SENSOR-LESS FIELD ORIENTED SPEED CONTROL OF INTERIOR PERMANENT MAGNET SYNCHRONOUS MOTOR DRIVES USING MRAS OBSERVER AND NEURAL ADAPTIVE CONTROLLER

M.Sc THESIS

BY

GECHANU ABEBE DURBE

HAWASSA UNIVERSITY, HAWASSA, ETHIOPIA

AUGUST, 2020

SENSOR-LESS FIELD ORIENTED SPEED CONTROL OF INTERIOR PERMANENT MAGNET SYNCHRONOUS MOTOR DRIVES USING MRAS OBSERVER AND NEURAL ADAPTIVE CONTROLLER

GECHANU ABEBE DURBE

A THESIS SUBMITTED TO

DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING

HAWASSA INSTITUTE OF TECHNOLOGY

SCHOOL OF GRADUATE STUDIES

HAWASSA UNIVERSITY

IN PARTIAL FULFILLMENT OF THE

REQUIREMENTS FOR THE

DEGREE OF

MASTER OF SCIENCE IN ELECTRICAL AND COMPUTER ENGINEERING

(CONTROL AND INSTRUMENTATION ENGINEERING)

HAWASSA, ETHIOPIA

AUGUST, 2020

EXAMINERS' APPROVAL SHEET

We, the undersigned, members of the Board of Examiners of the final open defense by **Mr. Gechanu Abebe** Id. No: **PGCon/019/09**, have read and evaluated his/her thesis entitled **"Sensor-Less Field Oriented Speed Control of Interior Permanent Magnet Synchronous Motor drives using -MRAS Observer and Neural Adaptive controller"**, and examined the candidate. This is, therefore, to certify that the thesis has been accepted in partial fulfillment of the requirements for the degree.

Mr.Muluken Regassa		Aug 9/2020
Name of Chair Person	Signature	Date
Dr.-Ing. Mengesha Mamo		August 7/2020
Name of Major Advisor	Signature	Date
Mr.Yeshitla Hailu		07/08/2020
Name of Internal Examiner	Signature	Date
Eskinder Antench Lt/col. Dr. Defence University, College of Engineering, Academic Dean		04/08/20
Name of External Examiner	Signature	Date
SGS Approval	Signature	Date

Declaration

I hereby declare that this MSc thesis entitled **"Sensor-Less Field Oriented Speed Control of Interior Permanent Magnet Synchronous Motor Drives Using MRAS Observer and Neural Adaptive Controller"** is my original work and has not been presented for a degree in any other university, and will not be presented by me to any other university for similar or any other degree award, and all sources of material used for this thesis have been duly acknowledged.

Name: Gechanu Abebe

Signature: _____

This M.Sc thesis entitled **"Sensor-Less Field Oriented Speed Control of Interior Permanent Magnet Synchronous Motor Drives Using MRAS Observer and Neural Adaptive controller"** has been submitted for examination with my approval as thesis advisor.

Name: Dr.-Ing.Mengesha Mamo

Signature: _____

Place and Date of Submission: _____

ACKNOWLEDGMENTS

First and foremost I would like to thank the almighty God and his mother for helping me to achieve my goal and to complete this thesis successfully. This work will remain as a great honor of my life and my heart is filled with nothing but gratefulness to all of you. And then, I would like to thank University of Hawassa for giving the chance to study my master of degree and supporting me financially. I would like to express my deepest thanks and gratitude to my advisor Dr.-Ing.Mengesha Mamo, for guidance and support.

I am also thankful to Mr.Mesfin Fanuel, Head of Electrical and Computer Engineering for his suggestions and help to this work in all possible aspects.

My thanks are extended to my fellow colleagues in Electrical Engineering, especially in the control engineering stream, Mr.Muluken Regassa and Mr.Yeshitla Hailu, who built an academic and very friendly study environment that made my study at the university most enjoyable and skillful.

Finally, I am deeply indebted to my parents for their support and concerning, which encourages and strengthened me to accomplish this program.

Tables of Contents

List of Figures

List of Tables

List of Abbreviations

PMSM	Permanent Magnet Synchronous Motor
IPMSM	Interior Permanent Magnet Synchronous Motor
SPM	Surface-Mounted Permanent Magnet Motor
SVPWM	Space-Vector Pulse Width Modulation
VC	Vector Control
VSI	Voltage Source Inverter
MRAS	Model Reference Adaptive System
ANN	Artificial Neural Networks
NNC	Neural network control
PWM	Pulse Width Modulation
PI	Proportional Integral
FOC	Field-oriented control
MO	Magnitude Optimum
SO	Symmetric Optimum
AC	Alternating current
ADC	Analog to Digital Converter
AM	Adjustable model
APIC	Adaptation proportional integral controller
IGBT	Insulated-gate bipolar transistor
RBFNN	Radial Basis Function Neural Network

ABSTRACT

In this thesis, develops a Model Reference Adaptive System (MRAS) is used to estimate speed for interior permanent magnet synchronous motor (PMSM) using Neural Network (ANN) .The algorithm is based on field oriented control theory. The proposed control scheme was a model reference adaptive system (MRAS) speed observer based on stator currents with space vector pulse width modulation technique.

The objective of the proposed sensor-less control is to improve the speed control performance and robustness of PMSM drive under load variations. For speed estimation performed at low speed, pi-radial basis function neural network observer is proposed instead of the traditionally used pi-controller in adaptation mechanism of the MRAS.In this work, the traditionally used pi-controller in adaptation mechanism of the MRAS observer is replaced by pi-radial basis function neural network observer. The improvement in Speed response during load variations is shown by comparative study between conventional PI and neural PI. Simulation result showed the performance of speed estimator under various operating conditions using MATLAB/SIMULINK environment.

CHAPTER ONE

1. Introduction

1.1 Background of the study

Nowadays, the permanent-magnet synchronous motor has emerged as an alternative to induction motor due to the increasing energy saving demand. Permanent magnet synchronous motor drives has been widely used in high performance applications such as industrial drives and robots, due to outstanding features such as high speed, high-energy efficiency, and high torque-to-inertia ratio, high power factor, fast response and rugged construction. These qualities render permanent magnet synchronous motor (PMSM) ,it has increasingly used in, washing machines, electrical vehicles, aircraft ,robotic automation, escalators and industrial servo drives [1]. The control methods used for the PMSM are: V/f control, field oriented and direct torque control, it is important to know the information about the rotor speed and position [2].

The interior PMSM is primarily associated with high-performance motor control and fed by a voltage source inverter using a space vector pulse-width modulation technique. The presence within the electrical drives with sensors represents a disadvantage under an economical aspect because; it decide cost increase, temperature sensitivity and reliability decrease. Field oriented control, type of the control methods of the PMSM, which achieve fast dynamic response and the flexibility normally obtained in control of separately excited dc motor.

Position sensor with higher quality is a necessary component part of the drive system employed in industrial and automotive applications. But the high cost and strict requirement extremely limit the application in the drive system. Moreover, there are some applications, where there is no room to put the speed sensor or the nature of the environment does not allow the use of any additional speed sensor. So that, it is highly desired to develop sensor less technology and elimination of the speed encoder is highly encouraged to increase the mechanical robustness of the system [3]. The sensor less technology can make drive cheaper and also increase the reliability of the drive system. In [4]adopts the modern reference adaptive control scheme, which uses the PMSM itself as the reference model to estimate the speed of the motor. An ANN was configured for a same application, speed estimation by a learning process to analysis the dynamic performances.

1.2. The statement of problem

Speed control of traditional PMSM drive systems is obtained by taking rotor position or speed information from shaft sensor such as an optical encoder or resolver. The use of these types of sensors increases system complexity, weight and the cost. To overcome these problems, to considerable interest in sensor less vector control of a PMSM drives. A sensor less control techniques, have been developed to eliminate the rotor position sensor in PMSM applications to improves system reliability, robustness and the dynamic performance.

The algorithms for estimating the rotor speeds based on rotor flux observers are complicated and parameters dependent. The designing procedure of speed controllers will be very difficult if a complex mathematical model of PMSM is used. The model reference adaptive systems like a robust controllers, it have good dynamic and static performance even, structured and unstructured uncertainties appear. Therefore, this model can be designed by using simplified models of PMSM. In this thesis presents a MRAS method, based on the information taken from measurement of stator currents. One of the key issues to constitute a high quality adaptive control system are the implementation of adaptive law in the adaptive mechanism that mean the problem using MRAS to estimate the speed or identify parameters is the construction of adaptive law. There are optimal design methods of local parameters, the Lyaunov Stability Theory and the Popov stability design method used for the design of the adaptive law, which are complex, therefore very difficult to obtain the wide spread promotion in reality.

1.3. Objectives of the thesis

1.3.1General objective

To design and analysis Sensor less field oriented speed control of Interior Permanent Magnet Synchronous Motor (IPMSM) drive with Artificial Neural Network (ANN) based MRAS observer. It is expected that this control scheme can track the reference speed well under the conditions of the variation of speed command and load torque.

1.3.2. Specific objectives

The specific objectives of the thesis are:

- ✓ To study the mathematical model of PMSM for saliency drive system.
- ✓ To model PMSM based on stationary and rotor reference frame.
- ✓ To design PMSM drive system using field oriented control scheme.
- ✓ Estimated rotor speed using ANN based MRAS.
- ✓ To develop Neural networks for optimal PI parameters tuning to ensure stability.
- ✓ To compare the speed estimation performance of the MRAS observer for motor parameter variation.

1.4. Literature review

In recent years, an artificial neural network has been used in complex process control, and has attracted widespread attention [5]- [6].The performance of the controller declines, when the dynamics of the system changes. In order to overcome this problem, developed Artificial Neural Network will be used in place of PI controller. When the motor is run speed and the load parameters are unknown, ANN to achieve accurate control of the speed. The neural network adaptive PID control scheme which is locally approximated by the RBF network, based on concept of this control used for pi-controller, adopted in this thesis [7].To replace the PI controller a new ANN controller is feed forward neural network implemented [8].The novel method based on the documents [9] [10] proposed the estimated speed based on MRAS and RBFN, which estimates the position and speed for PMSM sensor less vector control system. The Proposed method adopts PMSM itself as a reference model and the PMSM current model as the adjustable model. The error signal which is difference between q-axis estimated current and q-axis actual current fed into the radial propagation feed forward network regulator to obtain estimated speed. In this thesis, error signal take the difference between estimated current and actual current from adaptation mechanism. In [11]- [12],the basic theoretical knowledge of MRAS theory for estimating PMSM speed was given, and the q -d axis equation of stator current using for adjustable model was described. The adaptive law was constructed by Popov stability theory. These establish the theoretical foundation for this thesis. In [13], used MRAS made sensor-less control in the case of load changes. This fully demonstrates the good robustness of the MRAS. To adjust the parameters of the PI regulator using fuzzy controller in the speed observer based on MRAS [14].The whole system had a good dynamic and steady performance in a large speed range. In [15]- [16], constructed adaptive law by Popov and Lyapunov stability analysis method for parameter identification, and made a contrast, but the speed was also acquired by sensor. In [17]- [18], they established different adaptive laws according to different stability of the system [19]. Popov stability criterion can consider the stability and the design of adaptive law can be dictated by its design principles. The stability is analyzed through modern control theory and the transfer function of the speed observer is constructed. The PI regulator in the speed observer can be used as the adjustable link, and use root locus analysis to adjust the ratio of PI and avoid the complex algorithm of fuzzy PI in [14].

1.5. The Methodology

The following methodologies have been used for the accomplishment of this thesis:

- Different works of literatures studied the theoretical information regarding to sensor less speed control PMSM drive which includes the control properties of field oriented control schemes and a comparison of previous work in the related field is presented.
- The PI controller parameters are calculated by finding the transfer function of the controller and then setting the appropriate gains from the root locus and bode plot to achieve fast adaptive ,current and speed loop which is independent of the load torque variation.
- Adaptive controller design using Popov stability criteria and mathematical modeling for a nonlinear system for interior PMSM and the estimation of rotor speed is based on currents of stator that are measurable state variables.
- Design the speed estimation schemes based on neural network based MRAS observer.
- MATLAB/Simulink Simulation has been utilized for comparison of tracking performance for both conventional MRAS and neural network based adaptive observer.
- Finally, formulate the conclusion and future of the thesis based on the Simulink result.

CHAPTER TWO

2. Theoretical review of permanent magnet motors drives

2.1 INTRODUCTION

A Permanent Magnet Synchronous Motor (PMSM) is a synchronous motor that uses permanent magnets for the rotor to produce the air gap magnetic field rather than using electromagnets .The permanent magnet synchronous motor is cross between an induction motor and brushless DC motor. Like a brushless DC motor, it has a permanent magnet rotor and windings on the stator. However, the stator structure with windings constructed to produce a sinusoidal flux density in the air-gap of the machine resembles that of an induction motor. Motors are electromagnet devices used to convert electrical energy into useful mechanical work. There are two major classifications of ac motors. The first is induction motors that are electrically connected to power source. Through electromagnetic coupling, the rotor and the stator fields interact, creating rotation without any other power source.

The second is synchronous motors that have fixed stator windings that are electrically connected to the ac supply with a separate source of excitation connected to field windings when the motor is operating at synchronous speed. The stator of a PM synchronous motor has conventional three phase windings. In the rotor PM materials have the same function of the field winding in a conventional synchronous machine. Their development was possible by the introduction of new magnetic materials, like the rare earth materials. PM-Synchronous Motors offer a number of advantages in designing modern motion control systems. [20]

Today, these motors are designed to be more powerful while also having a lower mass and lower moment of inertia. Hence, they are commonly used in industrial automation for traction, robotics or aerospace, where greater power and heightened intelligence are required.

2.1.1 Classification of Permanent Magnet Synchronous Motor

Based on the directions of the field flux and the placement of the magnets, permanent magnet synchronous motor is classified as discussed below:

A. Magnetization of Permanent Magnet

PM motors can be classified by the magnetized with certain orientation or direction of PMs such as radial and parallel magnetization. The radial magnetization is along the radius of rotor while the parallel magnetization is parallel to the edges of rotor. The magnetization orientation strongly influences the quality of the air gap flux density distribution and indirectly affects the power density in a given arrangement of the machine with PMs. Both of magnetization orientation are prevalent in practice whereas other forms of magnetization are yet to make their presence felt even when they have been known to possess unique advantages in some cases.

B).Direction of Field Flux and Flux Density Distribution

PMSM and PM brushless motors [21]- [22], are broadly classified depending on the direction of field flux. In radial field, the flux direction is along the radius of the machine .The radial field permanent magnet motors are the most commonly used. In axial field, the flux direction is parallel to the rotor shaft. PM motors are classified on the basis of the flux density distribution and the shape of current excitation. The PMSM has a sinusoidal-shaped back EMF and is designed to develop sinusoidal back EMF waveforms.

C).Permanent Magnet Radial Field Motors

In PMSMs, the magnets can be placed in different ways on the rotor, depending on the placement of the magnets. They are called either as Surface mounted (SM) or Interior permanent magnet synchronous motor. SMPM motors, a PM mounted on the surface of the rotor as shown in fig.2.1. The outer periphery of rotor laminations arrangement provides the highest air gap flux density as it directly faces the air gap without the interruption of any other medium. Surface mounted are again of two types, magnet projects from the surface of the rotor is called projecting type and, its inserted into the rotor is called inset type, to provides a smooth rotor surface. [22]

Drawbacks of such an arrangement are lower structural integrity and mechanical robustness as they are not tightly fitted into the rotor laminations to their entire thickness. At low speed used

this applications because of the limitation that the magnets will fly apart during high-speed operations. These motors are considered to have small saliency, thus having practically equal inductances in both quadrature and direct axes.

Interior PM Motors have interior mounted permanent magnet rotor. Each permanent magnet is mounted inside the rotor. In this configuration saliency is available and the air gap of d-axis is greater compared with the q axis gap resulting that the q-axis inductance has a different value than the d -axis inductance. By designing a rotor magnetic circuit such that the inductance varies, as a function of rotor angle, the reluctance torque can be produced in addition to the mutual reaction torque of synchronous motors. These motors to concern have saliency with q-axis inductance greater than the d-axis inductance. The interior PM rotor construction is mechanically robust and therefore suited for high-speed operation such as PCB manufacturing, spindle drives and hybrid electric vehicles (HEV) etc.

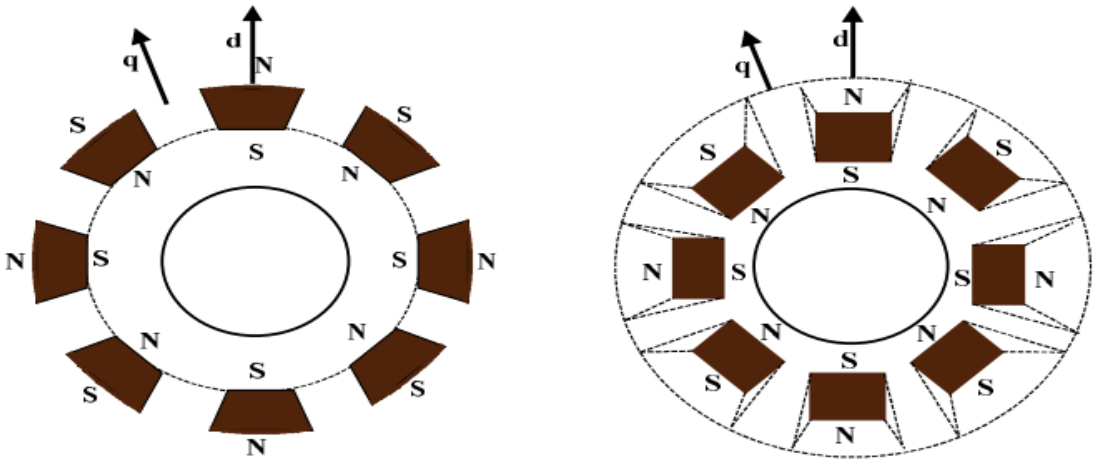


Figure.2.1 Surface PM (SPM) and Interior PM (IP) Synchronous Machine

2.2. Modeling of Interior Permanent Magnet Synchronous Motor

Under this topic, detailed modeling of PM motor drive system is required for proper simulation of the system. For simplicity, the model of PMSM without damper winding has been developed on rotor reference frame using the following assumptions: - the stator core saturation is ignored, Eddy currents and hysteresis-losses are negligible and the induced -EMF is sinusoidal. [23]

First, it's necessary to know the rotor position. During the alignment, the rotor is aligned with axis of phase A, aligned with the direct (flux producing) axis show in fig 2.2. In this state, the rotor position is set to zero (required voltage in d-axis and rotor position is set to zero, static voltage vector, which causes that rotor attracted by stator magnetic field and to align direct axis).
The common analysis of permanent magnet synchronous motor is d-q axis mathematical model as discussed in chapter 2. It can be used to analyze not only the permanent magnet synchronous motor steady state operating characteristics, but also to analyze the transient performance of the motor.

