

As a conclusion, the simulation results show ANFIS based GA-FOPID controllers are less sensitive to rotor resistance variation, input reference change, and load application as well as more robust, while FOPID controller is more affected by rotor resistance variation.so, ANFIS based GA-FOPID controller outperforms FOPID controllers in all performance criteria.

Table 4. 1 Speed controller comparison of performance parameter results at no load

Performance criteria	GA-FOPID					ANFIS FOPID
	Normal	1/2Rs	2Rs	1/2Rr	2Rr	Normal
Rise time(sec)	0.355	0.315	0.315	0.3	0.338	0.307
Settling time(sec)	0.412	0.47	0.48	0.5	0.52	0.403
Peak time(sec)	0.365	0.364	0.39	0.368	0.38	0.309
Steady-state error (%)	7	6.4	6.5	4	8	0.5
Overshoot (%)	9	7	10	8	11	6

CHAPTER FIVE

CONCLUSIONS AND RECOMMENDATIONS

5.1 Conclusions

This paper presented that ANFIS based GA-FOPID controller was used to optimize speed control of IM motor.

Induction motors have some special characteristics like low cost, less maintenance, lower weight, high efficiency, improved ruggedness, and reliability widely used in many areas of industrial application. On the other hand, mechanical loads in the industries should not only be driven at a constant speed but should also be driven at the desired speed by varying the speed of the motor. So, based on these two conditions the need for speed control methods for the induction motor arises. As we know different kinds of speed control methods are used for an induction motor, and this thesis focuses on the ANFIS based GA-FOPID speed controllers which are designed for speed control of indirect vector control of an induction motor drive. The parameters of the FOPID controller are optimally tuned by the Genetic Algorithm. Performance of all controllers is compared in simulation at four various conditions, no-load, loaded, reference speed input variation, and motor parameter variation.

From the obtained simulations result the performance of speed control of the indirect vector control of an induction motor for GA-FOPID control has been analyzed. According to the simulation result, the GA-FOPID speed controller has a settling time of 0.412sec, 9% overshoot, 0.355sec rise time, 0.365sec peak time, and 7% steady-state error for 120 rad/s at no-load conditions. On the other hand, the ANFIS based GA-FOPID controller has a settling time of 0.403sec, 6% overshoot, 0.3.7sec rise time, and 0.5% steady-state error for 120 rad/s. Thus, values show that the ANFIS based GA-FOPID speed control of indirect vector control of induction motor gives a better speed response than that of the conventional FOPID controller at no-load conditions. The performance of FOPID and ANFIS based GA-FOPID controller evaluated under the application of sudden load (4Nm) change and constant setpoint condition for both performed control systems. The settling time, rise time, and peak time values are increased when compared to the system simulated under no-load conditions.

This shows that when the load changes the system performance will be decreased. There is no variation in the case of overshoot but there is a steady-state error in the result with the system simulated under no-load conditions. The system also simulated under setpoint change conditions but the dynamic performance in both controllers is similar to the system results using constant setpoint conditions.

From the simulation results, it can be concluded that the proposed ANFIS based GA-FOPID controller improves the overshoot, rise time, settling time, good tracking of reference speed, and maintaining the speed even while applying the load torque at the no-load and on-load condition as well as motor parameter reference input variation. Also, with these results ANFIS based GA-FOPID controller has more flexibility from a control perspective.

5.2 Recommendations

Even though ANFIS eliminates unwanted trial and error methods to determine parameters of membership functions using layers of ANN, it does not determine which type of membership function is suitable for the data provided. Therefore, additional investigations are needed to determine the type of membership function used in the ANFIS controller. The motor parameters variation in this thesis work was focused on the resistance value. Therefore, study the other motor parameters variation will remain a remarkable study.

This thesis focuses on GA optimization techniques from various optimization techniques to determine the optimized parameters of FOPID controllers. So this thesis can also be extended for the application of other optimization techniques like Local Search Optimization, higher-order slide mode, Ant-Colony Optimization (ACO), and Particle Swarm Optimization those may improve the performance of controllers. I advise that new coming researcher to implementation prototype for this system by using hardware implementation tools like Digital signal processing as well as Field programmable gate array.

REFERENCE

[1] Muhammad H. Rashid, *"Power Electronics: Circuits, Devices & Applications."* Bangladesh 4th edition, 2014.

[2] F. Giri, *"AC electric motors control: advanced design techniques and applications."* United Kingdom: John Wiley & Sons, 2013.

[3] M. Andrzej and Trzynadlowski, *"Control of induction motors."* San Diego: Academic Press, 2000.

[4] A.E.Fitzgerald, *"Electrical Machine."* Cambridge sixth edition, MA March 5, 2006.

[5] B. S. Naik, "Comparison of Direct and Indirect Vector Control of Induction Motor," *Int. J. New Technol. Sci. Eng.*, vol. 1, no. 1, pp. 110–131, 2014.

[6] N. Mohan, *"Advanced electric drives,"* Second edi. John Wiley & Sons, Inc., Hoboken, New Jersey, 2014.

[7] B. BIMAL K, "Modern Power Electronics and AC Drives." Pearson Education, 4th edition, 2004.

[8] M. Filippich, "Digital control of a three phase induction Motor," The University of Queensland, 2002.

[9] M. Li, "Differential-algebraic approach to speed and parameter estimation of the induction motor," univeresity of Tennessee, 2005.

[10] S. J. Chapman, *"Electric Machinery Fundamentals,"* Fifth Edit. 2004.

[11] M. Jannati, T. Sutikno, N. R. N. Idris, and M. J. A. Aziz, "A Novel Method for Rotor Field-Oriented Control of Single-Phase Induction Motor," *Int. J. Electr. Comput. Eng.*, vol. 5, no. 2, p. 205, 2015.

[12] P. Nagasekhar Reddy and J. A. Linga, Reddy P, "Sensorless control of induction motor using Simulink by direct synthesis technique," *Int. J. Electr. Eng.*, vol. 4, no. 1, pp. 23–32, 2011.

[13] F. R. Yasien and M. S. Ali, "Performance Comparison between Conventional PI and PI-Like Fuzzy Speed Controller for Three Phase Induction Motor," *Am. Sci. Res. J. Eng. Technol. Sci.*, vol. 42, no. 1, pp. 148–156, 2018.

[14] S. Al-Ratrout, A. Saleem, and H. Soliman, "Optimization methods in fractional order control of electric drives: A comparative study," in *2015 10th International Symposium on Mechatronics and its Applications (ISMA)*, 2015, pp. 1–6.