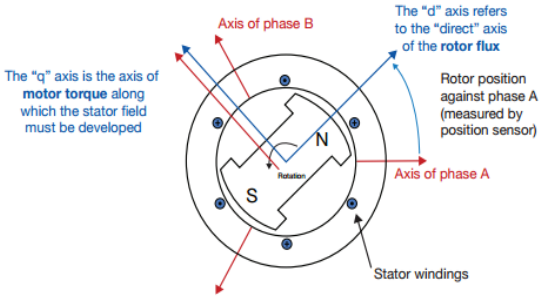


Figure 2.2: Rotor position

2.2.1 Co-ordinate Transformations

The dynamic model of the three phase AC machines are characterized by the voltage equations, the flux linkage equations, and the electromagnetic torque. The inductances is time dependent. Accordingly, the variables of the AC machines model are time varying, as long as the rotor is rotating. Hence, Park and Clarke Transformation are necessary to reduce the complexity of the dynamic model.

2.2.1.1. Park and Inverse Park Transformation

The Park's transformation is a well-known transformation that converts the quantities into two phase synchronously rotating frame.
The forward and the inverse Park transformation are given as:

$$\begin{cases} [f_{dpo}] = [T_{dpo(\theta)}][f_{abc}] \\ [f_{abc}] = [T_{dpo(\theta)}][f_{dpo}] \end{cases} \quad (2.1)$$

Where, f generic variable which can be current, voltage, flux linkage etc. T is the transformation matrix and T^{-1} is the inverse transformation matrix.

Where the d-q-0 transformation matrix is defined as:

$$[T_{dpo(\theta_d)}] = \frac{2}{3}\begin{bmatrix} \cos(\theta_d) & \cos\left(\theta_d - \frac{2\pi}{3}\right) & \cos\left(\theta_d + \frac{2\pi}{3}\right) \\ -\sin(\theta_d) & -\sin(\theta_d - \frac{2\pi}{3}) & -\sin(\theta_d + \frac{2\pi}{3}) \\ \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \end{bmatrix} \quad (2.2)$$

And the inverse is given by:

$$[T_{dpo(\theta_d)}]^{-1} = \frac{2}{3}\begin{bmatrix} \cos(\theta_d) & -\sin(\theta_d) & 1 \\ \cos\left(\theta_d - \frac{2\pi}{3}\right) & -\sin(\theta_d - \frac{2\pi}{3}) & 1 \\ \cos\left(\theta_d + \frac{2\pi}{3}\right) & -\sin\left(\theta_d + \frac{2\pi}{3}\right) & 1 \end{bmatrix} \quad (2.3)$$

Where, θ_d is the transformation angle, the positive q-axis is defined as leading the positive d-axis by 90^0 in the original Park's transformation.

2.2.2 d-q axis mathematical model of IPMSM

The PMSM is modeled in the d-q reference frame fixed rotor is described by the following equations.

A). Electrical equations of the IPMSM

The electrical equations of the PM synchronous motor can be described in the rotor rotating reference frame**,** written in the (d-q axis) rotor flux reference frame are described as follows: [24]

- Voltage equations with repect to rotor axis are given by:

$$\begin{cases} V_q = R_s i_q + \frac{d}{dt}\lambda_q + \omega_r \lambda_d \\ V_d = R_s i_d + \frac{d}{dt}\lambda_d - \omega_r \lambda_q \end{cases} \quad (2.4)$$

- The d-q-axis stator flux linkages are given by:

$$\begin{cases} \lambda_d = L_d i_d + \lambda_{PM} \\ \lambda_q = L_q i_q \end{cases} \quad (2.5)$$

- Substituting equations (2.5) into (2.4), the following equation is obtained.

$$\begin{cases} V_d = R_s i_d + L_d \dfrac{d}{dt} i_d - L_q \omega_r i_q \\ V_q = R_s i_q + L_q \dfrac{d}{dt} i_q + \omega_r (L_d i_d + \lambda_{PM}) \end{cases} \quad (2.6)$$

- Stator current equations in *d -q* axis of the machine can be written as:

$$\begin{cases} \dfrac{di_d}{dt} = -\dfrac{R_s}{L_d} i_d + \dfrac{L_q}{L_d} \omega_r i_d + \dfrac{1}{L_d} u_d \\ \dfrac{di_q}{dt} = -\dfrac{R_s}{L_q} i_q - \dfrac{L_d}{L_q} \omega_r i_d + \dfrac{1}{L_q} u_d + \dfrac{\lambda_{PM}}{L_q} \omega_r \end{cases} \quad (2.7)$$

Where, V_d ,V_q,i_d and $i_q$the stator voltages and currents; R_s,L_d and L_q the stator resistance and inductances,$\omega_r$is motor electrical speed and $\lambda_{PM}$is the permanent magnet flux linkage (denoted by pm in Simulink model).

The developed electromagnetic torque motor is written as [25]

$$T_e = \frac{3}{2}\frac{p}{2}([L_d - L_q] i_d i_q + \lambda_{PM} i_q) \quad (2.8)$$

Thus produced torque is composed of two distinct mechanisms. The first term of (2.8) corresponds to "the mutual reaction (excitation) torque" occurring between $i_q$and the permanent magnet flux λ_{PM} while the second term corresponds to "the reluctance torque" due to the differences in d-axis and q-axis reluctance. If the permanent magnets are mounted on the rotor surface, and there is no significant internal asymmetry in the iron parts of the rotor, the direct-axis and quadrature-axis inductances of the machine are approximately equal, $L_d = L_q$ to expression of the electromagnetic torque function becomes (2.9). [2]

$$T_e = \frac{3}{2}\frac{p}{2}\lambda_{PM} i_q \quad (2.9)$$

Where, T_e is the electromagnetic torque [Nm] and *p* is the number of poles.

B).Mechanical equation of the IPMSM

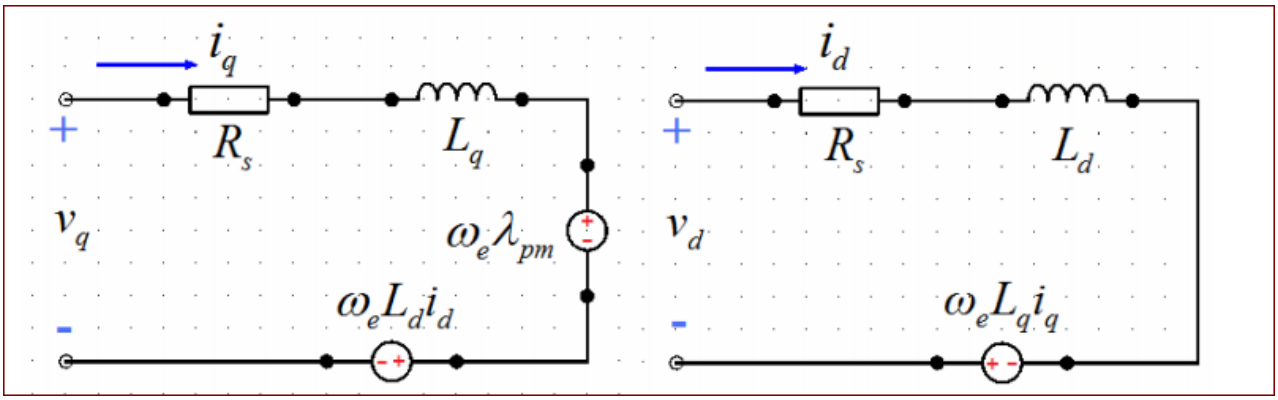
The torque that is generated by the energy conversion process is used to drive mechanical loads. Its expression is related to mechanical parameters via the fundamental law of the dynamics as follows:

$$\begin{cases} T_m = \boldsymbol{T_L} + \mathrm{B}\omega_m + \mathbf{J}\,\dfrac{d\omega_m}{dt} \\ \dfrac{d\omega_m}{dt} = \dfrac{1}{J}(T_m - \boldsymbol{T_L} - \mathrm{F}\omega_m) \\ \omega_m = \dfrac{2}{p}\omega_r \end{cases} \qquad (2.10)$$

Where J is a moment of inertia [kg ∗m2]; B is the viscous friction constant [Nm. Sec/rad]; T_L is the load torque [Nm]; ω_r is the rotor electrical speed;$\omega_m$is the rotor mechanical speed.

2.2.3. Equivalent Circuits of the PMSM

From the dynamic equation (2.4) the equivalent circuit of the PMSM can be derived for the stator q-axis and d-axis coordinates. During steady state operation, the *d-q* axis currents are constant quantities. Hence the dynamic equivalent circuit can be reduced to the steady state circuit shown in Figure 2.3.



(a) q-axis circuit (b) d-axis circuit

Figure 2.3: Equivalent circuit model of PMSM in the rotor reference frame

CHAPTER THREE

3. FIELD ORIENTED CONTROL (FOC) OF PMSM

3.1. Introduction

Field oriented control of PMSM is one important variation of vector control methods. The basic idea of the field oriented control algorithm classified motor stator current into a magnetic field-generating part and a torque-generating part, represented by a vector, in a rotating reference frame (with a d-q coordinate system).The implementation of this technique will be carried out using two current regulators, one for the direct-axis component and another for the quad-axis component, and one speed regulator.

The principle with field-oriented control is that it controls the motor current in d-q-frame, instead of directly in a-b-c frame, which maintains a 90 degree electrical angle between rotor and stator-field components. The benefit with this is that since the d-q- frame rotates together with the rotor; the electrical state of the motor does not depend on the rotor position.

The goal of field oriented control on the synchronous and asynchronous machine is to separately control the torque producing and magnetizing the flux components. Field-oriented control (FOC), known like as vector control or decoupling control aims to control effectively the motor torque and flux in order to force the motor to accurately track the command trajectory regardless of the machine and load parameter variation or any extraneous disturbances [26].In FOC ,the decoupling control strategy for PMSM and applied the control methods such as PI, FLC and neural network on the control frame of FOC scheme.

3.2. Field Oriented Control (FOC) control properties

Field Oriented Control is the technique used to achieve the decoupled control of torque and flux by transforming the stator current quantities (phase currents) from stationary reference frame to torque and flux producing currents components in rotating reference frame. A flux is controlled through the d-axis current while the torque is controlled through the q-axis current.

It is not enough to use only the d-q-axis transformation, to achieve decoupled control of flux and torque since there is coupling between the two axes which may be canceled out by subtracting

from the d-q-axis reference voltages the appropriate coupling terms. Two types of field oriented control are possible for the PMSM: rotor oriented FOC and stator oriented FOC. Due to its advantages like low torque and current ripple, constant VSI switching frequency, low audible noise FOC control scheme is chosen for further analysis [27]. The control property block has as input the reference torque and output the d-q axis reference currents. The most common control properties are discussed. [2]

A. Unity power factor strategy

In order to control the d-q axis currents, under this control strategy there is no phase different between the current vector and the Voltage vector. Since the power factor cos (θ) = 1 the reactive power is zero. That means that the input power to the PMSM is only active power thus the *VA* rating of the VSI is minimized.

B. Constant stator flux strategy

This control property the magnitude of the stator -flux, limited thus limiting the stator voltage requirement. When the stator-flux is limited then also the torque producing capability PMSM is also, limited. Usually in the case of the stator flux control for PMSM the magnitude of the stator flux is maintained constant and equal to the permanent magnet flux linkage.

C. Constant torque angle strategy

In this control strategy the d- axis reference current is kept zero, while the vector current is align with the q –axis, in order to maintain the torque angle equal with 90 degree (between the current space vector and permanent magnet flux axis). Its simplicity, especially for SPMSM. In case of IPMSM, excitation flux is step-up by magnets; subsequently no magnetizing current is needed from the supply. As a result, the electromagnetic torque will be directly proportional to the q-axis component of the stator current vector.Hence better dynamic performance is obtained by controlling the electro-magnetic torque separately. Therefore, this torque can be written by

$$T_e = \frac{3}{2}\frac{p}{2}\lambda_f i_q \qquad (3.1)$$

D. Maximum torque per- ampere strategy

The main idea of this control strategy is develop the torque using minimum value of stator current amplitude. In interior PMSM, the q-axis inductance is most often greater than the d-axis inductance, therefore, the negative i_d will be needed to produce the positive torque. The positive i_d value is always avoided since it resists the field torque to be produced. In this case, the i_d components is not equal zero, and may cancel the reluctance torque produced by high saliency ratio. In the case of SPMSM, the same to as the constant torque angle property since there is almost no difference in the *d-q - axis* inductances thus the reluctance torque is almost in existent.

3.3. Space Vector Definition and Projection

The Field Orientated Control (FOC) consists of controlling the stator currents represented by a vector. This control is based on projections, which transform a three phase, time variant system and speed dependent system into a two co-ordinate (d and q coordinates) time invariant system. These projections lead to a structure similar to DC machine control.

The field oriented control of PMSM deals with three reference frames namely the three phase stator reference frame, two- phase stator reference frame and two- phase rotor reference frame as in figure 3.1. [28]

1. Stator reference frame (a-b-c) in which the (a-b-c) are co-planar, at 120 degrees apart.

2. An orthogonal reference frame (α, β) in the same plane as three phase stator reference frame (a-b-c) in which the angle between the two axes is 90 degrees instead of 120 degrees. The α -axis is aligned with β-axis in the second frame. Assuming that a- axis and α-axis are in the same direction and the two phase (α, β) currents are still depends on time and speed.

3. Rotor reference frame (d-q), in which the d axis is along the N and S poles (the rotor magnetic axis) or along the flux vector of the rotor and the quadrature- axis is at 90 degrees to the d -axis and rotates at the same speed as the rotor.

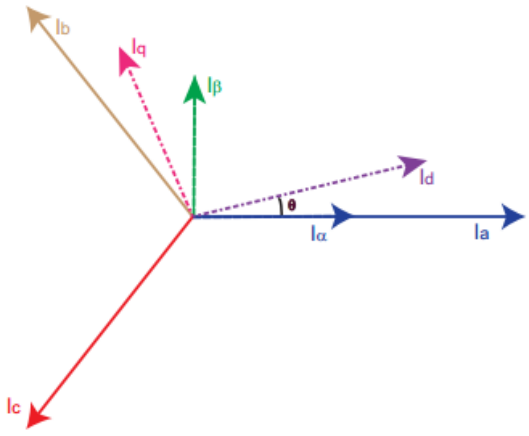


Figure 3.1. Representation of reference frames in FOC

The space vectors can be used electively to model IPMSM as these are easier for modeling and control purposes. The phase variables can also be easily transformed to the space vector models. In addition, the space vectors do not include the zero sequence components because the stator winding is delta connected or even if the winding is star connected, the neutral point is open. Figure 3.2 shows the transformations done for decoupling the stator currents into the torque (I_q) and flux (I_d) producing components are controlled with the PI controller, the controlled outputs which are the voltages, are then transformed back (inverse transformation) to the stator reference frame.

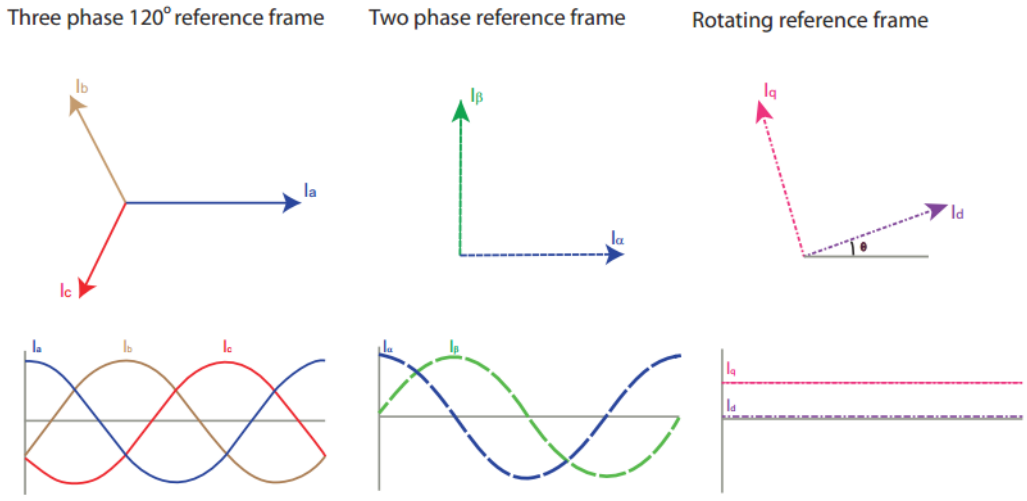


Figure 3.2.Three current reference frames and their wave forms [28]

❖ Transformations

Transforming from a stator reference frame of a PMSM into a rotary reference frame is used to get the time in varying values of the motor parameters. In FOC, the components i_q and id i_d are measured in the rotating reference frame. Hence the measured stator currents have to be transformed from the three phase time variant stator reference frame to the two axis rotating d-q rotor reference frame. This can be done in two steps as shown in Figure 3.3.