[15] B. R. Ramchandra Bhosale, “Indirect Vector Control of Induction motor using Fuzzy Logic Controller,” in *International Conference on Advances in Engineering and Technology, ICAET-2014, Jan*, vol. 8.

[16] M. Popescu, *“Induction motor modelling for vector control purposes.”* Helsinki University of Technology, Laboratory of Electromechanics, Espoo, 2000.

[17] B. Akin and N. Garg, “Scalar (V/f) control of 3-phase induction motors,” *Texas Instruments Inc. Texas*, 2013.

[18] M. Zeraoulia, M. E. H. Benbouzid, and D. Diallo, “Electric motor drive selection issues for HEV propulsion systems: A comparative study,” *IEEE Trans. Veh. Technol.*, vol. 55, no. 6, pp. 1756–1764, 2006.

[19] M. P. Kazmierkowski, R. Krishnan, and F. Blaabjerg, *“Control in power electronics:Selected problems.”* Acadamic press, 2002.

[20] R. W. De Doncker and D. W. Novotny, “The universal field oriented controller,” *IEEE Trans. Ind. Appl.*, vol. 30, no. 1, pp. 92–100, 1994.

[21] I. Petras, “Fractional-Order Nonlinear Systems: Modeling, Analysis and Simulation,” *Beijing, Beijing Springer-Verlag Berlin Heidelb.*, 2011.

[22] A. Tepljakov, “Fractional-order calculus based identification and control of linear dynamic systems,” *Est. ,Grant no. 8738*, 2011.

[23] D. Cafagna, “Fractional calculus: A mathematical tool from the past for present engineers,” *IEEE Ind. Electron. Mag.*, vol. 1, no. 2, pp. 117–122, 2007.

[24] I. P. Riccardo Caponetto, Giovanni Dongola and Luigi Fortuna, *“Fractional order systems: modeling and control applications.”* World Scientific, 2011.

[25] Z. Li, “Fractional order modeling and control of multi-input-multi-output processes.” License CC BY-SA 4.0, Jan 01, 2015.

[26] P. C. Krause, O. Wasynczuk, S. D. Sudhoff, and S. Pekarek, *“Analysis of electric machinery and drive systems.”* United States of America ,John Wiley and sons, 2002.

[27] C.-M. Ong, *“Dynamic simulation of electric machinery: using MATLAB/SIMULINK,”* vol. 5. New Jersey: Prentice Hall, 1998.

[28] M. Ahmad, *“High performance AC drives: modelling analysis and control.”* London: Springer, 2010.

[29] S.-K. Sul, *“Control of electric machine drive systems.”* United States of America, John

Wiley & Sons, Inc, 2011.

[30] T. F. Chan and K. Shi, *"Applied intelligent control of induction motor drives,"* First edit. 2011.

[31] R. Krishnan, *"Electric motor drives: modeling, analysis, and control."* Prentice Hall New Jersey, 2001.

[32] S. Goyat and R. K. Ahuja, "Speed control of induction motor using vector or field oriented control," *Int. J. Adv. Eng. Technol.*, vol. 4, no. 1, p. 475, 2012.

[33] B.Basnet, "DSP based implementation of field oriented control of three-phase induction motor drives," *InInternational J. Innov. Eng. Technol.*, vol. 8, no. 2, pp. 65–69, 2017.

[34] L. Umanand and S. R. Bhat, "Online estimation of stator resistance of an induction motor for speed control applications," *IEE Proceedings-Electric Power Appl.*, vol. 142, no. 2, pp. 97–103, 2001.

[35] P. Zhang, B. Lu, and T. G. Habetler, "A remote and sensorless stator winding resistance estimation method for thermal protection of soft-starter-connected induction machines," *IEEE Trans. Ind. Electron.*, vol. 55, no. 10, pp. 3611–3618, 2008.

[36] V. G. BT, "Comparison Between Direct and Indirect Field Oriented Control of Induction Motor," *Int. J. Eng. Trends Technol*, vol. 43, no. 6, pp. 364–369, 2017.

[37] F. A. Hassan and L. J. Rashad, "Particle swarm optimization for adapting fuzzy logic controller of SPWM inverter fed 3-phase IM," *Eng. Tech. J.*, vol. 29, no. 14, pp. 2912–2925, 2011.

[38] F. Bekele, "Study and Implementation of DSP Based Sensorless Speed Control of Induction Motor." Addis Ababa Institute of Technology, Addis Ababa, Ethiopia, 2011.

[39] F. Blaschke, "The principle of field orientation as applied to the new TRANSVECTOR closed loop control system for rotating field machines," *Siemens Rev.*, vol. 39, no. 5, pp. 217–220, 2001.

[40] A. Chourasia, V. Srivastava, A. Choudhary, and S. Praliya, "Comparison study of vector control of induction motor using rotor flux estimation by two different methods," *Int. J. Electron. Electr. Eng.*, vol. 7, no. 3, pp. 201–206, 2014.

[41] A. Mishra and P. Choudhary, "Speed control of an induction motor by using indirect vector control method," *Int. J. Emerg. Technol. Adv. Eng.*, vol. 2, no. 12, pp. 144–150, 2012.

[42] B. Tolunay, “Space Vector Pulse Width Modulationfor Three-Level Converters: a LabVIEW Implementation.” Industrial Technology ICIT, Hong Kong, Februari, 2012.

[43] S. Assefa Sefu Gebreegzabher, “Control Scheme Modeling and Analyzing of Three-phase Inverter for Induction Motor Drive Using SVPWM Technique for Three Wheel Electric Vehicles (Bajaj) Applications.” ASTU, 2019.

[44] E. Simon, “Implementation of a speed field oriented control of a 3-phase pms motor using tm320f240,” 1999.

[45] C. MacLeod, “The synthesis of artificial neural networks using single string evolutionary techniques.” 1999.

[46] R. Krishnan and A. S. Bharadwaj, “A review of parameter sensitivity and adaptation in indirect vector controlled induction motor drive systems,” *IEEE Trans. Power Electron.*, vol. 6, no. 4, pp. 695–703, 1991.

[47] I. Eksin, “An intelligent hybrid fuzzy pid controller,” vol. 5, no. Cd, 2006.

[48] M. Theses and R. Citation, “Analysis of Speed Control Using Hybrid PID-Fuzzy Controller for Induction Motors,” 2015.

[49] G. V Deshpande, “SPEED CONTROL OF INDUCTION MOTORS USING HYBRID PI PLUS FUZZY CONTROLLER,” vol. 6, no. 5, pp. 2253–2261, 2013.

[50] C. a Monje, Y. Q. Chen, B. M. Vinagre, D. Xue, and V. Feliu, *“Fractional-order Systems and Controls. Fundamentals and Applications.”* Springer-Verlag, London Limited, 2010.

[51] A. Oustaloup, P. Melchior, P. Lanusse, O. Cois, and F. Dancla, “The CRONE toolbox for Matlab,” Nov. 2000, pp. 190–195.

[52] I. Podlubny, “Fractional-order systems and PI/sup/spl lambda//D/sup/spl mu//-controllers,” *IEEE Trans. Automat. Contr.*, vol. 44, no. 1, pp. 208–214, 1999.

[53] C. Y. Wang, “Study on fractional order PID controller parameter tuning method and design,” *Jilin Univ.*, 2013.

[54] F. Padula and A. Visioli, “Set-point weighting for fractional PID controllers,” *Proc. Fract. Differ. its Appl.*, 2010.

[55] K. Ogata, *Modern control engineering*, 5th editio. Upper Saddle River, NJ, USA: Pearson, 2009.

[56] Y. Luo, Y. Chen, Y. Luo, and Y. Chen, “Fractional-Order Proportional Derivative Controller Synthesis and Implementation for Hard-Disk-Drive Servo System,” *IEEE*

Trans. Control Syst. Technol., pp. 281--289, 2014.

[57] B. Dandil, “Fuzzy neural network IP controller for robust position control of induction motor drive,” *Expert Syst. Appl.*, vol. 36, no. 3, pp. 4528–4534, 2009.

[58] J. S. R. Jang, “ANFIS: Adaptive-Network-Based Fuzzy Inference System,” *IEEE Trans. Syst. Man Cybern.*, vol. 23, no. 3, pp. 665–685, 1993.

[59] A. Abraham, “Adaptation of fuzzy inference system using neural learning,” in *Fuzzy systems engineering*, Springer, 2005, pp. 53–83.

[60] J. –S. R. Jang., “Self-learning fuzzy controller based on the temporal back-propagation,” vol. 3, no. 5, pp. 714–723, 1992.

[61] J.-S. Jang and C.-T. Sun, “Neuro-fuzzy modeling and control,” *Proc. IEEE*, vol. 83, no. 3, pp. 378–406, 1995.

[62] Y. Tsukamoto, “An approach to fuzzy reasoning method: Advances in Fuzzy Set Theory and Applications,” *North-Holland, Amsterdam*, pp. 137–149, 1979.

[63] C. C. Lee, “Fuzzy Logic in Control Systems: Fuzzy Logic Controller, Part II,” *IEEE Trans. Syst. Man Cybern.*, vol. 20, no. 2, pp. 419–435, 1990.

[64] T. Takagi and M. Sugeno, “Derivation of fuzzy control rules from human operator’s control actions,” in *IFAC Proceedings Series*, Jul. 1984, vol. 16, no. 13, pp. 55–60.

[65] Roland S. Burns, *“Advanced Control Engineering.”* Boston, Johannesburg, Butterworth Heinenann(BH), 2001.

[66] A. Miloudi, E. A. Al-Radadi, and A. DRAOU, “A variable gain PI controller used for speed control of a direct torque neuro fuzzy controlled induction machine drive,” *Turkish J. Electr. Eng. Comput. Sci.*, vol. 15, no. 1, pp. 37–49, 2007.

[67] M. Watany, M. A. Eltantawie, and S. A. Abouel-seoud, “Application of an Adaptive Neuro Fuzzy Inference System for Low Speed Planetary Gearbox Vibration Control,” *Beijing, Beijing Springer-Verlag Berlin Heidelb.*, vol. 34, no. 3, pp. 323–341, 2015.

Reference link

[L1] Reference link for Figure 2.2, Figure 3.8, Figure 3.20 (accessed on January 11, 2021)

1. https://www.enggcyclopedia.com/wp-content/uploads/2012/09/Figure-1-Electric-motor-structure.png
2. https://image.slidesharecdn.com/spvm-140605022805-phpapp01/95/sinusoidal-pwm-and-space-vector-modulation-for-two-level-voltage-source-converter-13-638.jpg?cb=1406352751
3. https://image.slidesharecdn.com/svpwm-150306110056-conversion-gate01/95/advanced-techniques-of-pulse-width-modulation-30-638.jpg?cb=1425639761

Research Fund Acknowledgment

This thesis work is funded by Adama Science and Technology University under the grant number ASTU/SM-R/176/20