❖ a, b, c to α-β (Clarke transformation):

The measured motor currents are first translated from the 3-phase reference frame to the two axis orthogonal stationary reference frame. However, the quantities are still in a stationary reference frame while the rotor reference frame is constantly rotating. The transform is expressed by the following equations [29].

$$\begin{bmatrix} i_\alpha \\ i_\beta \end{bmatrix} = \frac{2}{3}\begin{bmatrix} 1 & \frac{-1}{2} & \frac{-1}{2} \\ 0 & \frac{\sqrt{3}}{2} & -\frac{\sqrt{3}}{2} \end{bmatrix}\begin{bmatrix} i_a \\ i_b \\ i_c \end{bmatrix} \tag{3.3}$$

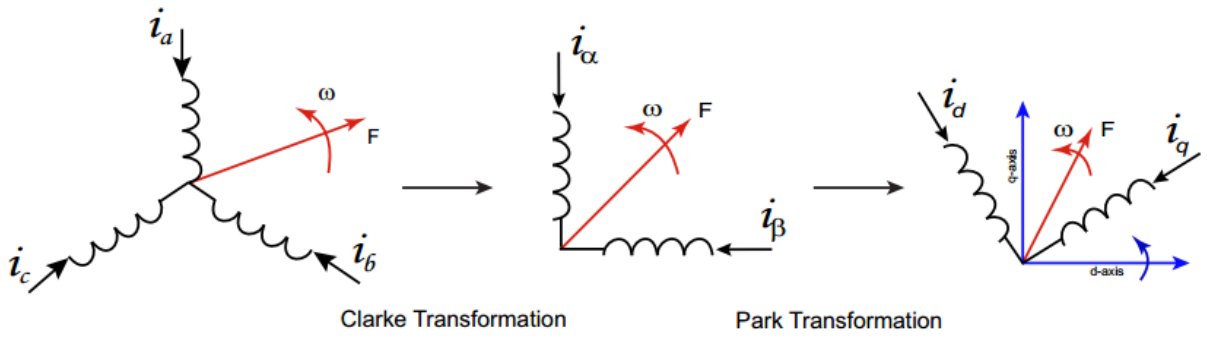


Figure 3.3: Forward Transformations [28]

❖ α-β to d-q (Park transformation):

Park transformation converts a two orthogonal stationary frame into a two orthogonal rotating frame (d-q), where (d-axis) is aligned with the magnetic -axis of the rotor which has a permanent magnet in PMSM. The flux and torque components of the current vector are determined by the following equations: [30]

$$\begin{cases} i_d = i_\alpha \cos\theta + i_\beta \sin\theta \\ i_q = -i_\alpha \sin\theta + i_\beta \cos\theta \end{cases} \quad (3.4)$$

❖ d-q to α-β (Inverse Park transformation):

The quantities in rotating reference frame are transformed to two axis orthogonal stationary reference frame using Inverse Park transformation. By applying Inverse Park's transformation, the stator voltage can be transformed to rotor reference frame is expressed by the following:

$$\begin{cases} v_\alpha = v_d \cos\theta - v_q \sin\theta \\ v_\beta = v_d \sin\theta + v_q \cos\theta \end{cases} \quad (3.5)$$

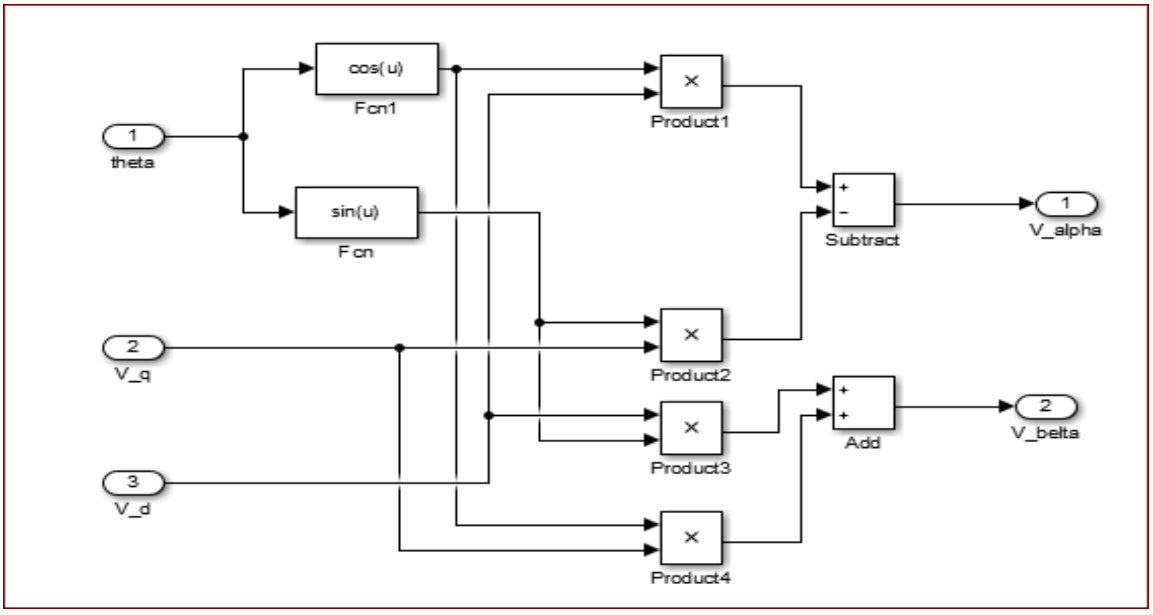


Figure 3.4. Simulation block diagram of Inverse Park transformation

3.4. Field Oriented Control principle for speed control of PMSM

In order to achieve better dynamic performance, an efficient control scheme needs to be applied to control the PM motor. With the mathematical processing, advanced control strategies can be implemented, which uses mathematical transformations in order to decouple the torque generation and the magnetization functions in the PM motors. These are called as field oriented control or vector control. FOC control is actually control of phase and amplitude for a motor stator voltage or current vector at the same time .To obtain the dq–axis current feedbacks in DC quantities, first a-b-c-axis AC currents are transformed into stationary values using Clarke transformation and then the stationary current values acquired from Clarke transformation are set up as inputs to the Park transformation along with the rotor position feedback signal to generate the equivalent DC feedback quantities in dq reference frame. In PMSM, the rotor windings are already along the dq–axis, only the stator windings quantities need transformation from three-phase quantities to the two-phase, d-q-axis rotor rotating reference frame quantities.

The Field Oriented Control technique, two proportion integral (PI) controllers are used to regulate the d-axis and q-axis currents, and a PI controller is used to regulate speed. The d-axis component of the current is made zero or other control techniques used in order to improve the performance and accuracy. This current is compared with motor d-axis current component and the error is fed to PI controller to find d-axis reference voltage component. The outputs of the current PI regulators generate stator dq–axes voltage references which are also in DC quantities (it is possible to use a different controller). Thus an inverse of the previous process has to be applied to get the reference voltage waveforms in stationary reference frame.

General, the motor current signals can be got by using current sensor and the actual current value of d-q axis can be calculated through the Clack transform and Park transform. The d-q axis actual currents are compared with the current reference signals. Then the voltage reference signals can be got through PI regulation and the space vector pulse-width modulation (SVPWM) technique generates PWM pulses to switch the voltage source inverter to produce sinusoidal output voltage of required frequency.

3.5 Design of PI Controllers Modeling

PI controllers are used widely for motion control systems. They consist of a proportional gain that produces an output proportional to the input error and an integration to make the steady state error zero for a step change in the input. To design PI-controllers, it's necessary to know the transfer function, which obtain from model of PMSM in d and q-axis. Model is divided into two orthogonal axis enabling independent control of flux and torque component.

Field Oriented Control scheme of a PMSM needs three PI Controller and two control loops. Two current controllers in rotating frame under current loops that control the torque and flux. The outer loop is the speed loop that controls the speed of the motor. In order optimum regulation and stability, internal loop has no influence on the speed loop; the bandwidth of the current loop should be at least 6-8 times higher than the band width of the speed loop.

Both of magnitude optimum and symmetric optimum are tuning methods based on finding a controller that gives the open and closed loop frequency response a desired shape. The first step in the application of these tuning methods for PI -controller is to determine appropriate transfer function which models the process. To describe appropriate model of IPMSM, d-q coordinate system fixed to machine rotor is used.

Once the transfer function is determined, the controller is able to shape the open- loop transfer function in a desired manner.

- The Magnitude Optimum (MO) method expects the system's transfer function in open loop of the following form:

$$G_{ol\_MO}(s) = \frac{1}{2T_{\mu}s(1 + T_{\mu}s)} \tag{3.6}$$

- The Symmetric Optimum (SO) method assumes the system's open loop transfer function as follows:

$$G_{ol\_SO}(s) = \frac{4T_{\mu}s + 1}{8T_{\mu}s(1 + T_{\mu}s)} \tag{3.7}$$

- The transfer functions system for the MO and SO in closed loop is present in evaluates eqn.3.8 and eqn.3.9 respectively:

$$G_{cl_{MO}}(s) = \frac{1}{2{T_{\mu}}^2 s^2 + 2T_{\mu}s + 1} \tag{3.8}$$

$$G_{cl_{SO}}(s) = \frac{4T_{\mu}s + 1}{8{T_{\mu}}^{3}s^{3} + 8{T_{\mu}}^{2}s^{2} + 4T_{\mu}s + 1} \quad (3.9)$$

Where, T_{μ} is sum of all small delays in loop and it's much smaller than constant time.

Table-1. Step responses for MO and SO methods [31]

Optimum parameters	MO	SO
Rise time	$4.7T_{\mu}$	$3.1T_{\mu}$
Setting time (2%)	$8.4T_{\mu}$	$16.5T_{\mu}$
Overshoot	4.3%	43.4%
Phase margin	65.5°	37°

3.5.1 Design PI Current of Controller

Current controller design is usually based on linear control system methods such as Bode diagram, root locus or by optimization functions. Current loop controller design is very important, for permanent magnet synchronous motor drives in high performance applications. Current controller, which is usually a PI controller, can be designed based on the self-inductance and stator resistance for each phase of the motor [32].The performance of IPMSM is highly dependent on the design of the speed and current controllers. The speed and current controllers are used in the outer and an inner control loop of FOC scheme.

The fast control loop executes two independent current control loops. They are the direct and quadrature axis current PI controllers. The current PI -controllers' outputs are summed with the corresponding d and q- axis components of the decoupling stator voltage. Thus, the desired space vector for the stator voltage is obtained and then applied to the motor. To achieve completely independent control of the direct-axis stator current i_d(field producing component) and the

quadrature-axis stator currenti_q (torque-producing component) it is necessary to cancel the effect of these coupling terms at the output of the current PI regulator [10].

The d-q-axis of stator currents can only be decoupled control if the stator voltage equations are decoupled, so these stator current components are indirectly controlled by controlling the terminal voltages of the synchronous motor. An exact decoupling action could be obtained by including inductance saturations. A decoupling and feed forward algorithm reduces the cross-coupling effects between the two axes and simplifies the PI regulators design as sketched in fig 3.5.

The d-q-axis stator voltage of eqn. (2.6) can be expressed by linear and decoupling terms as follows: Linear components V_d^{lin} and V_q^{lin} are set by the outputs of the current controllers.

$$\begin{cases} V_d = V_d^{lin} + V_d^{dec} = \left[R i_d + L_q \frac{d}{dt} i_d\right] - L_q \omega_r i_q \\ V_q = V_q^{lin} + V_q^{dec} = \left[R i_q + L_q \frac{d}{dt} i_q\right] + \omega_r \left(L_q i_d + \lambda_f\right) \end{cases} \quad (3.10)$$

Where

$$V_d^{lin} = \left(R + L_d \frac{d}{dt}\right) i_d, \quad V_q^{lin} = \left(R + L_q \frac{d}{dt}\right) i_q \quad (3.11)$$

And $$V_d^{dec} = -L_q \omega_r i_q , \qquad V_q^{dec} = \omega_r \left(L_q i_d + \lambda_{pm}\right) \quad (3.12)$$

The linear voltage components (V_d^{lin} *and* V_q^{lin})are the outputs of the current controllers which control the d-q-axis stator currents and added to the decoupling voltage components V_d^{dec} and V_q^{dec}. Finally, it can be seen that the decoupling technique transforms the nonlinear motor model to a linear one that can be controlled by the PI current loop controller.

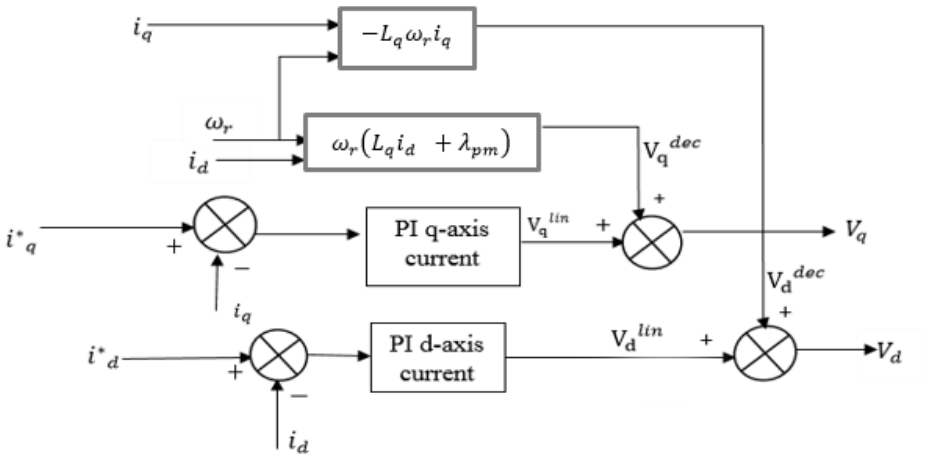


Figure 3.5: Block diagram of Voltage decoupling.

3.5.1.1Tuning the i_q PI controller

The diagram of the q-current controller is the one presented in Fig.3.6. The decoupling between the d and q-axis are performed. [33]

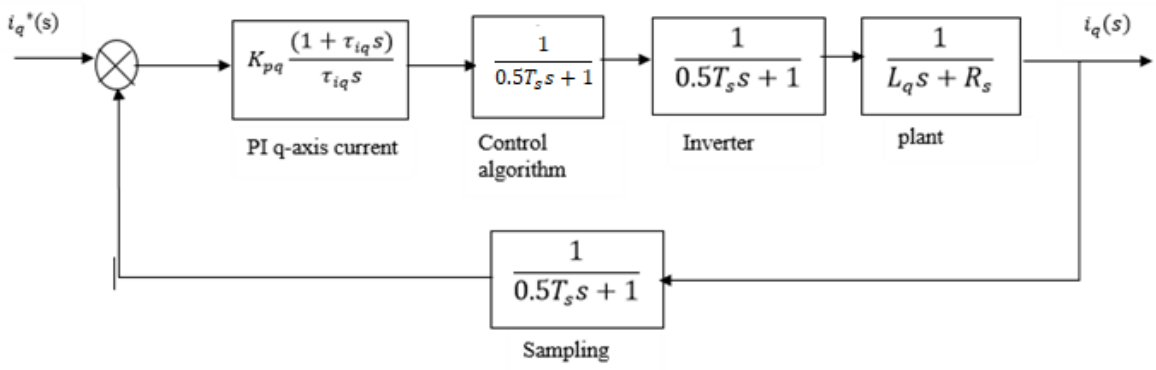


Figure 3.6: The structure of the q-axis current controller

The blocks from the figure above are explained in details in the following part:

- The transfer function of the PI controller block is given by. [34]

$$G_{PI\_iq}(s) = K_{pq}\frac{(1+\tau_{iq}s)}{\tau_{iq}s} \tag{3.13}$$

Where,K_{pq} is proportional gain and τ_{iq} is integral time constant.

- The plant transfer function block is determining from the q-axis voltage equations after the decoupling term has been removed and considering current as input and voltage as output.

$$G_{P\_iq}(s) = \frac{i_q(s)}{V_q(s)} = \frac{1}{L_q s + R_s} = \frac{\frac{1}{R_s}}{\frac{L_q}{R_s}s + 1} = \frac{\frac{1}{R_s}}{T_{sq}s + 1} \tag{3.14}$$

Where,$T_{sq} = \frac{L_q}{R_s}$ is q-axis motor time constant of the motor.

- ❖ The sampling block is the delay introduced by the current sensor. It is also a first order transfer function with the time constant equal to 0.5m [sec].Means, the sampling frequency choosing for my system is the values equal to 2 kHz.
- ❖ The control algorithm block represents the delay introduced by the digital calculations. It has the form of a first order system (time constant= $0.5T_s$).
- ❖ The inverter block is delay emulating presence of the inverter driven by pulse-width modulation (time constant= $0.5T_s$).

In order to have a simpler control loop the feedback path is moved to the forward path as shown in Fig.3.7. [35].

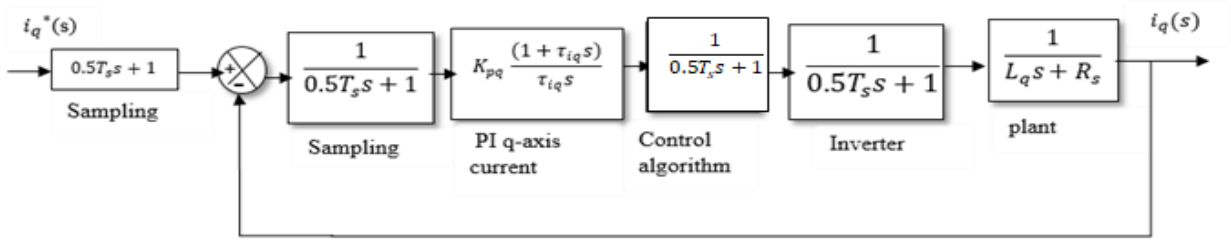


Figure 3.7:Design of the q-axis current loop with unity feedback

The equivalent open loop transfer function of the i_q loop has the following expression:

$$G_{ol\_iq}(s) = \frac{1}{0.5T_s s + 1} * K_{pq}\frac{(1+\tau_{iq}s)}{\tau_{iq}s} * \frac{1}{0.5T_s s + 1} * \frac{1}{0.5T_s s + 1} * \frac{\frac{1}{R_s}}{T_{sq}s + 1} \tag{3.15}$$

In controller the design, PI-controller gain is determined using by pole assignment controller design, the denominator transfer function (3.15) of corresponding order is compared with denominator of the close-loop transfer function. To improve the response characteristics of the

system *by ″*pole placement method*″*, use zero placement approach first by cancelling the zero of controller equation. This gives:

$$1 + \tau_{iq}s = 1 + T_{sq}s \Rightarrow \tau_{iq} = T_{sq} = \frac{L_q}{R_s} = \frac{0.1024}{5.8} = 0.017655[sec] \qquad (3.16)$$

In order to find out the gain of the PI controller, The sampling time of the simulation model was chosen to be T_s = 0.5 m(sec).All the first-order transfer functions that introduce delays are approximated by one transfer function that has the time constant equal to with the sum of all small time constants from the system.

$$T_{si} = 3 * 0.5T_s = 1.5T_s = 0.75m[sec] \qquad (3.17)$$

By doing this approximation, this open-loop transfer function rearrangement as the tuning types.

$$G_{ol\_iq} = K_{pq}\frac{(1 + \tau_{iq}s)}{\tau_{iq}s} * \frac{1}{(T_{si}s + 1)} * \frac{\frac{1}{R_s}}{(T_{sq}s + 1)} = \frac{1}{\frac{R_s\tau_{iq}}{K_{pq}}s(1 + T_{si}s)} \qquad (3.18)$$

The magnitude optimum criterion is used in design both current regulators, while the symmetric optimum applied for the speed regulator.

Comparing the obtained open-loop transfer function of the q-axis current from eqn 3.18 with the generic open loop transfer function for a second order system in Eq.3.6, where the damping ratio is set ζ =0.707 ,obtained the gain of the PI controller can be found [36].

Finally, parameters are tuned to the following values:

$$\begin{cases} T_{\mu} = T_{si} = 1.5T_s = 0.75m[sec] \\ 2T_{\mu} = \dfrac{R_s\tau_{iq}}{K_{pq}} \Rightarrow K_{pq} = \dfrac{R_s\tau_{iq}}{2T_{\mu}} = \dfrac{L_q}{3T_s} = 68.2667 \\ K_{iq} = \dfrac{R_s}{2 * T_{si}} = 3866.67 \end{cases} \qquad (3.19)$$

The q-axis current, PI transfer function in s domain has the fallowing form:

$$G_{PI\_iq}(s) = K_{pq}\frac{(1 + \tau_{iq}s)}{\tau_{iq}s} = K_{pq} + \frac{K_{iq}}{s} = 68.2667 + \frac{3866.67}{s} \qquad (3.20)$$

The Bode diagram of the open loop system is plotted in Fig.3.8

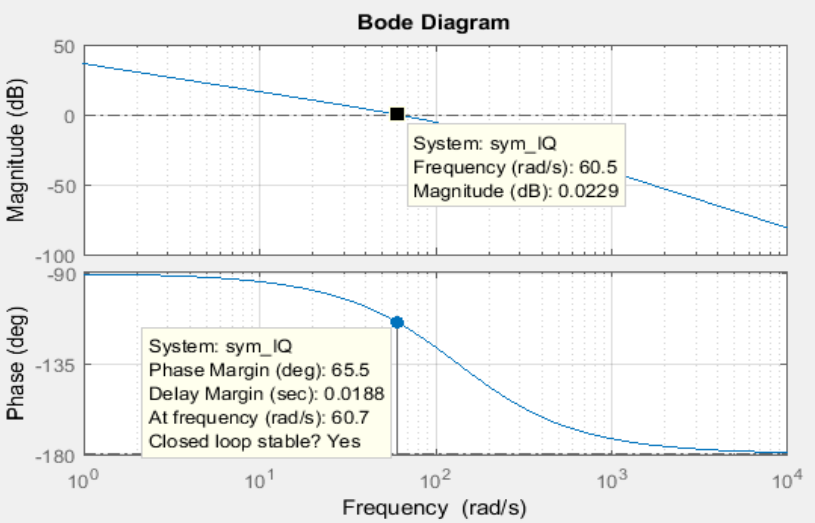


Figure 3.8: Bode plot for the i_q loop

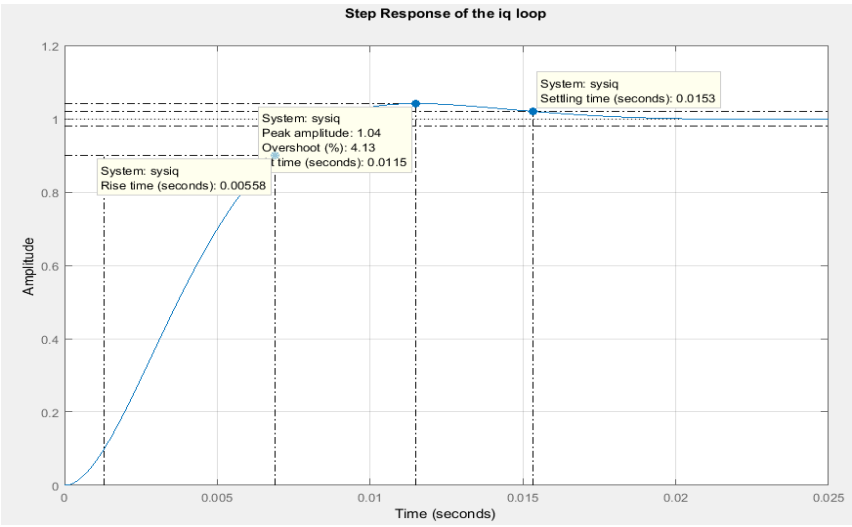


Figure 3.9: Step response for the i_q loop

As shown in Fig.3.8 the $i_q$closed loop system is stable. The gain margin value is GM=0.0229db and the phase margin values is PM=65.5deg.The step response of the closed loop system is presented in fig.3.9, with characterized parameters. In the following an equivalent time constant for i_q loop is derived. The reason for doing this time constant, viewed as a delay, used

fortuning the outer loop. [37].Taking account of the close loop transfer function has the following configuration.

$$G_{cl_{iq}}(s) = \frac{1}{2{T_{si}}^2 s^2 + 2T_{si}s + 1} \tag{3.21}$$

The q-current loop can be expressed now like in Fig.3.10

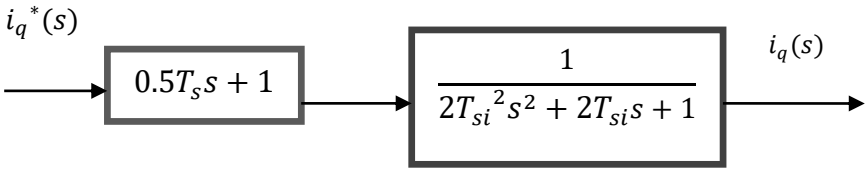


Figure 3.10: i_q current loop

- If the transfer function $0.5T_s s + 1$ is approximated like

$0.5T_s s + 1 \approx \frac{1}{(1-0.5T_s s)}$ [35]

- Then transfer function of the q-current closed loop becomes:

$$G_{cl\_iq}(s) = \frac{{i_q}^*(s)}{i_q(s)} = \frac{1}{(1 - 0.5T_s s)} * \frac{1}{2{T_{si}}^2 s^2 + 2T_{si}s + 1} \tag{3.22}$$

Second order component in (3.22) is much smaller than unity, and can be neglected. Thus, $G_{cl\_iq}(s)$ is reduced to first-order system. Time constant of entire q-current loop is computed.

$$T_{eq} = 2T_{si} - 0.5T_s = 2.5T_s = 1.25m[sec] \tag{3.23}$$

And resulting, approximated transfer function is:

$$G_{cl\_iq}(s) = \frac{1}{T_{eq}s + 1} \tag{3.24}$$

3.5.1.2 Tuning the i_d PI controller

The tuning of the d-axis current loop is designed the same to as the q-axis current loop, in's' domain. Therefore, delays have been introduced, due to the delays in a real-life system. These delays are the same as for the i_{sq}current loop. The only difference from the q current loop of the IPMSM, which is different due to the difference of L_d and L_q inductances.

After cancelling out the cross-copled terms, between the transfer function flux component, i_d and input voltage, $V_d$in *d*-axis stator current is given by equation (3.8).

$$\frac{i_d(s)}{V_d(s)} = \frac{1}{L_d s + R_s} = \frac{\frac{1}{R_s}}{\frac{L_d}{R_s}s + 1} = \frac{\frac{1}{R_s}}{\tau_{id}s + 1} \qquad (3.25)$$

Where, $\tau_{id} = \frac{L_d}{R_s}$ is *d*-axis motor constant times.

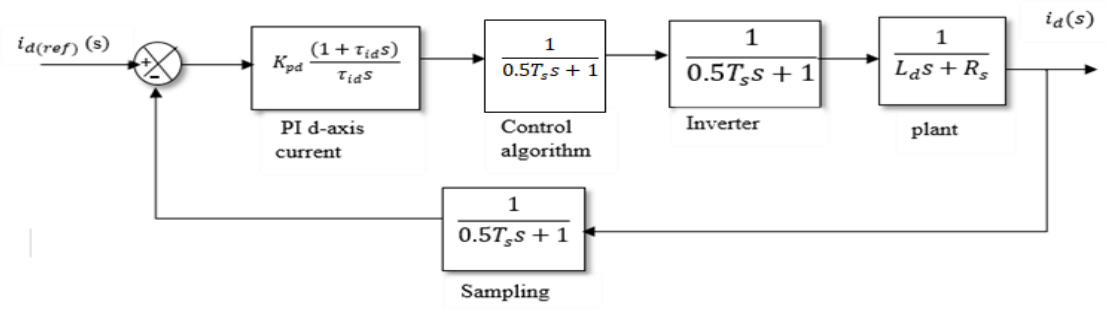


Figure 3.11: Design of the d-axis current loop

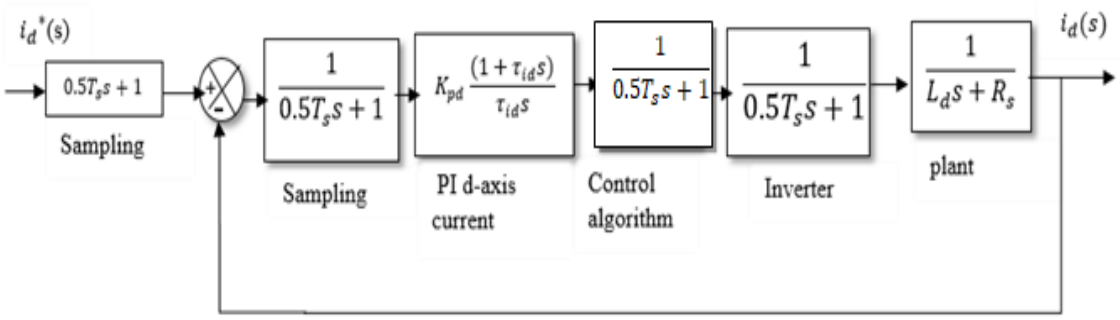


Figure 3.12: Design of the $i_d(s)$ current loop with unity feedback [37]

All delays introduced are the same as for the q-axis current-loop, so the equivalent time constant has the same value as for the q-axis current loop. In the tuning process of PI controllers, the nonlinearities are generally neglected, and tuning is done following the criteria adopted for electric drives [38].

The equivalent open-loop transfer function of the i_{sd} loop has the following expression:

$$G_{ol\_id}(s) = K_{pd}\frac{(1+\tau_{id}s)}{\tau_{id}s} * \frac{1}{0.5T_s s+1} * \frac{1}{L_d s+R_s} * \frac{1}{0.5T_s s+1} * \frac{1}{0.5T_s s+1}$$

$$G_{ol_{id}}(s) = K_{pd}\frac{(1+\tau_{id}s)}{\tau_{id}s} * \frac{1}{L_d s+R_s} * \frac{1}{T_{si}s+1} \qquad (3.26)$$

The equivalent open loop transfer function, taking account of the canceling-out of the pole of the IPMSM transfer function by zero of the PI transfer function, and also of the approximated time constant of the delays is as follows:

$$G_{ol_{id}}(s) = \frac{K_{pd}}{\tau_{id}s} * \frac{1}{R_s(T_{si}s+1)} = \frac{1}{\frac{R_s\tau_{id}}{K_{pd}}(1+T_{si}s)} \qquad (3.27)$$

The comparison of the ideal, first-order transfer function with denominator of the transfer function of current control loop results in values of proportional gain K_{pd} and time integral constant τ_{id} of PI-controller in d-axis while respecting prescribed settling time:

$$\begin{cases} T_{\mu} = T_{si} = 1.5T_s \\ 2T_{\mu} = \dfrac{R_s\tau_{id}}{K_{pd}} => K_{pd} = \dfrac{R_s\tau_{id}}{2T_{\mu}} = \dfrac{R_s\frac{L_d}{R_s}}{2*1.5T_s} = \dfrac{L_d}{3T_s} = 29.8667 \\ \dfrac{K_{pd}}{K_{id}} = \dfrac{L_d}{R_s} => K_{id} = \dfrac{R_s * K_{pd}}{L_d} = \dfrac{R_s}{3T_s} = 3866.67 \end{cases} \qquad (3.28)$$

The resulting, transfer function of the pi-current controller, for the d-axis current-loop has the following expression:

$$G_{PI\_id}(s) = K_{pd}\frac{(1+\tau_{id}s)}{\tau_{id}s} = K_{pd} + \frac{K_{id}}{s} = 29.866 + \frac{3866.67}{s} \qquad (3.29)$$

3.5.2 Speed Controller Design

Speed controller calculates the difference between the reference speed and the actual speed producing an error, which is fed to the PI controller. Speed control system can follow the input changes with a certain response speed and control precision. The main role of control system is to solve the problem of dynamic response speed. Speed loop can suppress load disturbance and speed fluctuation. It can track the given speed with small error. For the design of speed PI controller, the transfer function of the speed loop with all the delays considered and incorporating the inner q-axis current loop.

The blocks from the fig.3.13 are described in the following:

- The PI speed controller with the transfer function presented in equation (3.30)

$$G_{pI_\omega}(s) = K_{p\omega}\frac{(1+\tau_{i\omega}s)}{\tau_{i\omega}s} \tag{3.30}$$

Where,$K_{p\omega}$is the proportional gain of the speed controller and $\tau_{i\omega}$ is the time integral.

- The inner current-loop can be expressed like a first-order system with the time constant T_{eq}, the transfer function presented in equation (3.31)

$$G_{cl\_iq}(s) = \frac{1}{T_{eq}s+1} \tag{3.31}$$

- The plant block is calculated from the mechanical equation of the PMSM as shown in equation (3.32)

$$\frac{3}{2}n_{pp}\left(\left[L_d - L_q\right]i_d i_q + \lambda_{PM} i_q\right) = T_L + \mathrm{B}\omega_m + J\frac{d\omega_m}{dt} \tag{3.32}$$

Taking the Laplace transform, after rewriting the speed differential equation

$$\begin{cases} J\dfrac{d\omega_m}{dt} = \dfrac{3}{2}n_{pp}\left(\left[L_d - L_q\right]i_d i_q + \lambda_{PM} i_q\right) - T_{\boldsymbol{L}} + \mathrm{B}\omega_m \\ \omega_m(s) = \dfrac{1}{\mathrm{Js}+\mathrm{B}}\left(\dfrac{3}{2}n_{pp}\left(\left[L_d - L_q\right]i_d(s) i_q(s) + \lambda_{PM} i_q(s)\right) - \boldsymbol{T_L}(s)\right) \end{cases} \tag{3.33}$$

When a mechanical load is applied to an electric motor, control system treats it as disturbance. As a result, when the system reaches to a stable point, the load torque will be rejected.

To simplify the close loop, transfer function the disturbances are not taken into consideration and the feedback path is moved on the forward part as in the case of q-current controller.

$$\omega_r(s) = \frac{n_{pp}k_t}{\mathrm{Js + B}} i_q(s) \tag{3.34}$$

Where, $k_t = \frac{3}{2} npp\lambda_{PM}$ and $\omega_r = npp\omega_m$

The close-loop system speed with unity feedback in presented shown in Figure 3.13. [39]

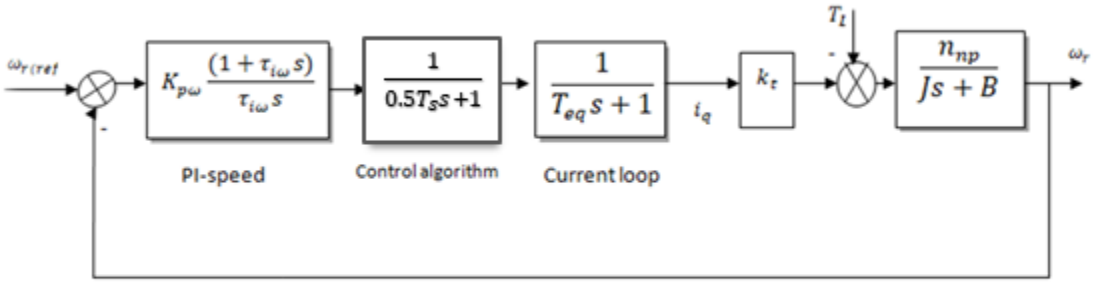


Figure 3.13: Block diagram of the design the speed loop with torque load

The open-loop transfer function is the one presented in equation (3.35)

$$G_{ol\_\omega}(s) = K_{p\omega}\frac{(1+\tau_{i\omega}s)}{\tau_{i\omega}s} * \frac{1}{T_{eq}s+1} * \frac{1}{0.5T_s s+1} * \frac{n_{pp}K_t}{Js+B} \tag{3.35}$$

In order to find the gain of the PI speed controller, as in the current loop, the speed loop contains some delays that need to be defined approximated into one time constant as presented in equation (3.36) .For this loop, it is common to use a sampling time ten times higher than the sampling time of the q-current loop.

$$T_{s\omega} = T_{eq} + 0.5T_s = 2.5T_s + 0.5T_s = 1.5m[sec] \tag{3.36}$$

The open loop transfer function becomes:

$$G_{ol\_\omega}(s) = K_{p\omega}\frac{(1+\tau_{\omega}s)}{\tau_{\omega}s} * \frac{1}{T_{s\omega}s+1} * \frac{n_{pp}K_t}{Js+B} = \frac{n_{pp}K_tK_{p\omega}(1+\tau_{i\omega}s)}{\tau_{i\omega}s(1+T_{s\omega}s)Js+B} \tag{3.37}$$

The speed close-loop transfer function of with PI-controller is as follows:

$$G_{cl\_\omega}(s) = \frac{n_{pp}K_tK_{p\omega}(1+\tau_{i\omega}s)}{\tau_{i\omega}T_{s\omega}Js^3 + \tau_{i\omega}(J+T_{s\omega}B)s^2 + \tau_{i\omega}(B+K_{p\omega}n_{pp}K_t)s + K_{p\omega}n_{pp}K_t} \quad (3.38)$$

Then simplifying the expression as method of optimum the following equations obtained.

$$G_{cl\_\omega}(s) = \frac{(1+\tau_{i\omega}s)}{\frac{\tau_{i\omega}T_{s\omega}J}{n_{pp}K_tK_{p\omega}}s^3 + \frac{\tau_{i\omega}(J+T_{s\omega}B)}{n_{pp}K_tK_{p\omega}}s^2 + \frac{\tau_{i\omega}(B+K_{p\omega}n_{pp}K_t)}{n_{pp}K_tK_{p\omega}}s + 1} \quad (3.39)$$

In order to obtain an optimal response is necessary to tune the regulator according to the optimum symmetric method. The same ways designing process as in the MO method is followed to adapt the transfer function current to used Symmetric criterion in the speed control loop. With the help of equation close-loop transfer function of OS method and (3.9) PI controller gains, find out as presented below.

$$\begin{cases} 4T_\mu = \tau_{i\omega} \\ 8{T_\mu}^3 = \dfrac{\tau_{i\omega}T_{s\omega}J}{n_{pp}K_tK_{p\omega}} \\ 8{T_\mu}^2 = \dfrac{\tau_{i\omega}(J+T_{s\omega}B)}{n_{pp}K_tK_{p\omega}} \end{cases} \quad (3.40)$$

$$\begin{cases} \tau_{i\omega} = 4T_{s\omega} = 4(1.5m)[sec] = 6m[sec] \\ K_{p\omega} = \dfrac{J}{2n_{pp}T_\mu K_t} = \dfrac{0.0087}{2(1.599)*1.5*10^{-3}*2} = 0.90682 \\ K_{i\omega} = \dfrac{K_{p\omega}}{4T_{s\omega}} = 151.136 \end{cases} \quad (3.41)$$

The transfer function of the PI speed controller is found:

$$PI_\omega = K_{p\omega}\frac{(1+\tau_{i\omega}s)}{\tau_{i\omega}s} = 0.90682\frac{(1+6s)}{6s} = 0.90682 + \frac{151.136}{s} \quad (3.42)$$

3.6.Space Vector Pulse-Width Modulation Techniques

Space vector PWM refers to a special switching scheme of the six power semiconductor switches of a three phase power converter. Space vector PWM (SVPWM) has become a popular PWM technique for three-phase voltage-source inverters in applications such as control of induction and permanent magnet synchronous motors. The main advantage of the SVPWM technique is that the switching losses are low, low output harmonic distortions and offer 15% increase in the DC-link voltage compared with the conventional sinusoidal pulse-width modulation [40]- [41].

In SPWM, modulating sine waves is compared with a carrier triangular waveform to obtain PWM pulses for the three phases and in the space-vector modulation, the required pulses can be generated by comparing the modulating functions with the triangular waveform [42]. These PWM pulses are given to the gate of the inverter switches to get a controlled three phase output voltage which can be given to the motor input.