APPENDIXES

Appendix A: MATLAB Simulink Models

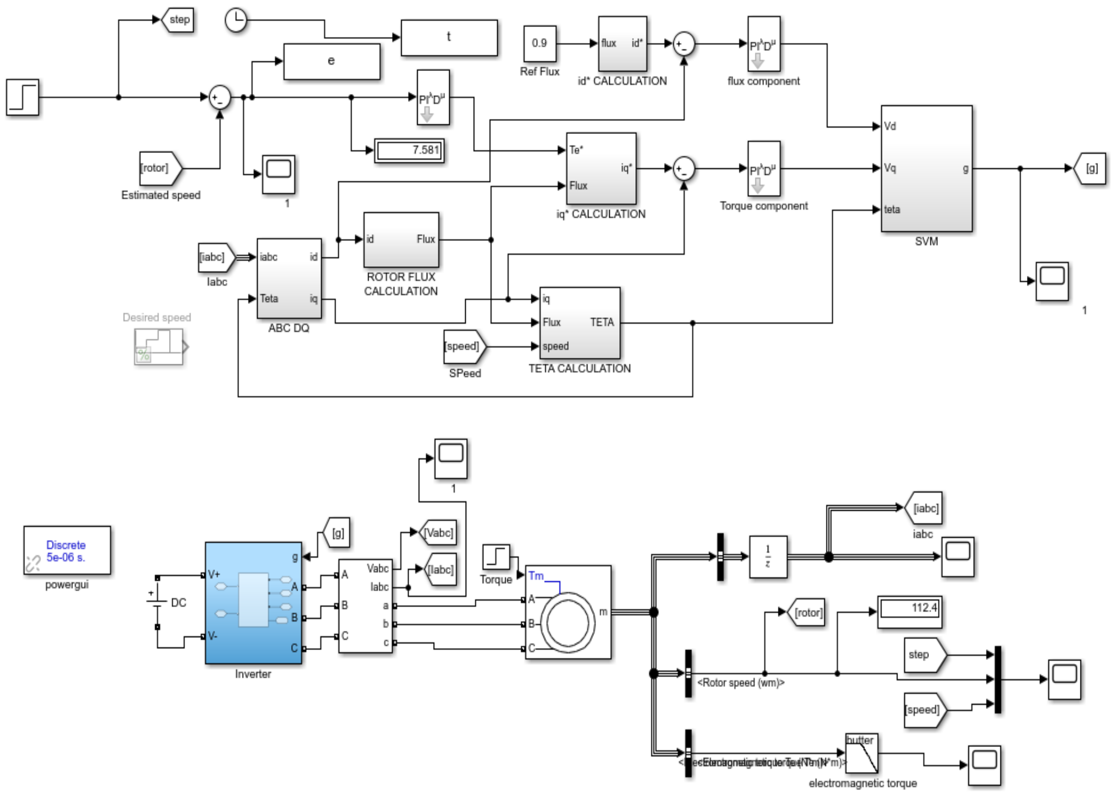
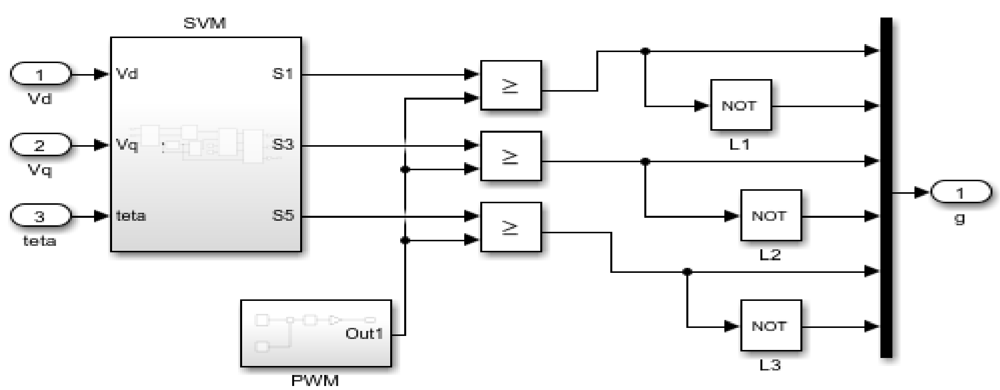


Figure.A.1 Simulink model for FOPID and IFOC.



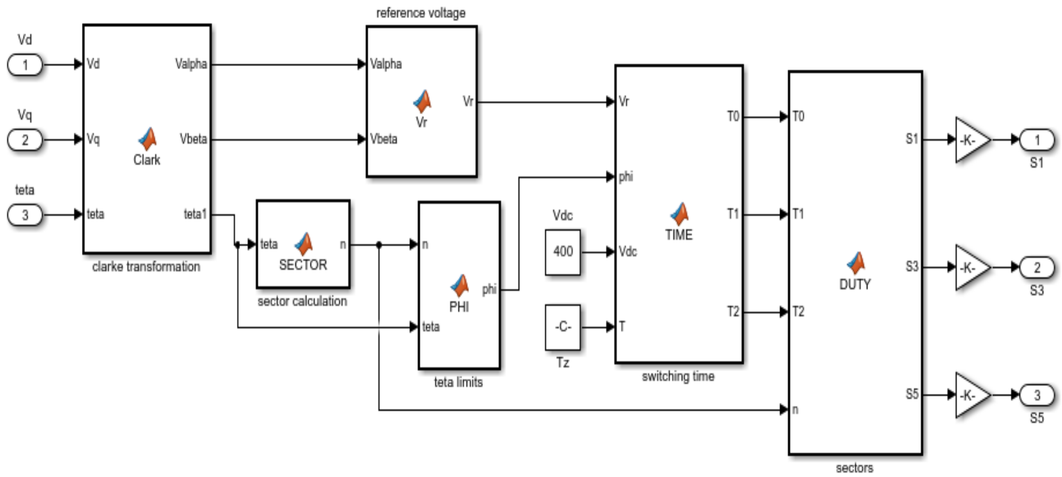


Figure.A.2 Simulink subsystem model for space vector modulation.

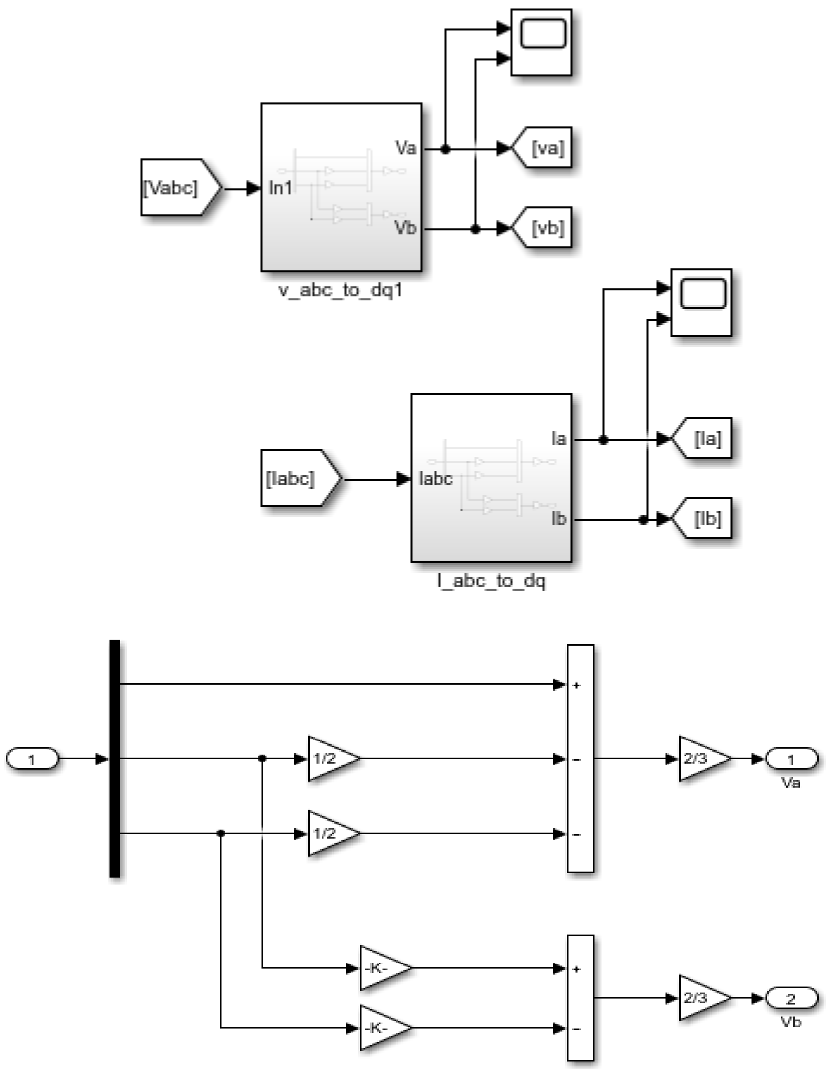


Figure.A.3 Simulink subsystem abc to dq transformation

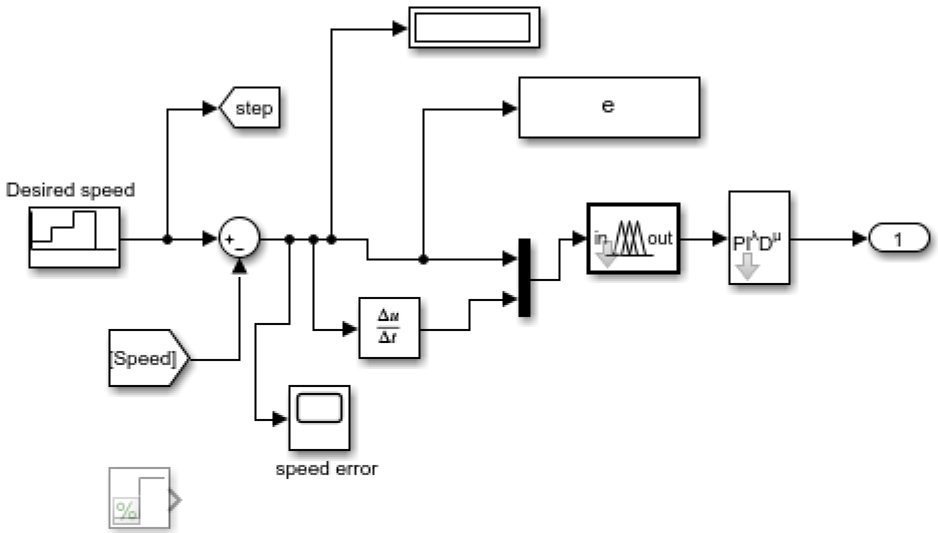


Fig.A.4 Simulink subsystem model for ANFIS controller

Appendix B: Appendixes Related To M-Files for GA Optimization

```
function Tune_model_ind_fopid
clear all
options= optimoptions('ga','PlotFcn',@gaplotbestf);
options.PopulationSize= 30;
options.MaxGenerations= 20;
nvar= 5; % Number of parameters
LB= [0.1 0.1 0.1 0.1 0.1]; % Lower bound of tuning parameters
UB= [5 5 5 5 5]; % Upper bound of tuning parameters
[x,fval]= ga(@EvalObj,nvar,[],[],[],[],LB,UB,[],options)
simulink_model= 'model_ind_fopid';
load_system(simulink_model);
gains= get_param(simulink_model,'modelworkspace');
gains.assignin('kp1', x(1));
gains.assignin('ki1', x(2));
gains.assignin('kd1', x(3));
gains.assignin('lambda1', x(4));
gains.assignin('mu1', x(5));
```

```
function obj= EvalObj(x)
simulink_model= 'model_ind_fopid';
load_system(simulink_model);
gains= get_param(simulink_model,'modelworkspace');
gains.assignin('kp1', x(1));
gains.assignin('ki1', x(2));
gains.assignin('kd1', x(3));
gains.assignin('lambda1', x(4));
gains.assignin('mu1', x(5));
simOut= sim(simulink_model,'SaveOutput','on');
t= simOut.find('t');
e= simOut.find('e');
n= length(e);
ee= abs(e);
obj= 0;
for i= 2:n
   stepsize= t(i)-t(i-1);
   obj= obj+0.5*(ee(i-1)+ee(i))*stepsize; % ITAE
end
```

Appendix C: Appendixes Related To M-Files for SVPWM

C.1 M-file for Determine Valpha, Vbeta, and Angle alpha:

```
function [Valpha, Vbeta, teta1] = Clark(Vd, Vq, teta)
Valpha= Vd*cos(teta)-Vq*sin(teta);
Vbeta= Vd*sin(teta)+Vq*cos(teta);
teta1= atan2(Vbeta, Valpha);
```

C.2 M-file For Determine Gate Signal and Their Sequences

```
function [S1, S3, S5] = DUTY(T0, T1, T2, n)
S1=0;
S3=0;
S5=0;
if (n==1)
```

```
    S1=T1+T2+T0/2;
    S3=T2+T0/2;
    S5=T0/2;
end
if(n==2)
    S1=T1+T0/2;
    S3=T1+T2+T0/2;
    S5=T0/2;
end
if(n==3)
    S1=T0/2;
    S3=T1+T2+T0/2;
    S5=T2+T0/2;
end
if(n==4)
    S1=T0/2;
    S3=T1+T0/2;
    S5=T1+T2+T0/2;
end
if(n==5)
    S1=T2+T0/2;
    S3=T0/2;
    S5=T1+T2+T0/2;
end
if(n==6)
    S1=T1+T2+T0/2;
    S3=T0/2;
    S5=T1+T0/2;
end
```

C.3 M-file for sector selection

```
function phi = PHI(n, teta)
```

```
phi=0;
if(n==1)
   phi=teta;
end
if(n==2)
   phi=teta-pi/3;
end
if(n==3)
   phi=teta-2*pi/3;
end
if(n==6)
   phi=teta+pi/3;
end
if(n==5)
   phi=teta+2*pi/3;
end
if(n==4)
   phi=teta+pi;
end
```