In the space-vector modulation, a three-phase two-level inverter can be driven to eight switch combination where the inverter has six active states (1-6) and two zero states (0 and 7). The switch combinations can be represented as binary codes that correspond to the top switches $S_1, S_3 and\ S_5$ of the inverter that generate three-phase voltage outputs. When an upper transistor is switched on, the corresponding lower transistor is switched off [43].

The circuit model of atypical three-phase voltage source inverter as shown in Fig.3.14.The circuit has a full-bridge topology with three inverter legs, each consisting of two power switches. The circuit allows only positive power flow from the supply system to the load via a full-bridge diode rectifier. Negative power flow is not possible through the rectifier diode bridge. The six switching power devices can be constructed using power BJTs, GTOs, IGBTs, etc. The choice of switching devices consider the desired operating power level, required switching frequency, and acceptable inverter power losses.

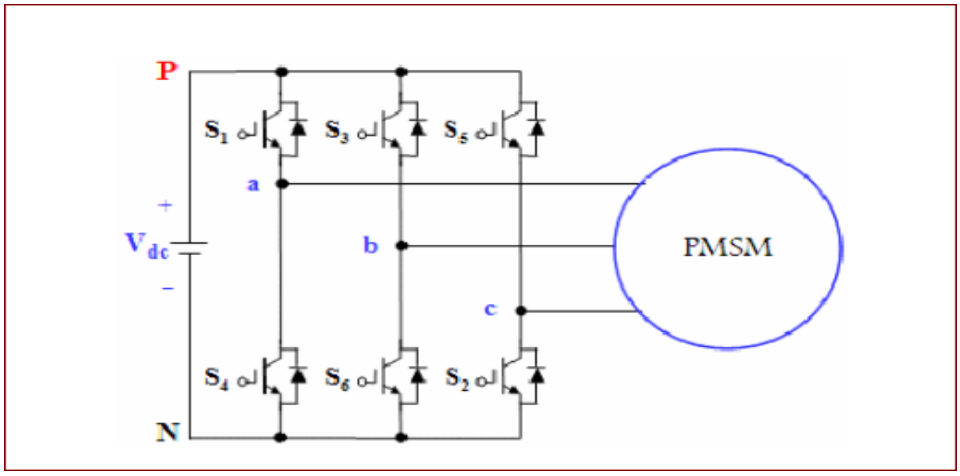


Figure. 3.14 Three-phase voltage source PWM Inverter. [44]

When the reference voltage vector passes through each sector, different sets of switches in Table 2 will be turned on or off.As a result, when V_{ref} rotates through one revolution in space, the inverter output varies one electrical cycle over time.The inverter output frequency coincides with the rotating speed of the reference voltage vector. The zero vectors (V0 and V7) and active vectors (V1 to V6) do not move in space [45].They are referred to as stationary vectors. Fig.3.15 shows the reference vector Vref in the first sector [43].The six active voltage space vectors are shown on the same graph with an equal magnitude of 2Vdc/3 and a phase displacement of 60 degrees. The inverter cannot produce a desired reference voltage vector directly. It is possible to decompose eight the reference vector into vectors that lie on two zero and active vectors, which are located at the center of hexagon. [46]

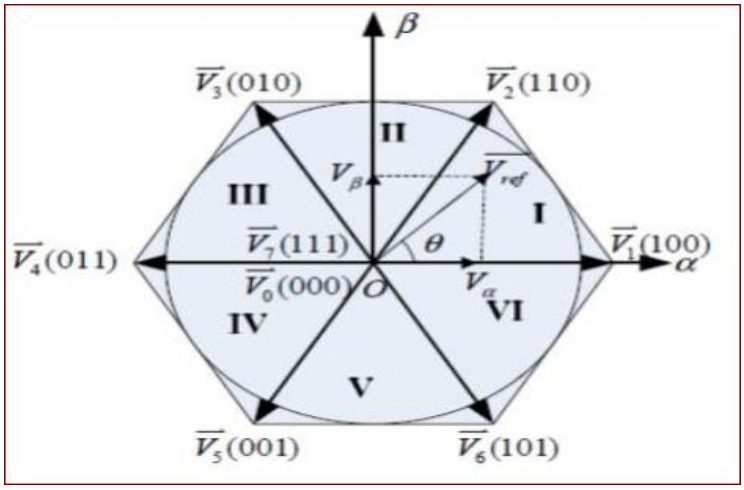


Figure. 3.15 Space vector hexagon for the two-level VSI

Since, all three Line-to-Neutral voltages must be sum to zero; a space vector V_s is easily calculated by the formula.

$$\begin{cases} V_a = V\cos(\omega t) \\ V_b = V\cos\left(\omega t + \dfrac{2\pi}{3}\right) \\ V_c = V\cos\left(\omega t + \dfrac{4\pi}{3}\right) \end{cases} \tag{3.43}$$

$$V_s = V_{an}e^{j0} + V_{bn}e^{j\frac{2\pi}{3}} + V_{cn}e^{j\frac{4\pi}{3}} \tag{3.44}$$

Where,$V_{an}, V_{bn}$and $V_{cn}$are the phase voltages of the inverter output.

$$\begin{cases} V_{an} = \left(\dfrac{1}{3}\right)(2 * V_{a0} - V_{b0} - V_{c0}) \\ V_{bn} = \left(\dfrac{1}{3}\right)(2 * V_{b0} - V_{a0} - V_{c0}) \\ V_{cn} = \left(\dfrac{1}{3}\right)(2 * V_{c0} - V_{a0} - V_{b0}) \end{cases} \tag{3.45}$$

The relationship between the switching variable vector $[a\text{-}b\text{-}c]^t$and the line to line voltage vector $[V_{ab}V_{bc}V_{ca}]^t$of the three phase inverter.

$$\begin{bmatrix} V_{ab} \\ V_{bc} \\ V_{ca} \end{bmatrix} = V_{dc}\begin{bmatrix} 1 & -1 & 0 \\ 0 & 1 & -1 \\ -1 & 0 & 1 \end{bmatrix}\begin{bmatrix} a \\ b \\ c \end{bmatrix} \tag{3.46}$$

Where, [a-b-c] is a vector representing the upper switches of the inverter.

The potential of point a, point b & point c with respect to the center point of the dc-link is known if the conducting states of the switches are known. When an upper switch is ″ON″, the potential of a, b & c is V_{dc} and when the lower switch is "ON", the potential of a, b & c is -V_{dc}.

Also, the relationship between the switching variable vector $[a\text{-}b\text{-}c]^t$ and the phase voltage vector $[V_aV_bV_c]^t$can be express:

$$\begin{bmatrix} V_{an} \\ V_{bn} \\ V_{cn} \end{bmatrix} = \frac{V_{dc}}{3}\begin{bmatrix} 2 & -1 & -1 \\ -1 & 2 & -1 \\ -1 & -1 & 2 \end{bmatrix}\begin{bmatrix} a \\ b \\ c \end{bmatrix} \tag{3.47}$$

The expression 3-phase voltage in the (α, β) frame by the general Clarke transform equation:

$$\begin{bmatrix} V_\alpha \\ V_\beta \end{bmatrix} = \frac{2}{3}\begin{bmatrix} 1 & -\frac{1}{2} & -\frac{1}{2} \\ 0 & \frac{\sqrt{3}}{2} & -\frac{\sqrt{3}}{2} \end{bmatrix}\begin{bmatrix} V_{an} \\ V_{bn} \\ V_{cn} \end{bmatrix} \quad (3.48)$$

The eight switching combinations of the derived output and three-phase voltages in terms of V_{dc} in the (α, β) frame, are shown in Table-2.

Table-2. Switching patterns and output voltages of a 3-phase power inverter

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_{ab}	V_{bc}	V_{ca}		
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

(Note that the respective voltage should be multiplied by V_{dc})

In general, the SVPWM implementation involves the following steps:

Step -1.Determine V_α, Vβ, V_{ref}, and angle (α)

Step- 2.Sector identification

Step- 3. Determine time duration T_1,T_2 andT_0

Step- 4. Determine the switching time of each transistor (S1 to S6)

Step- 5.The determination of vector switching points

Step- 1: Determine V_α, $V\beta$, V_{ref}, and angle (α)

Using the co-ordinate transformation to 2-Φ stationary reference frame in Fig-3.16, theV_α, V_β, V_{ref} and angle (α) can be determined as following.

$$V_\alpha = V_{an} - V_{bn}\cos(60) - V_{cn}\cos(60) = V_{an} - \frac{1}{2}V_{bn} - \frac{1}{2}V_{cn} \quad (3.49)$$

$$V_\beta = 0 - V_{bn}\cos(30) - V_{cn}\cos(30) = \frac{\sqrt{3}}{2}V_{bn} - \frac{\sqrt{3}}{2}V_{cn} \quad (3.50)$$

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

$$\begin{bmatrix} V_\alpha \\ V_\beta \end{bmatrix} = \begin{bmatrix} 1 & -\frac{1}{2} & -\frac{1}{2} \\ 0 & \frac{\sqrt{3}}{2} & -\frac{\sqrt{3}}{2} \end{bmatrix} \begin{bmatrix} V_{an} \\ V_{bn} \\ V_{cn} \end{bmatrix} \quad (3.51)$$

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

$$|V_{ref}| = \sqrt{{V_\alpha}^2 + {V_\beta}^2} \quad , \quad \alpha = \tan^{-1}\frac{V_\beta}{V_\alpha} = \omega t = 2\pi f t \quad (3.52)$$

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

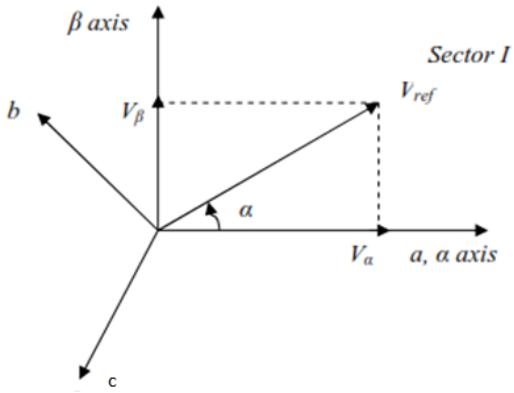


Figure 3.16: Voltage space vector and its components in (a-b-c -axis) [47]

Step- 2: Sector identification

The sector identification can be done by coordinate transformation [48]- [49]or by repeated comparison of the 3-phase reference voltages [50]- [51] , which the instantaneous reference space vector lays. In this thesis to identify the sector where there reference voltage space vector is present. It is necessary to know in which sector the reference output lies in order to determine the switching time and sequence. The phase voltage corresponding to eight switching states: six non-zero vectors and two zero vectors at the origin. Depending on the reference voltages, the angle of the reference vector can be determined the sector as per the table-3 [52].

Table-3 Sector Definition

Sector	Degrees
1	$0< \alpha \leq 60^o$
2	$60^o < \alpha \leq 120^o$
3	$120^o < \alpha \leq 180^o$
4	$180^o <\alpha \leq 240^o$
5	$240^o < \alpha \leq 300^o$
6	$300^o< \alpha \leq 360^o$

Step -3. Determine time duration T_1,T_2 and T_0

The reference space vector rotates and moves through different sectors of the complex plane as time increases. In each PWM cycle, $V_{ref}$is sampled at a fixed input sampling frequency. During this time, the sector is determined and the modulation vector V_{ref} is mapped onto two adjacent vectors.

Eight space vectors can be represented as a complex–vector expression:

$$V_k = \begin{cases} \frac{2}{3} V_{dc} expj(k-1)\frac{\pi}{3}, k = 1,2,\dots,6 \\ 0 \qquad\qquad\qquad\quad k = 0,7 \end{cases} \tag{3.53}$$

After calculating the value of V_{ref}, switching time duration $T_1, T_2\ and\ T_0$has to be calculated for sector 1.

The time duration of V_{ref} is calculated:

$$\int_0^{T_s} \vec{V}_{ref}\, dt = \int_0^{T_1} \vec{V}_1\, dt + \int_{T_1}^{T_1+T_2} \vec{V}_2\, dt + \int_{T_1+T_2}^{T_s} \vec{V}_0\, dt \qquad (3.54)$$

$$T_s = T_1 + T_2 + T_0 \qquad (3.55)$$

Where ,$T_1, T_2\ and\ T_0$, are the switching time durations, V_1 ,V_2 and zero vector respectively.T_s,is the switching time calculated by $T_s =$ sample time$= \frac{1}{f_s}$ and $f_s$is the switching frequency.

- **Switching time duration at sector 1**

Figure 3.17 shows the corresponding space vectors and time durations ($T_1, T_2\ and\ T_0$)in sector 1. In the first sector reference voltage is in between $V_1$and V_2.By switching the vectors V_1 and V_2 for $T_1$and T_2time seconds, the space phasor voltage is obtained. Switching time in Sector1 for voltage vector V_1 to be applied i.e. T_1sec duration and for vector V_2 i.e. T_1 sec duration can be found out by applying volt-second balance for reference voltage and applied voltage as stated below: [53].The vectors $V_0$and $V_7$are switched for T_0/2seconds.

For each switching period T_s the geometric summation can be expressed mathematically as [54]

$$T_s.\vec{V}_{ref} = \vec{V}_1.T_1 + T_2.\vec{V}_2 \qquad (3.56)$$

The average voltage for the sector 1 which is made by vectors V0, V1, V2and V7 is given by equation(3.42). (Where $0 \le \alpha \le 60^0$)

$$T_s \begin{bmatrix} V_\alpha \\ V_\beta \end{bmatrix} = T_s.|V_{ref}| \begin{bmatrix} \cos(\alpha) \\ \sin(\alpha) \end{bmatrix} = T_1 \frac{2}{3} V_{dc} \begin{bmatrix} 1 \\ 0 \end{bmatrix} + T_2 \frac{2}{3} V_{dc} \begin{bmatrix} \cos(\frac{\pi}{3}) \\ \sin(\frac{\pi}{3}) \end{bmatrix} \qquad (3.57)$$

Dividing these in real and imaginary parts simplifies the calculation for each duration time:

Real part: $T_s V_{ref} \cos(\alpha) = T_1 \frac{2}{3} V_{dc} + T_2 \frac{1}{3} V_{dc}$

Maginary Part : $T_s V_{ref} \sin(\alpha) = T_2 \frac{1}{\sqrt{3}} V_{dc}$

Therefore, switching time duration for sector 1 are given as:

$$\begin{cases} T_1 = T_s \frac{3}{2} \frac{|V_{ref}|}{V_{dc}} \frac{\sin\left(\frac{\pi}{3} - \alpha\right)}{\sin\left(\frac{\pi}{3}\right)} = T_s M_i \sin\left(\frac{\pi}{3} - \alpha\right) \\ T_2 = T_s \frac{3}{2} \frac{|V_{ref}|}{V_{dc}} \frac{\sin(\alpha)}{\sin\left(\frac{\pi}{3}\right)} = T_s M_i \sin(\alpha) \\ T_0 = T_s - T_1 - T_2 \end{cases} \quad (3.58)$$

The ratio of the peak value of the carrier and the modulating wave is defined as the frequency modulation ratio and it's denoted asM_i. [55].

$$M_i = \frac{|V_{ref}|}{\frac{2}{3}V_{dc}} \frac{\sqrt{3}}{2} = \frac{\sqrt{3}|V_{ref}|}{V_{dc}} \quad (3.59)$$

The generalized equations for calculating time duration of the respective space vectors in any of the sectors is given by equations (3.60).

$$\begin{cases} T_1 = T_s\sqrt{3}\frac{|V_{ref}|}{V_{dc}} \sin\left(\frac{n}{3}\pi - \alpha\right) \\ T_2 = T_s\sqrt{3}\frac{|V_{ref}|}{V_{dc}} \sin\left(\alpha - \frac{n-1}{3}\pi\right) \\ T_0 = T_s - T_1 - T_2 \end{cases} \quad (3.60)$$

Where , n=1 through 6(that is selector 1 to 6).

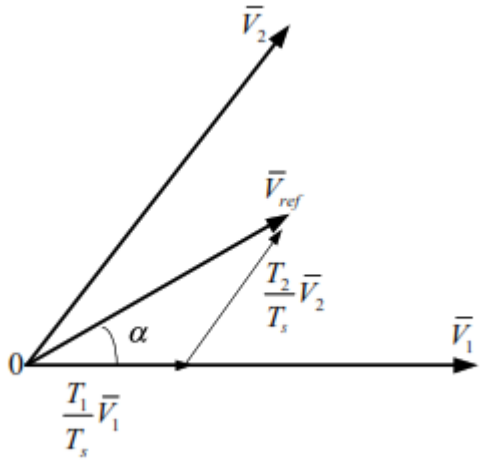


Figure 3.17: Reference voltage as a combination of adjacent vectors in sector I [56].

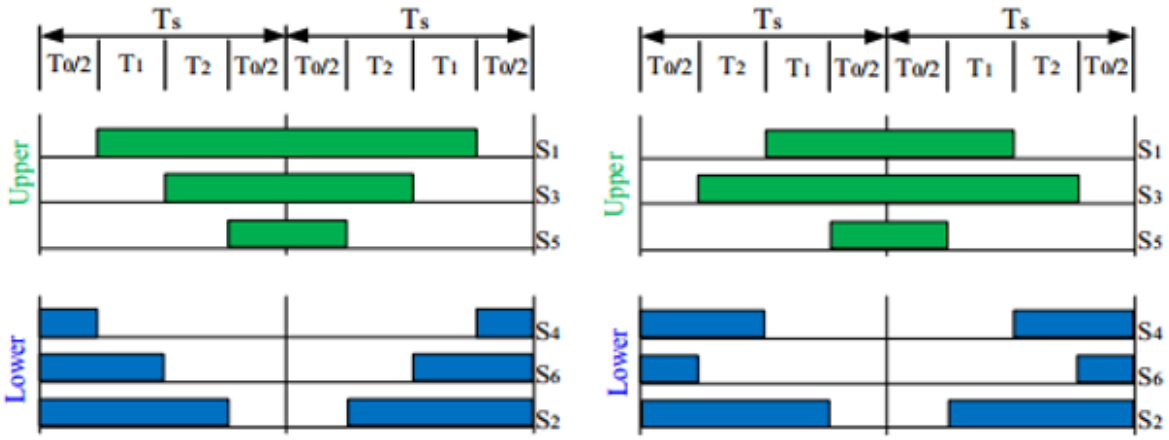
Step- 4. Determine the switching time of each transistor (S_1 *to* S_6)

In the SVPWM strategy, four types of switching patterns are as follows:- symmetric sequence, right aligned sequence, alternating zero vector sequence and highest current not switched sequence.

Select from this switch to arrange the switching sequence design for a given $\vec{V}_{ref}$ is not unique. It should assure two way for the minimization of the device switching frequency, when only two adjacent active and zero vectors are used in a sector.

a) The transition from one switching state to the next, one being switched ON and other being switched OFF in the same leg.

b) The movement of $\vec{V}_{ref}$ moving from one sector to the next requires no or minimum number of switches. Thus, space vector PWM switching uses patterns of 0127-7210 in first sectors, 0327-7230 in second sector and so on. Fig.3.18 shows space vector PWM switching patterns at sector 1 and 2. [57]



(a) Sector 1. (b) Sector 2.