C.4 M-file for Determine Vref

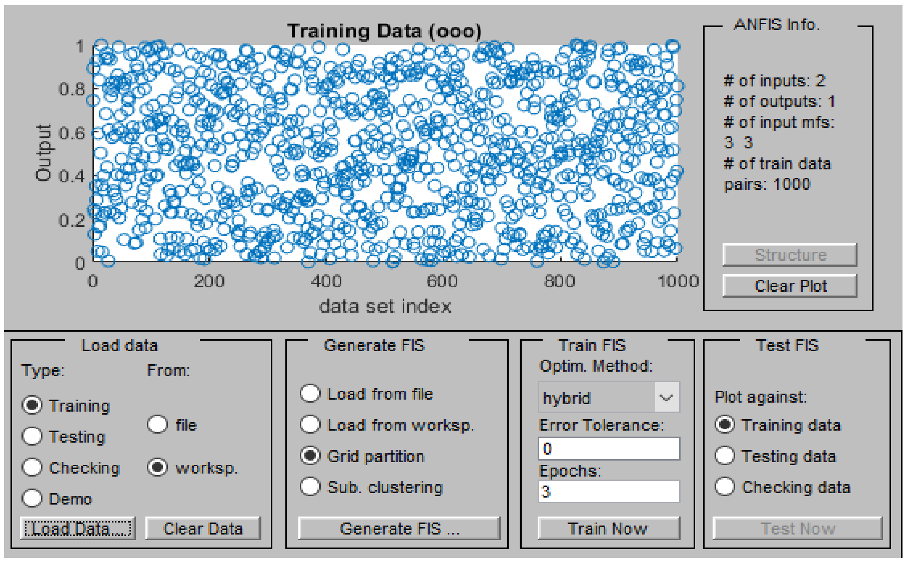
```
function Vr = Vr(Valpha, Vbeta)
Vr=0;
Vr=sqrt(Valpha^2+Vbeta^2);
```

C.5 M-file for Determine Duration Time for Each Sectors

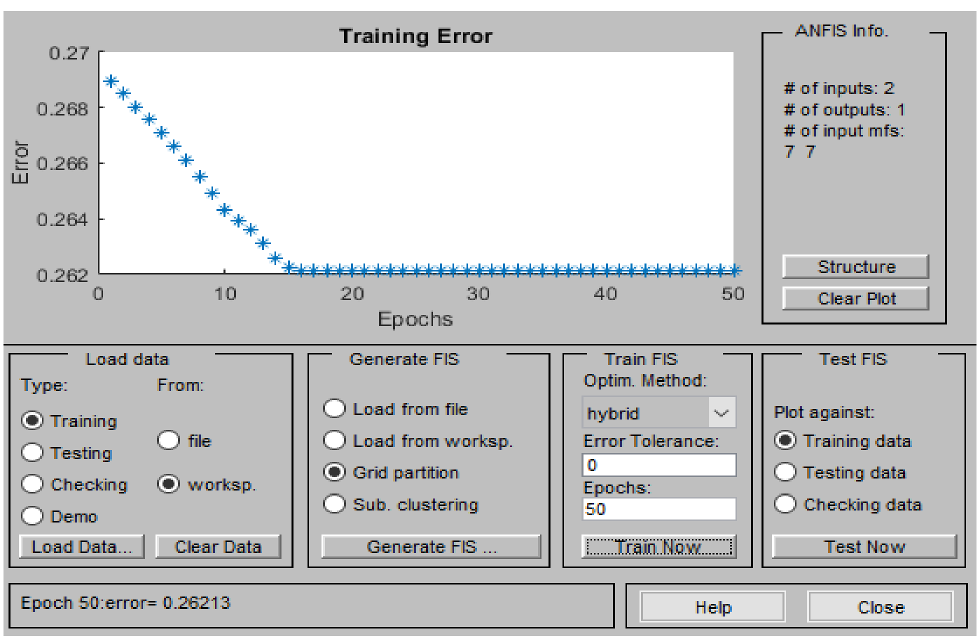
```
function [T0, T1, T2] = TIME(Vr,phi,Vdc, T)
a=sqrt(3)*(Vr/Vdc);
T1=T*a*sin(pi/3-phi);
T2=T*a*sin(phi);
T0=T-T1-T2;
```

Appendix D: ANFIS Membership Functions, Rule, and Surface Viewers

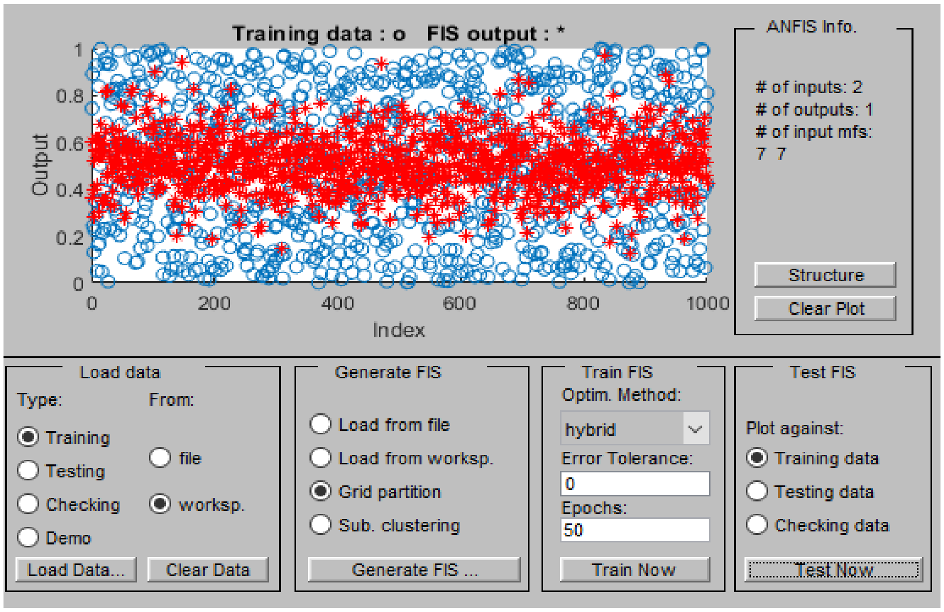
Membership functions for input and output variables have been chosen with generalized bell shaped and linear respectively .Universe of discourse of input and output variables are divided in to seven fuzzy sets. Those are LN (Large Negative), MN (Medium Negative), SN (Small Negative), Z (Zero), LP (Large Positive), MP (Medium Positive), SP (Small Positive).



a)