Figure. 3.18 Space Vector PWM switching patterns for sector 1 and 2

The pattern includes the adjacent active vectors and zero vectors symmetrically distributed into respective switching periods in each sector. It is noted that the switching status of lower switches S4, S6 and S2 is complement to that of upper switches. The on time calculations of upper switches in each sector are as on Table-4 [58].

Table-4. Switching Time Calculation at Each Sector

Sector	Upper Switches(S_1,S_3,S_5)	Lower Switches(S_4,S_6,S_2)
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	$S_1= T_1+T_2+ T_0/2$ $S_3=T_0/2$ $S_5=T_1+T_0/2$	$S_4= T_0/2$ $S_6=T_1+T_2+T_0/2$ $S_2=T_2+T_0/2$

Step- 5.The determination of vector switching points

Symmetrical PWM sequence is used for switching the voltage vectors. The vector switching points can be calculated as below .The relation between sector number and vector switching points (T_a, T_b and T_c) is shown in Table-5.

$$\begin{cases} t_{aon} = 0.25(T - T_1 - T_2) \\ t_{bon} = t_{aon} + 0.5\,T_1 \\ t_{con} = t_{bon} + 0.5\,T_2 \end{cases} \qquad (3.46)$$

Table -5: The relation between N and vector switching points

	Sector number					
Vector Switching points	1	2	3	4	5	6
T_a	t_{bon}	t_{aon}	t_{aon}	t_{con}	t_{con}	t_{bon}
T_b	t_{aon}	t_{con}	t_{bon}	t_{bon}	t_{aon}	t_{con}
T_c	t_{con}	t_{bon}	t_{con}	t_{aon}	t_{bon}	t_{aon}

CHAPTER FOUR

4. Introduction

4.1. Principle of the MRAS control scheme

Model Reference Adaptive System (MRAS) is one of the most popular adaptive control method used in motor control applications for tracking and observing system parameters. The basics of MRAS are to use two independent models. It may be parallel model, series model, direct model and indirect model. In a MRAS, as shown in fig.4.1**,** adjustable model and reference model are connected in parallel. Similar to other estimation algorithm, it's also provides accurate estimation results.

A condition to form MRAS is that the reference model, which is independent of the variable to be estimated (the rotor speed), where provides reference the state values from the terminal voltage and current. Then adjustable model (or adaptive model), which is dependent on the variable to be estimated, estimates the state variable. The two models work at the same time, and compare both outputs**,** to variety a suitable adaptive rate, and adjust the parameter estimator to achieve the purpose of controlling the object output tracking reference model.

The adaptive mechanism should be designed, to regulate the current error convergence to zero and as the actual speed to estimate values of given parameter, which is used for the tuning in adjustable model and also for feedback. The variable in current model , modified by adaptation mechanism, in order to make its state variable $\hat{x}$ draw near x which also means the difference value approaches zero [59].The estimated value in this way is driven to its true value [60].The adaptation mechanism will be the commonly used PI-controller with fixed gains to guarantee the convergence of the system**.** The difference between real and estimated value can be expressed with the dynamic error equation [61].According to, model the state variables, speed estimation methods using MRAS can be classified into several types :- Rotor flux ,reactive power ,back –emf and stator current based on MRAS .

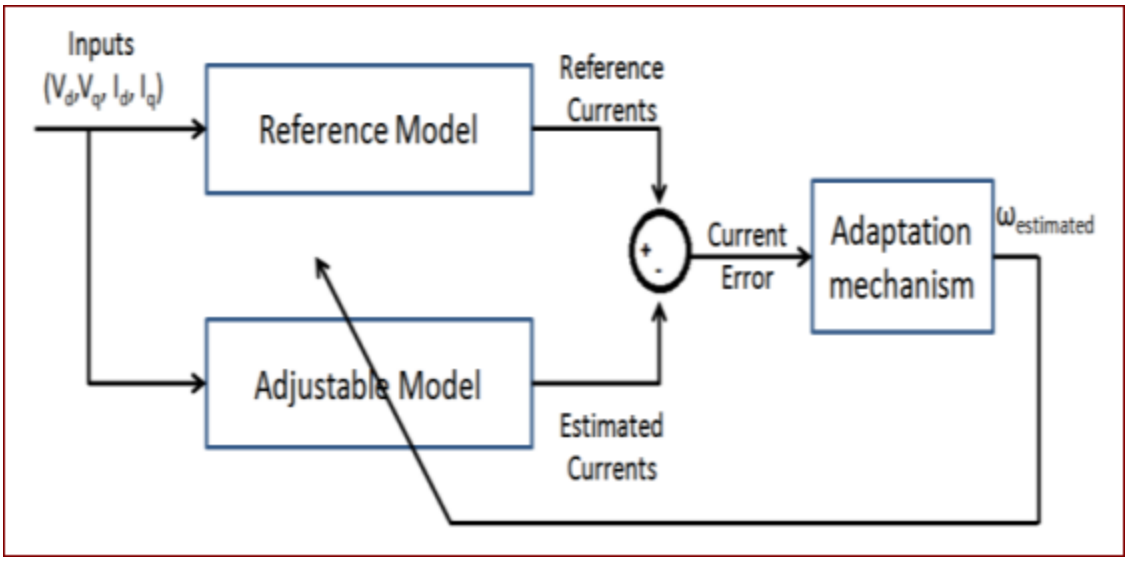


Figure. 4.1 Illustration of the Model Reference Adaptive System

4.2. Stator current based on M RAS speed estimator

In the presented MRAS estimator, this state variable is calculated on the basis of the speed-dependent current model .Depending on the quantity (i.e. the functional candidate) used to formulate the error signal; various kinds of MRAS are possible**.** In this thesis, based on the information taken from measurement of stator currents and it's developed with d- and q-components of stator current is called as the stator current based on MRAS speed estimator. This model is consists a reference model, an adaptive model and an adaptation mechanism to estimate the rotor speed. The block "Adjustable Model", which has the same structure as the reference one (the motor itself), but with adaptive or estimated parameters, instead of the unknown ones.

The adaptive model is also known as the current model and derived from the rotor voltage equation of the machine, and it contains the estimated rotor speed and stator current components. An error vector ε is derived using the difference between the outputs of the two dynamic models and is driven close to zero through an adaptive mechanism [62]- [63].As result, the estimated value of the rotor speed, also it's obtained through a certain adaptive mechanism. Then, the position estimation by integrating the rotor speed as shown in below [64].

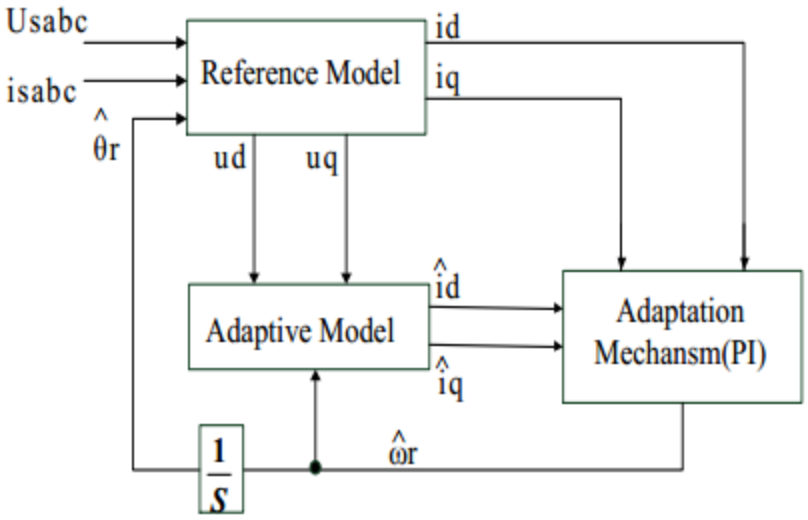


Figure 4.2 Block diagram of the estimation technique with MRAS [59]

The d-q axis stator currentsi_d and i_q are the state variables of the current model of the IPMSM, which is described by (4.1)

$$\begin{cases} L_d \dfrac{di_d}{dt} = -R_s i_d + \omega_r L_q i_q + u_d \\ L_q \dfrac{di_q}{dt} = -R_s i_q - \omega_r L_d i_d - \omega_r \lambda_{pm} + u_q \end{cases} \tag{4.1}$$

Where,$u_q, u_d, L_d, L_q, i_q and\ \ i_d$, are the stator voltage, the stator inductances and current of the motor in the d-q axis respectively, R_s are the stator resistance,$\omega_r$and λ_{pm} are rotor electric angular speed and flux respectively.

By rewriting equations (4.1) into a matrix form as below:

$$\frac{d}{dt}\begin{bmatrix} i_d \\ i_q \end{bmatrix} = \begin{bmatrix} \dfrac{-R_s}{L_d} & \omega_r \dfrac{L_q}{L_d} \\ -\omega_r \dfrac{L_d}{L_q} & \dfrac{-R_s}{L_q} \end{bmatrix} \begin{bmatrix} i_d \\ i_q \end{bmatrix} + \begin{bmatrix} \dfrac{1}{L_d} & 0 \\ 0 & \dfrac{1}{L_q} \end{bmatrix} \begin{bmatrix} u_d \\ u_q - \lambda_{pm}\omega_r \end{bmatrix} \tag{4.2}$$

Then, the Reference model can be rewritten as:

$$\frac{d}{dx}x = Ax + Bu \tag{4.3}$$

Where, $X = \begin{bmatrix} x_2 \\ x_2 \end{bmatrix} = \begin{bmatrix} i_d \\ i_q \end{bmatrix}, A = \begin{bmatrix} \frac{-R_s}{L_d} & \omega_r \frac{L_q}{L_d} \\ -\omega_r \frac{L_d}{L_q} & \frac{-R_s}{L_q} \end{bmatrix}$, B $= \begin{bmatrix} \frac{1}{L_d} & 0 \\ 0 & \frac{1}{L_q} \end{bmatrix}, u = \begin{bmatrix} v_d \\ v_q - \lambda_{pm}\omega_r \end{bmatrix}$

Considering speed estimation, with fixed values of the resistance and inductance, to constructed adjustable model from the reference model Equation (4.3), which contains variable speed to be identified and the reference model is the actual motor. [65]

So the state space axis stator current model is chosen as the state variable:

The adjustable model generates the estimated states is defined as

$$\frac{d}{dt}\begin{bmatrix} \hat{x}_1 \\ \hat{x}_2 \end{bmatrix} = \begin{bmatrix} -\frac{R_s}{L_d} & \hat{\omega}_r \frac{L_q}{L_d} \\ -\hat{\omega}_r \frac{L_d}{L_q} & -\frac{R_s}{L_q} \end{bmatrix}\begin{bmatrix} \hat{x}_1 \\ \hat{x}_2 \end{bmatrix} + \begin{bmatrix} \frac{1}{L_d} & 0 \\ 0 & \frac{1}{L_q} \end{bmatrix}\begin{bmatrix} \hat{u}_1 \\ \hat{u}_2 \end{bmatrix} \tag{4.4}$$

The operator^ denote estimated value.

Where, $\hat{x} = \begin{bmatrix} \hat{i}_d & \hat{i}_q \end{bmatrix}^T$ of the estimated currents and $\hat{\omega}_r$ is to be estimated speed.

Then, define the estimator as:

$$\frac{d}{dx}[\hat{x}] = [\hat{A}][\hat{x}] + [B][\hat{u}] \tag{4.5}$$

Equation (4.4) is the modeling equation for the model reference adaptive system, where current and voltage are the inputs to the system. In this case, outputs of the adjustable model are two estimates of the stator currents.

The error of state variable is compute as $\varepsilon = x\text{-}\hat{x}$,

$$\begin{cases} \varepsilon_d = i_d - \hat{i}_d \\ \varepsilon_q = i_q - \hat{i}_q \end{cases} \tag{4.6}$$

Then the estimation error dynamics is given by subtracting the adjustable model, equation (4.4) from the state system (4.3), and the error function system can be defined in equation**:**

$$\frac{d}{dt}[\varepsilon] = Ax + Bu - \hat{A}\hat{x} - B\hat{u} = Ax - \hat{A}x - \hat{A}\hat{x} + \hat{A}x + Bu - B\hat{u}$$

$$\frac{d}{dt}[\varepsilon] = \left(A - \hat{A}\right)x + \hat{A}(x - \hat{x}) + B(u - \hat{u}) \qquad (4.7)$$

$$\frac{d}{dt}[\varepsilon] = \hat{A}[\varepsilon] + \left(A - \hat{A}\right)x + B(u - \hat{u})$$

Next, simplify errors of the state variables form as:

$$\frac{d}{dt}\begin{bmatrix}\varepsilon_d \\ \varepsilon_q\end{bmatrix} = \begin{bmatrix} -\frac{R_s}{L_d} & \widehat{\omega}_r \frac{L_q}{L_d} \\ -\widehat{\omega}_r \frac{L_d}{L_q} & -\frac{R_s}{L_q}\end{bmatrix}\begin{bmatrix}\varepsilon_d \\ \varepsilon_q\end{bmatrix} + \begin{bmatrix} \frac{L_q}{L_d} i_q \\ -\frac{L_d}{L_q} i_d - \frac{\lambda_{pm}}{L_q}\end{bmatrix}[\omega_r - \widehat{\omega}_r] \qquad (4.8)$$

Using equation (4.8) ,the stator error model of the IPMSM expressed in the d-q synchronous reference frame is given as follow:

$$\frac{d}{dt}[\varepsilon] = [A][\varepsilon] + [W] \qquad (4.9)$$

Where $[\varepsilon] = \begin{bmatrix}\varepsilon_d & \varepsilon_q\end{bmatrix}^T$ is the error stator vector ,$[A]$ is the state matrix and hence can be included in the linear time invariant forward block, $[W]$ is the out put vector of the feedback block defined as follow:

$$[A] = \begin{bmatrix} -\frac{R_s}{L_d} & \widehat{\omega}_r \frac{L_q}{L_d} \\ -\widehat{\omega}_r \frac{L_d}{L_q} & -\frac{R_s}{L_q}\end{bmatrix}, [W] = Bu = \begin{bmatrix} \frac{L_q}{L_d} i_q \\ -\frac{L_d}{L_q} i_d - \frac{\lambda_{pm}}{L_q}\end{bmatrix}[\omega_r - \widehat{\omega}_r] = G[\widehat{\omega}_r - \omega_r]$$

The error between the two models fed an adaptive mechanism to turn out the observed variable.

4.3 Adaptive of speed observer design

The principle of the MRAS observer, to compare the outputs of the reference and the adjustable model, then use a mechanism to make them equal to each other. To determine the error dynamics, Popov's stability criterion is used, which gives sufficient condition for asymptotic stability. Also, the convergence of the speed estimation is guaranteed by the this theory [66].

This concerns the stability properties of a class of feedback systems as shown in Fig.4.3. Indeed, the derivation of the error is composed of two terms. The first is linear and the second is nonlinear.

According to Popov super stability theory, if [61]

(1) Transition matrix $H(s) = D(sI - A)^{-1}$ is a strictly positive matrix,

(2) Popov integral inequality is $\eta(0, t_0) = \int_0^{t_0} v^T w dt \geq -\gamma_0{}^2,\ in\ which, \forall\, t_0 \geq 0\,, \gamma_0{}^2$ is a finite positive constant independent on initial time.

Where v is the input vector, w is the output vector of the feedback block, $\gamma_0{}^2$ is a limited positive number, then $\lim_{t\to\infty} e(t) = 0$.So far, the MRAS speed estimation is asymptotic stable.

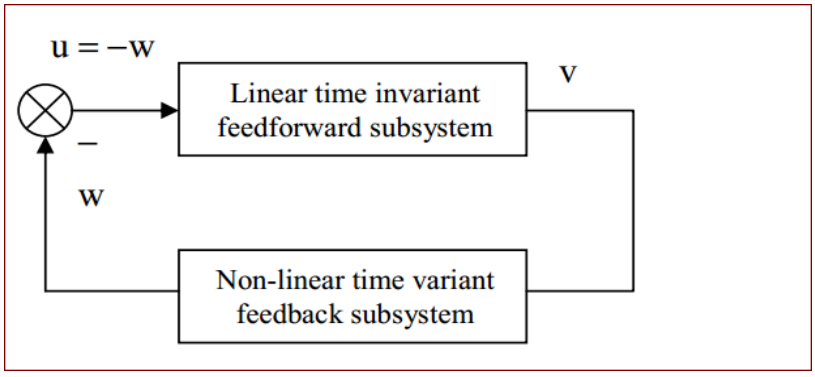


Figure 4.3- Standard Non-Linear Time varying feedback system

In figure 4.3 the input to the linear time-invariant system is u (which contains the stator voltage and currents), its output is v, which is the speed-tuning signal $v = [\varepsilon_d\ \ \varepsilon_q]^T$.The output of the non-linear time invariant system is w, and u = -w.

For designing the adaptive laws, the Popov's inequality criterion is made use of which consists of the following steps:

Step 1: Transform the MRAS into an equivalent system

Based on POPOV hyper stability criterion, identify the linear and nonlinear part in the system is necessary, where the system decompression must be as indicated in Fig.4.3.

First, to transform the MRAS into to a nonlinear time varying feedback system, which consists of a linear constant forward path and a non-linear feedback loop, is shown in Fig 4.4.
The equivalent forward square must be strictly positive real. Therefore, a linear compensator *D* is designed.
This describes a nonlinear feedback system:

$$\begin{cases} \dfrac{d\varepsilon}{dt} = [A]\varepsilon + I(-w) \\ v = D\varepsilon \end{cases} \qquad (4.10)$$

Where **w**=$(A - \hat{A})x + B(u - \hat{u}), choose$ **D** = **I** is the 2x2 identity matrix., $then$ $\boldsymbol{v}$ = Iε = ε.

$$W = \begin{bmatrix} \dfrac{L_q}{L_d} i_q \\ -\dfrac{L_d}{L_q} i_d - \dfrac{\lambda_{pm}}{L_q} \end{bmatrix} [\omega_r - \hat{\omega}_r] = G[\hat{\omega}_r - \omega_r]$$

Where, I is the 2x2 identity matrix.

Now, $\hat{\omega}_r$is a function of the state error and is time varying in nature, hence along with the adaption mechanism can be included in the nonlinear feedback system. Obviously, the stability and precision of the system is related to the construction of the adaptive mechanism.
The adaptation mechanism must be designed to ensure stability of the system. For this purpose, MRAS speed estimator generally can be represented by two basic subsystems, namely, linear time-invariant feed forward and nonlinear time-variant feedback subsystems. Then, the adaptation mechanism is designed according to Popov's hyper stability theory [67]- [68].

Thus to obtain the adaptation mechanism, first, the transfer function H(s) of the linear time-invariant feed-forward subsystem has to be obtained. It can be shown by lengthy calculations that in both of the schemes described in the following part this is strictly positive real. A possible proof uses the state variable form of the error equation, dv/dt = Av-w, which is obtained by subtracting the state variable equations of AM from the state-variable equations of the reference model (where A is the state matrix).

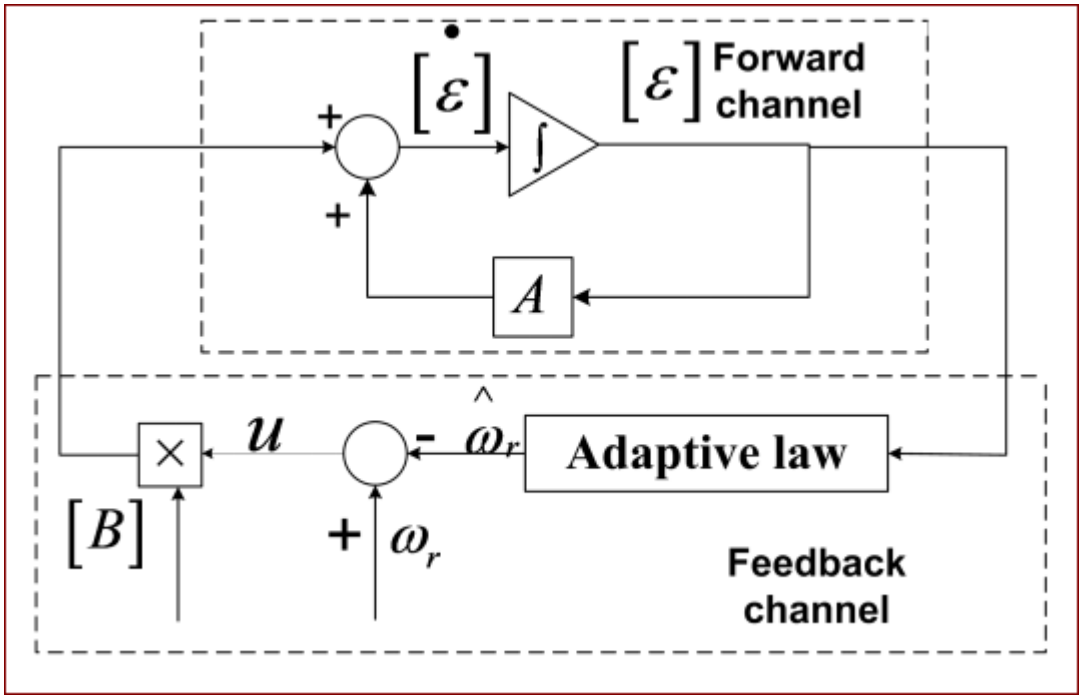


Figure 4.4: MRAS representation as a Non-Linear feedback system

Step 2: Ensure strictly positive real of the feed forward linear model

The second step is to ensure strictly positive real of the feed forward linear model. As to the strictly positive real of the feed forward linear model, a positive real lemma exists as described below. In formula (4.11),*D* could be obtained by lemma 1.

Lemma 1:

$$\begin{cases} \mathrm{PA} + \mathrm{A}^T\mathrm{P} = -Q \\ PI = D \end{cases} \tag{4.11}$$

If matrix Q, P is guaranteed as a positive definite matrix, Matrix D could make $H(s)$a strictly positive real matrix. Therefore, the adaptive system could be guaranteed as global asymptotic stability. Then ,H(s)is strictly positive real matrix ,if and only if, there exist a positive definite **s**ymmetric positive matrix P and Q.

Choosing $\mathrm{P}=\begin{bmatrix}1 & 0\\ 0 & 1\end{bmatrix}$ yields

$$Q = \begin{bmatrix} 2\dfrac{R_s}{L_d} & 0 \\ 0 & 2\dfrac{R_s}{L_q} \end{bmatrix} \tag{4.12}$$

Substituting the value of P and A to ensures that D is symmetrical positive definite and hence $H(s) = D(sI - A)^{-1}$is a strictly positive real transfer function and the feed forward linear model is strictly positive real.So that it is possible to calculate the transfer function the forward channel.

Transfer function:

$$H(s) = \frac{s + 2\frac{R_s}{L_d}}{S^2 + \left(\frac{R_s}{L_d} + \frac{R_s}{L_q}\right)S + \frac{{R_s}^2}{L_d L_q} + {\widehat{\omega}_r}^2} \qquad (4.13)$$

Substitute the value of motor parameter from appendix A:

$$H(s) = \frac{(s + 258.9826)}{S^2 + 186.105S + 7332.943 + {\widehat{\omega}_r}^2} \qquad (4.14)$$

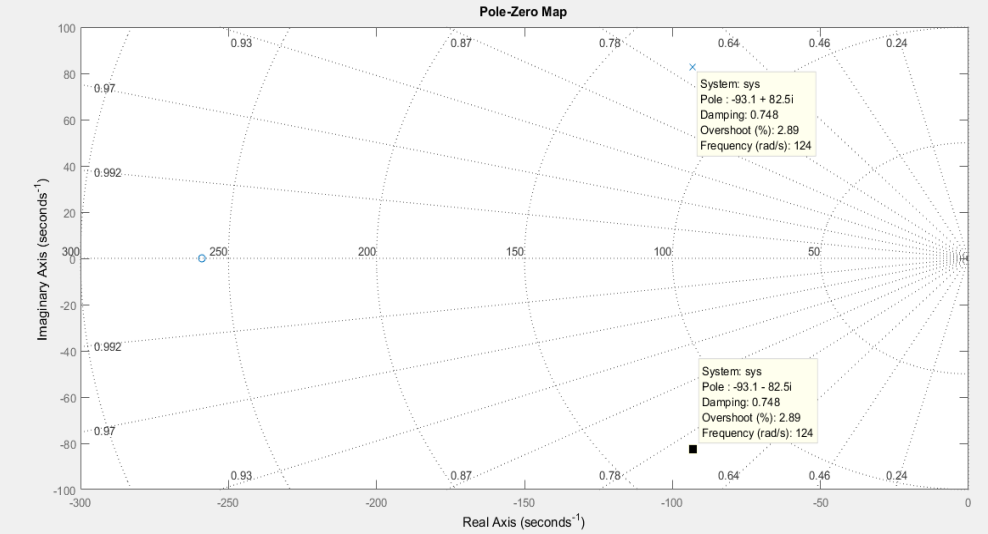


Figure 4.5: The pole-zero loci of H(s) about range of $\widehat{\omega}_r$starting at-100 up to 100 rad/s.

After obtain the H(s) of linear part, shown all the poles of the element for this matrix are with negative real parts. The value of ξ in general systems is 0.707 to calculated the $\widehat{\omega}_r = 100rad/s$,Its pole-zero loci is shown in Fig-4.5 for a range of $\widehat{\omega}_r$starting at -100 up to 100 rad/s. From the graph, show that with the increase of the speed, the poles also have negative real parts.

Step 3. Design the adaptive law

The third step isto design the adaptive laws which can ensure that the non linear feedback block (which includes the adaptation mechanism) satisfies the Popov's integral inequality. Effectively, the integral inequality of the non-linear feedback system must assure this inequality. The system is hyperstable if the forward path transfer matrix is strictly positive real and the input and output of the non-linear feedback block satisfies the Popov criterion [64]:

For the non-linear system to be stable, the following adaptation mechanism has been adopted in the observer. The asymptotic behavior of the adaptation mechanism is achieved by the simplified condition $\lim_{t\to\infty}[\varepsilon(\infty)]^T = 0$for any initialization.

$$\eta(0,t_1) = \int_0^{t_0} [\varepsilon]^T [w] dt \ \geq -{\gamma_0}^2 \quad for\ all\ t_0 \geq 0 \tag{4.15}$$

Where, γ_0 is a finite positive real

By using the Popov's theory, to obtain$\lim_{t\to\infty}[\varepsilon(\infty)]^T = 0$and therefore, the system of the MRAS speed estimation is asymptotically stable.

$$\eta(0,t_1) = \int_0^{t_0} [\varepsilon]^T [w] dt = [\varepsilon]^T \left[[A - \hat{A}\,][X] + B[u - \hat{u}]\right] dt \ \geq -{\gamma_0}^2 \tag{4.16}$$

Substituting for [ε] and [W] in this inequality from (4.10) into (4.15), using the definition of $\hat{\omega}_r$, Popov's criterion for the present system becomes:

$$\eta(0,t_1) = \int_0^{t_0} [\varepsilon^T][w] dt = \int_0^{t} \left[\frac{L_q}{L_d} i_q \varepsilon_d - \varepsilon_q \left(\frac{L_d}{L_q} i_d + \frac{\lambda_{pm}}{L_q}\right)\right] [\omega_r - \hat{\omega}_r]\, dt \geq -{\gamma_0}^2 \tag{4.17}$$

Consider system (4.17), one can easily prove that the observed rotor speed satisfies the following adaptation laws:

$$\text{let} \quad \hat{\omega}_r = f^1(\tau) + \int_0^{t_0} f^2(\tau) d\tau \tag{4.18}$$

Now find $\hat{\omega}_r$which satisfies (4.18) using the inegral inquality prperties and solution of this inquality can be found the following well-known relation [69]

Utilizing the following well-known integral inequality:

$$\int_0^{t_0} K f(t) \dot{f}(t) dt = \frac{1}{2} K[f^2(t) - f^2(0)] \geq -\frac{1}{2} K f^2(0) \text{k} > 0$$

$$\text{and} \int_0^{t} f^2(t)\, dt \geq 0 \tag{4.19}$$

Where

$$\begin{cases} f^2(\tau) = K_i \left[\frac{L_q}{L_d} i_q \varepsilon_d - \varepsilon_q \left(\frac{L_d}{L_q} i_d + \frac{\lambda_{pm}}{L_q} \right) \right] \\ f^1(\tau) = K_p \left[\frac{L_q}{L_d} i_q \varepsilon_d - \varepsilon_q \left(\frac{L_d}{L_q} i_d + \frac{\lambda_{pm}}{L_q} \right) \right] \end{cases} \quad (4.20)$$

The adaptive mechanism for the speed update is expressed as:

$$\hat{\omega}_r = K_{i\omega_{r\_est}} \int \left[\frac{L_q}{L_d} i_q \varepsilon_d - \frac{L_d}{L_q} i_d \varepsilon_q - \frac{\lambda_{pm}}{L_q} \varepsilon_q \right] dt + K_{p\omega_{r\_est}} \left[\frac{L_q}{L_d} i_q \varepsilon_d - \frac{L_d}{L_q} i_d \varepsilon_q - \frac{\lambda_{pm}}{L_q} \varepsilon_q \right] + \hat{\omega}_r(0) \quad (4.21)$$

Where, $K_{i\omega_{r\_est}}$ and $K_{p\omega_{r\_est}}$ are the PI speed observer controller and $\hat{\omega}_r(0)$is the initial estimated speed. The position of rotor can be estimated by integrating the estimated rotor speed.

$$\hat{\theta}_r = \int \hat{\omega}_r \, dt \quad (4.22)$$

4.4. PI parameter adjustment of speed observer

After constructed the adaptive law of the observer the transfer function of the adaptive PI-mechanism can be derived as [70]- [71].Hyper stability theory guarantees an asymptotically stable system if both the linear and non-linearparts satisfy a positivity condition. Usually, the input -W of the linear block equals the multiplication of the parameter error θ and the signal vector ξ used in the adaptation: $-W = \theta^T \xi$.So, the MRAS structure diagram can be obtained in Figure 4.6. which is constructed from a linear time-variant forward path transfer matrix and a nonlinear feedback block.

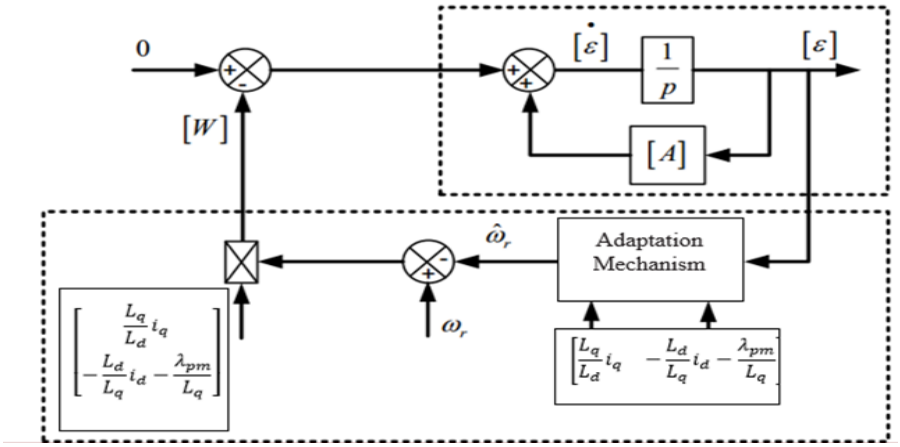


Figure 4.6.Equivalent scheme of the MRAS in the form of a closed looped system

A controllable, linear system with input u and output *v:*

$$\begin{cases} [\dot{\varepsilon}] = A[\varepsilon] + Bu \\ v = B^T \varepsilon \end{cases} \tag{4.23}$$

So, its transfer function is:

$$G_{p\_\hat{\omega}_r} = B^T (sI - A)^{-1} B$$

$$G_{p\_\hat{\omega}_r} = \frac{\left(s + \frac{R_s}{L_q}\right)\left(\frac{L_q}{L_d} i_q\right)^2 + \left(s + \frac{R_s}{L_d}\right)\left(\frac{L_d}{L_q} i_d + \frac{\lambda_{pm}}{L_q}\right)^2 + \left(i_d i_q + \frac{\lambda_{pm}}{L_d} i_q\right)\hat{\omega}_r \left(\frac{L_q}{L_d} - \frac{L_d}{L_q}\right)}{S^2 + \left(\frac{R_s}{L_d} + \frac{R_s}{L_q}\right) S + \frac{{R_s}^2}{L_d L_q} + {\hat{\omega}_r}^2} \tag{4.24}$$

The closed loop block diagram of the dynamic response of the rotor speed estimation for the MRAS structure diagram can be draw as in Fig.4.7.

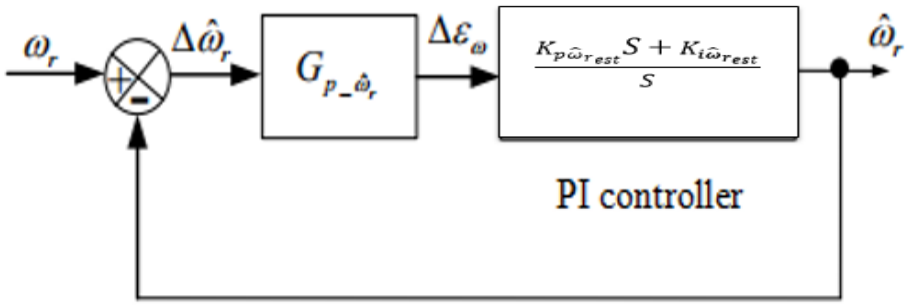


Fig. 4.7. Block diagram of closed-loop rotor speed estimator

So, this can be used to adjust the parameters of the PI controller by using the root locus analysis method for the simplified structure diagram of Fig-4.7 and make the system to achieve the desired results. Because the value of i_d and i_q are relatively small. So they can be neglected. This can get the open-loop transfer function of system.

$$G_{ol\_\hat{\omega}_r}(s) = \frac{\left(s + \frac{R_s}{L_d}\right) * \left(\frac{\lambda_{pm}}{L_q}\right)^2 * K_{p\hat{\omega}_r\_est}\left(s + \frac{K_{i\hat{\omega}_r\_est}}{K_{p\hat{\omega}_r\_est}}\right)}{S^3 + \left(\frac{R_s}{L_d} + \frac{R_s}{L_q}\right) S^2 + \left(\frac{{R_s}^2}{L_d L_q} + {\hat{\omega}_r}^2\right) s}$$

$$G_{ol\_\hat{\omega}_r}(s) = \frac{\left(s + \frac{R_s}{L_d}\right) k^* (s + z)}{S^3 + \left(\frac{R_s}{L_d} + \frac{R_s}{L_q}\right) S^2 + \left(\frac{{R_s}^2}{L_d L_q} + {\hat{\omega}_r}^2\right) s} \tag{4.25}$$

Where $k^* = \left(\frac{\lambda_{pm}}{L_q}\right)^2 K_{p\hat{\omega}_{r\_est}}$, $z = \frac{K_{i\hat{\omega}_{r\_est}}}{K_{p\hat{\omega}_{r\_est}}}$

The closed loop transfer function of the MRAS can be expressed as

$$\frac{\boldsymbol{\omega_r}}{\hat{\boldsymbol{\omega}}_{\boldsymbol{r}}} = \frac{G_{ol_{\hat{\omega}_r}}(s)\left(K_{p\hat{\omega}_{r_{est}}} + \frac{K_{i\hat{\omega}_{r_{est}}}}{S}\right)}{1 + G_{ol_{\hat{\omega}_r}}(s)\left(K_{p\hat{\omega}_{r_{est}}} + \frac{K_{i\hat{\omega}_{r_{est}}}}{S}\right)} \qquad (4.26)$$

The design of $K_{p\hat{\omega}_{r\_est}}$ and $K_{i\hat{\omega}_{r_est}}$is introduce to ensure stability, error tracking and robust operation. The design criteria for this adaptive PI controller is done using root locus by using the transfer function of the plant and set the time domain constraint. The constraint considered in this design is percent of overshoot; settling time and rise time are less than five percent, less than two second and less than two second respectively.

The design of $K_{p\hat{\omega}_{r\_est}}$ and $K_{i\hat{\omega}_{r_est}}$is selected to ensure that all the poles and zeros are located in the left hand side of s-plane and this allows for the required fast and stable response. The location of closed loop transfer function poles characterizes the control system dynamics. Therefore, the location of the PI controller zero (z=$\frac{K_{i\hat{\omega}_{r_est}}}{K_{p\hat{\omega}_{r_est}}}$) should be on the real axis to make sure that fast dynamics response.