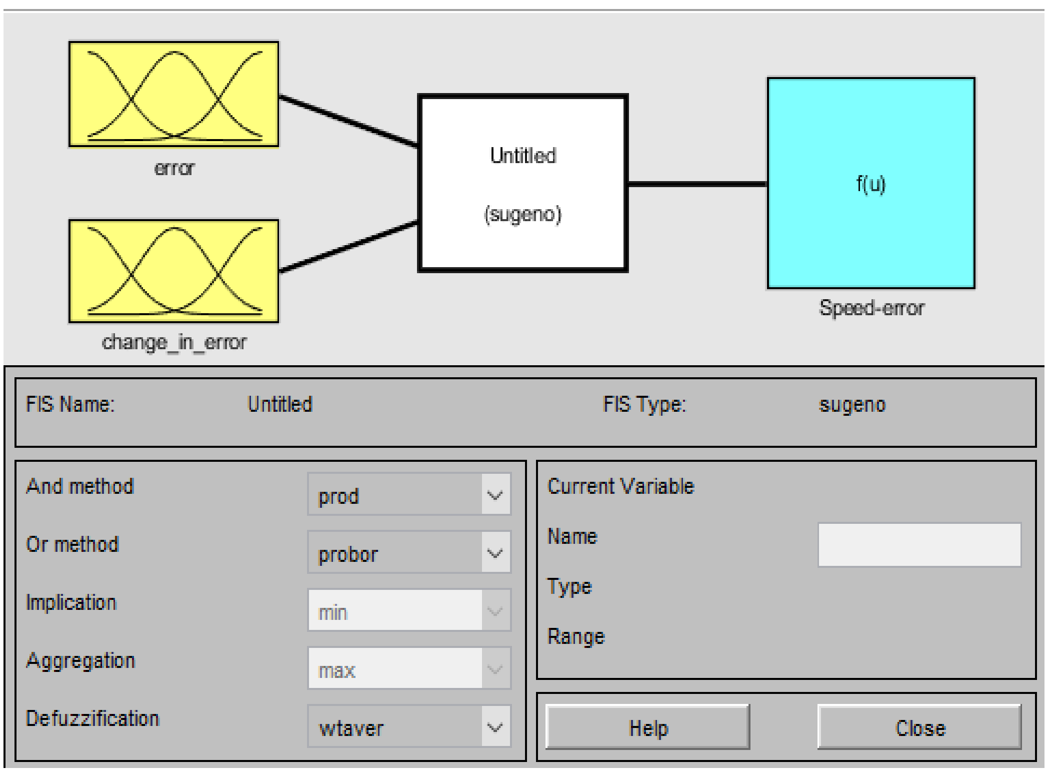


b)

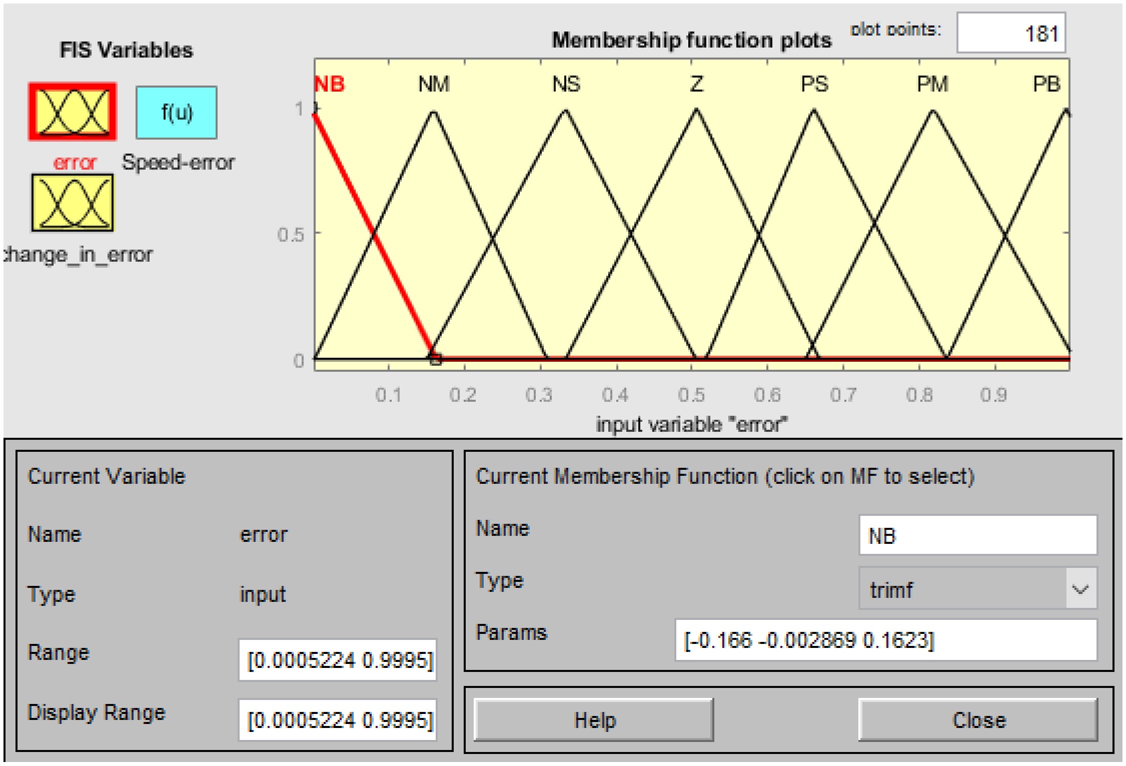


c)

Figure D.1 a) Training data sets b) Training error c) training output data

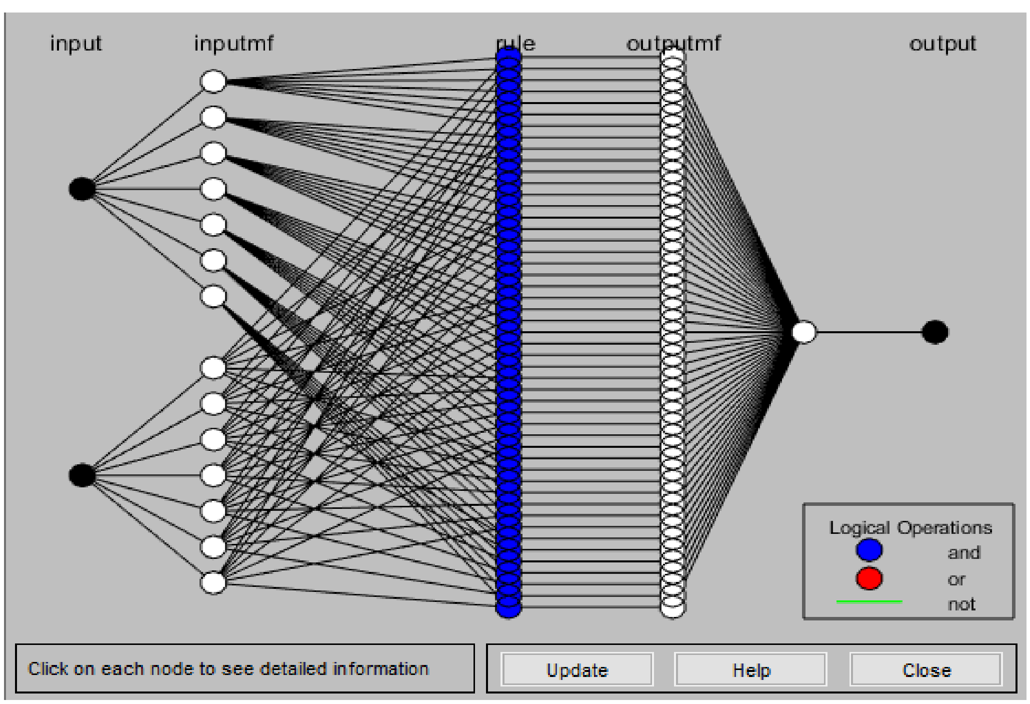


a)

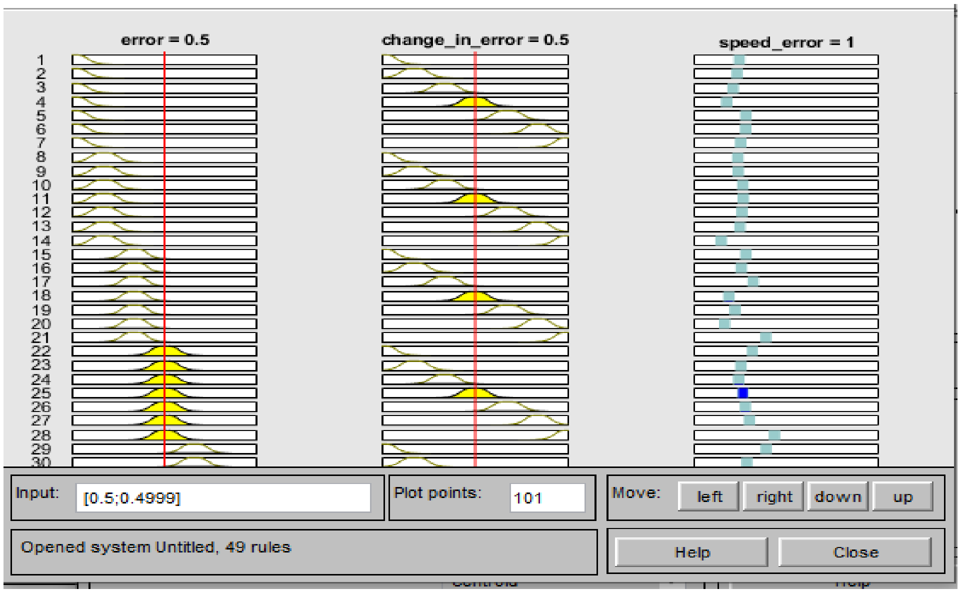


b)

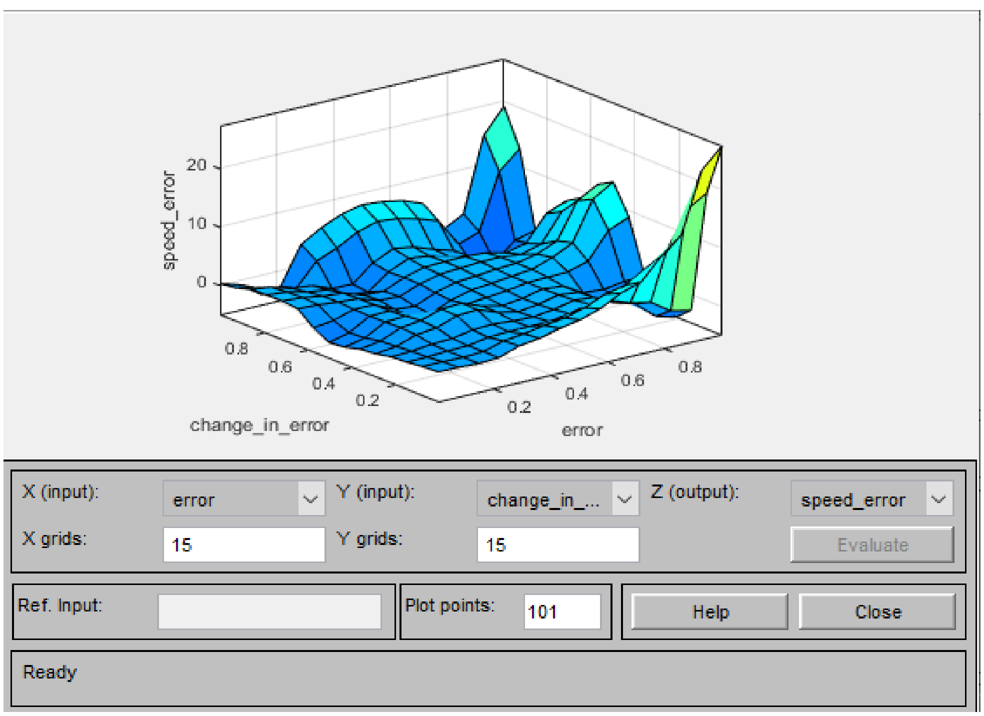
Figure D.2 a) FIS property editor b) Error MFs



a)



b)



c)

Figure D.3 a) Model structure b) rule viewer c) Surface viewer

Appendix E: Induction Motor Parameters

Table E.1: Parameters of Squirrel-cage induction motor used for simulation

Parameters	Symbol	Value
Types of rotor	-	Squirrel cage
Number of pole pair	P	2
Frequency (Hz)	F	50
Stator resistance(Ω)	R_s	0.6837
Rotor resistance(Ω)	R_r	0.351
Stator inductance(H)	l_s	0.004125
Rotor inductance(H)	L_r	0.004125
Mutual inductance(H)	L_m	0.001486
Moment of inertia(kgm2)	J	0.01
Rated voltage(v)	V_r	400
Rated speed(rad/s)	ω_r	120
Rated power(KW)	P_r	7.666
PWM frequency(kHz)	-	2
Load torque(N*m)	T_L	4
Friction factor (Nms-1)	F	0.00061