The integral gain $K_{i\omega_{r_est}}$is chosen to be high for fast tracking of speed and a low proportional gain $K_{p\omega_{r\_est}}$ is noise rejection performance [68] thus determine the value of k^* and z.Therefore, $\frac{K_{i\hat{\omega}_{r\_est}}}{K_{p\hat{\omega}_{r\_est}}}$ is taken of 121 and from the root locus diagram of $G_{ol\_\hat{\omega}_r}(s)$ in (4.25) the maximum possible gain of 'k*' should be obtained from the specifications of settling time, maximum overshoot and rise times of the specification. In this case k* = 0.00134 and using the relation $k^* = \left(\frac{\lambda_{pm}}{L_q}\right)^2 K_{p\hat{\omega}_{r\_est}}$ then the value of $K_{p\hat{\omega}_{r\_est}}$ should become 0.26m and also $K_{i\hat{\omega}_{r\_est}}$= 121* $K_{p\hat{\omega}_{r\_est}}$= 0.032

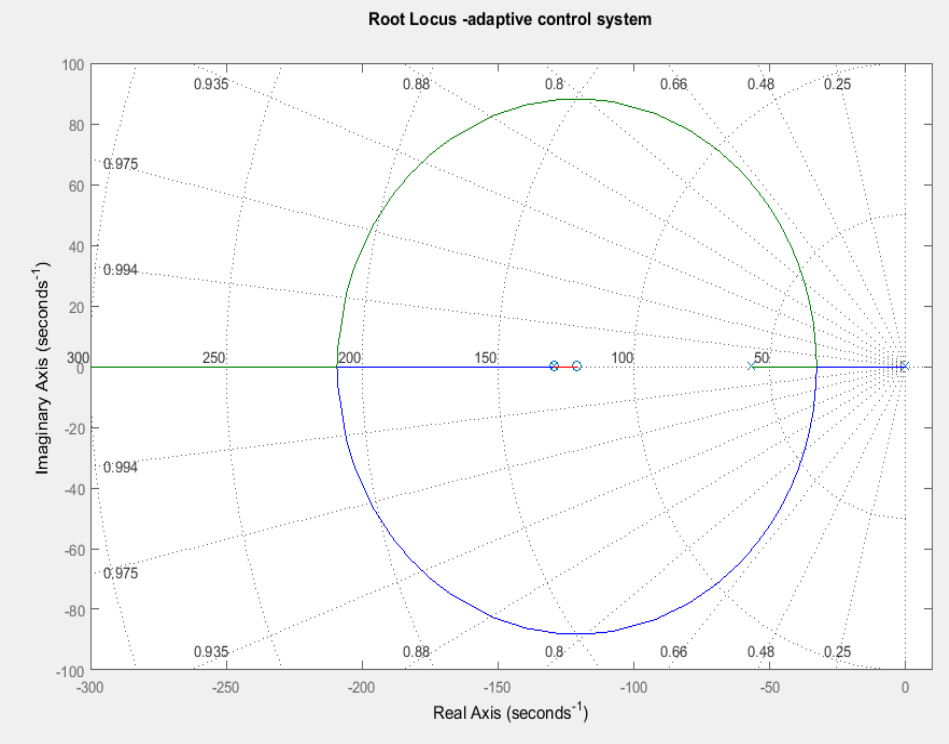


Figure 4.8 Root locus of closed loop rotor speed estimator

As shown in Figure 4.8 the design of $K_{p\hat{\omega}_{r\_est}}$ and $K_{i\hat{\omega}_{r_est}}$is selected to ensure that all the poles and zeros are located in the left hand side of s-plane and this allows for the required fast and stable response.

4.5. Neural Network Adaptive Controller

4.5.1 Introduction

Neural networks in computer world work on the human brain's mechanism of problem solving strategies. In a human brain, several linkages (connections) are provided through networks of axons and synapses to the computing elements called neurons. An artificial neuron is a computational model inspired in the natural neurons. Natural neurons receive signals through synapses located on the dendrites or membrane of the neuron. When the signals received are strong enough (surpass a certain threshold), the neuron is activated and emits a signal though the axon. This signal might be sent to another synapse, and might activate other neurons.

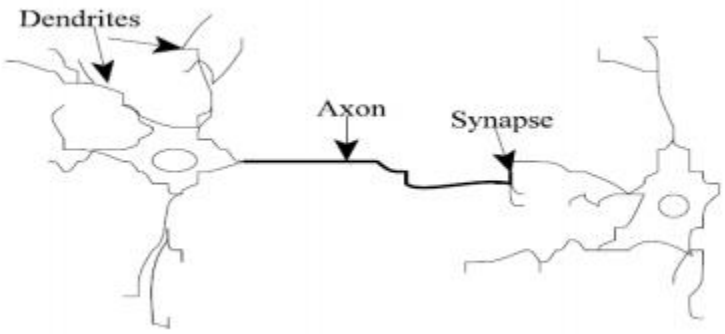


Figure 4.9. Components of a neuron.

The complexity of real neurons is highly abstracted when modeling artificial neurons. These basically consist of inputs (like synapses), which are multiplied by weights (strength of the respective signals), and then computed by a mathematical function which determines the activation of the neuron. Another function (which may be the identity) computes the output of the artificial neuron (sometimes in dependence of a certain threshold). ANNs combine artificial neurons in order to process information.

Recent progress in neural network theory provides us with new tools for modeling, identification and control of complex nonlinear dynamic systems while traditional and conventional control theory does the same for only linear systems. ANN can be developed by benefiting from the ability to model nonlinear systems with nonlinear structure controllers. In the drive systems, using ANN effect on the performance and durability of systems against to changing parameters.

Artificial NN can be applied successfully a motor whose load parameters are unknown [72]. RBFNN similar to the structure of feed-forward neural network algorithms are a powerful calculation instrument widely used in the identification field, pattern recognition and system modelling [73].Intelligent control, with special focus on neuron control can be used to solve difficult real control problems which are basically nonlinear, noisy and complex. This is because neural networks have an inherent ability to learn from input-output functions and approximation ability of different nonlinear functions [74].

The learning capability of an artificial neuron is achieved by adjusting the weights in accordance to the chosen learning algorithm and it develop with help of spreadsheet. Spreadsheets provide a popular and cost-effective alternative for teaching modeling and decision analysis techniques, but there currently exist few spreadsheet-based tools for teaching basic ANN concepts. [75]

4.5.2 Structure of the Neuron

The basic processing element of the connectionist architecture is often called neuron by analogy with neuron physiology.

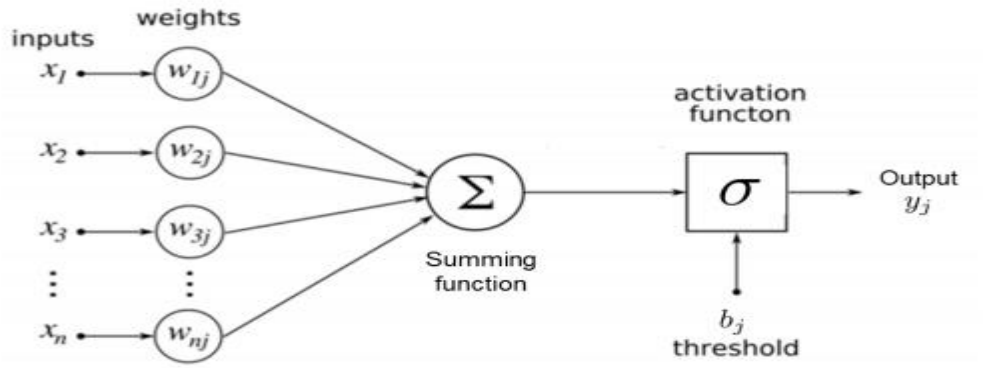


Figure 4-10. Nonlinear model of a neuron

A neuron is the fundamental unit for the operation of a NN.

In mathematical terms, neural model is shown in Fig.5.2 with three basic elements: [76]- [77]

1. A set of synapses or connecting links, with each element characterized by its own weight,

2. A weight summer or an adder for summing the input signals, weighed by the respective synapses of the neuron.

3. Anon dynamical, nonlinear activation function transforming the adder output into the output of the neuron which is also called activation function, use for limiting the amplitude of the output of a neuron.

In mathematical terms, the neuron can be described as

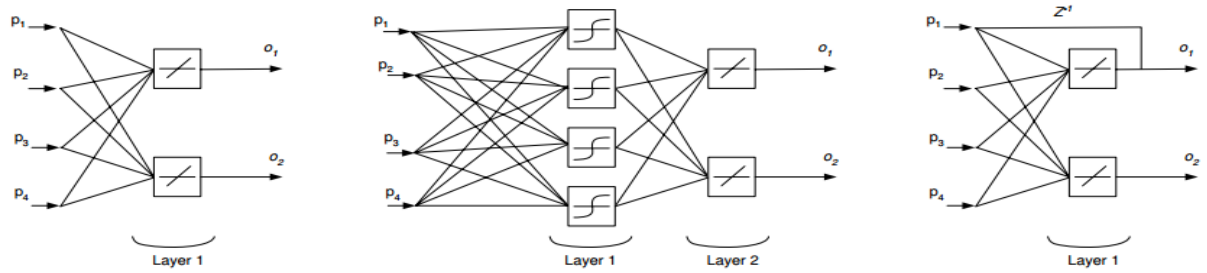
$$y_j = \sigma\left(\sum_{i=1}^{n} w_{ij}x_i + b_j\right) \qquad (4.27)$$

Where$x_1, x_2, \ldots\ldots x_n$are the input signals, $w_{1j}, w_{2j}, \ldots\ldots, w_{nj}$are the respective synaptic weights of neuron, j,$b_j$is the bias,σ (·)is the activation function, and$y_j$is the output signal of the neuron. The activation function σ(·)is often chosen as hard limit, linear threshold, sigmoid, hyperbolic tangent, augmented ratio of squares, or radial basis functions.

4.5.3. The Network Architecture

The neural network also can be viewed as a weighted directed graph in which artificial neurons are nodes and directed weighted edges represent connections between neurons. Three main types of ANN structures can be distinguished: single layer feed forward network, multilayer feed forward network and recurrent networks, as shown in Figure 4.11 [78].The most common type of single layer feed forward network is the perceptron.

The multilayer feed forward network configuration is characterized by a flow of information from the inputs to the outputs with no feedback in the layers. Increasing the number of neurons in the hidden layer or adding more hidden layers to the feed forward network allows the network to deal with more complex functions. Each neuron is connected only to neurons in the next layer and there is no connection between neurons in the same layer. Cybenko's theorem states that, "A feed forward neural network with a sufficiently large number of hidden neurons with continuous and differentiable transfer functions can approximate any continuous function over a closed interval [79].



(a).Single layer feed forward (b).Multilayer feed forward (c) Recurrent

Figure 4.11. Architectures for neural networks.

The second type of multi-layer network is recurrent. The recurrent neural network distinguishes itself from feed forward neural networks in that it has at least one feedback loop. It so that the output of the neuron can be fed back to the inputs of other neurons in the same or previous layers [80]In addition, some recurrent networks have unit delay elements denoted by Z^{-1}(recurrent-dynamic network). This gives the network partial memory due to the fact that the hidden layers and the input layer receives data at time (t) that includes information from (t-p), where (p)is the number of unit delays. This makes recurrent networks powerful in approximating functions depending on time.

4.6 RBF neural networks

4.6.1 Structure and Algorithms of Radial Basis Function

A Radial basis function (RBFNN) neural network is a three-layer feed forward network with single hidden layer [81], such as input layer, hidden layer and output layer. In RBF network is, the hidden layer space is constructed using a nonlinear function, so an input vector can be directly (do not need the weights) mapped into hidden space and the outputs of network are linear weighted sum of hidden unit output [82]. Also output of each hidden neuron depends only on the radial distance between the input vector and the center of the hidden neuron. The mapping relationships between hidden layers to output layers were described as a linear function, so it can greatly accelerate the learning speed and avoid local minimum problem could be avoided in the same way as multilayer perceptrons [83].

Hence the RBFNN is used for tuning the PI parameters in the control strategy.Generally,It is widely used as a powerful computational technique for pattern recognition, function approximation, system control and modeling and identification field because of its simple structure and rapid learning algorithm [73]. The most popular choice for the non-linearity is the Gaussian. The output layer is in regression problems a linear combination of hidden layer values representing mean predicted output [84]

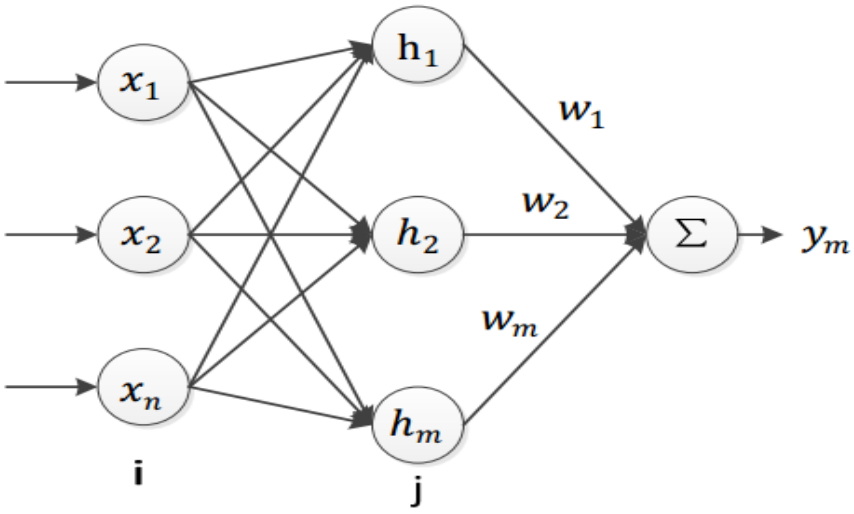


Figure 4.12 RBF neural network structure.

The thought is to imitate the system arrangement of restricted re-arranging in the human mind. Whereas; “near the center of the Gaussian Kernel function the RBF neural network produces the strongest response”. In the non-linearity in nature, the RBF–NN can be on purpose as a restricted estimate replica of controlled procedure. Also, input samples for RBF -NN doesn't need a special distribution. The inputs are assigned to the neurons in the input layer directly and the outputs are also taken from the output layer neurons directly. Thus in general, the number of neurons in input and output layer is equal to the number of inputs and outputs respectively. Furthermore, RBF is converges rapidly and on-line learning. Whereas, from universal block drawing of RBF network structure is exposed in Fig. 4.12

4.6.2 Design of adaptive PI controller based on RBFNN

The structure of the RBF neural network's block diagram is designed in this thesis, based the adaptive PI algorithm is shown in figure 4.13. Neural network adaptive PI controller adjusts the connection weights of neural network and the two parameters of PI according to the square error of the given input and system output as the objective function [85].The PI controller is applied to the controlled object, and makes the system output close to the given input of system.

A. Network output calculation

In the RBF network structure, X $= [x_1, x_2, \dots \dots x_n]^T$ is the input vector of the first layer . The Euclidean distance between the center c_j and the network input vector X is defined by $\|X - C_j\|$. The output vector of hidden layer, namely Radial basic vector is $\boldsymbol{h} = [h_1, h_2, \dots h_m]^T$ and h_j is Gaussian function with the following mathematical relation is [86]- [87].

$$h_j = exp\left[-\frac{\|X - C_j\|^2}{2{b_j}^2}\right], (\text{j} = 1,2, \dots, m \quad) \tag{4.28}$$

Where, $b_j = [b_1, b_2, \dots \ b_j \dots . b_m]^T$ are the basis width Parameter of j^{th} neuron in the hidden layer and j is the number of neurons in hidden layer, which determines the width basis function around the center, the weight vector from hidden layer to output layer is $\text{w} = [w_1, w_2, \dots w_m]^T$ and center vector of the network at node j $\boldsymbol{c_j} = [c_{j1}, c_{j2}, \dots, c_{ji} \dots . c_{jn}]^T$,($i$=1,2,3,…$n$).

The connections in the second layer are weighted and m-th neuron output network y_m is obtained by the weighted summation of the outputs of all hidden neurons connected to that output neuron:

$$y_m(k) = \sum_{j=1}^{nh} w_{m,j} h_j(X) \tag{4.29}$$

Where, nh is the number of the hidden layer neurons , $w_{m,j}$ is the weight link between j-th neuron of the hidden layer and the m-th neuron of the output layer and $h_j(X)$ is a radially symmetric scalar function with nh centers of the radial basis function. A commonly used radial basis function $h_j(.)$ is a Gaussian function.

B.PI Control Based on RBF

PI controllers have been widely used for level control because of its practicality and good control performance and the PI controller is selected as the main part of the proposed control strategy for the level control. The principle of novel control strategy is to combine the advantages of the conventional PI control method and advanced control technique (Radial Basis Function neural network) [88].

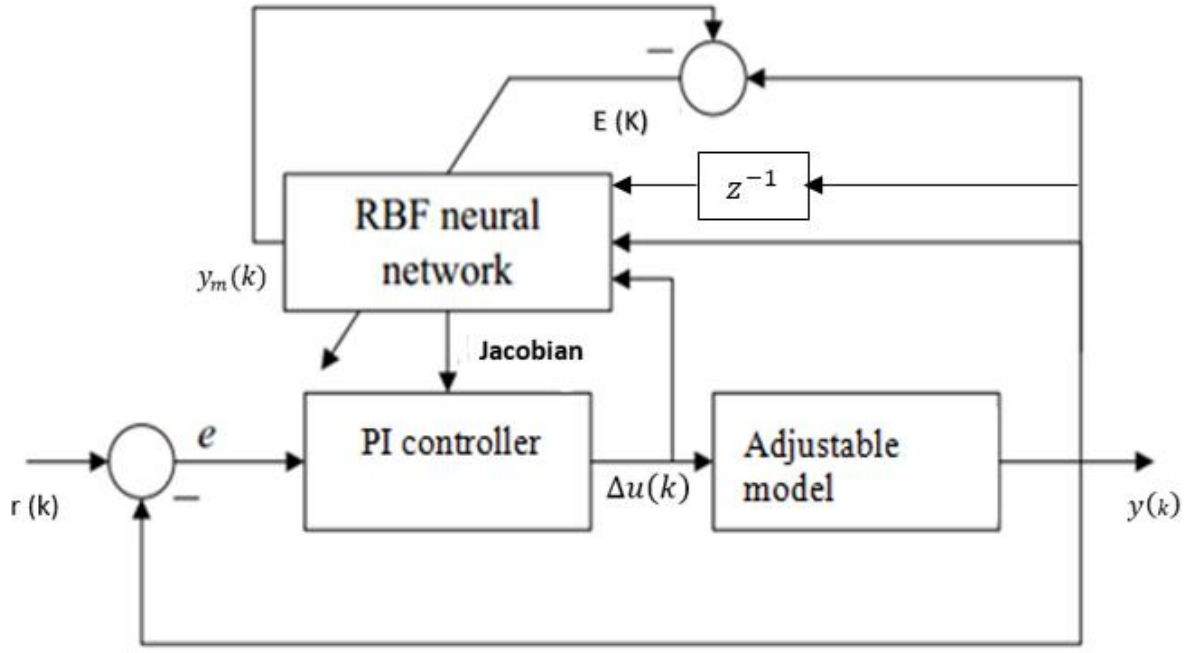


Figure 4.13. Block diagram of RBF neural network-PI controller

The designed RBF network has three layers: an input layer, a single hidden layer and an output layer as shown in Figure 4-14.

There are three inputs in this network and the input vector of the RBF network is given as:

$$\mathrm{X} = [y(k), y(k-1), \Delta u(k),]^T \qquad (4.30)$$

RBF neural network use 3-6-1 network structure. The input of the RBF neural networks are selected as, $\mathrm{X} = [\Delta u(k), y(k)]^T$ and the output of RBF neural network acts as traditional PI controller with self-adaptive capability through networks learning and training.

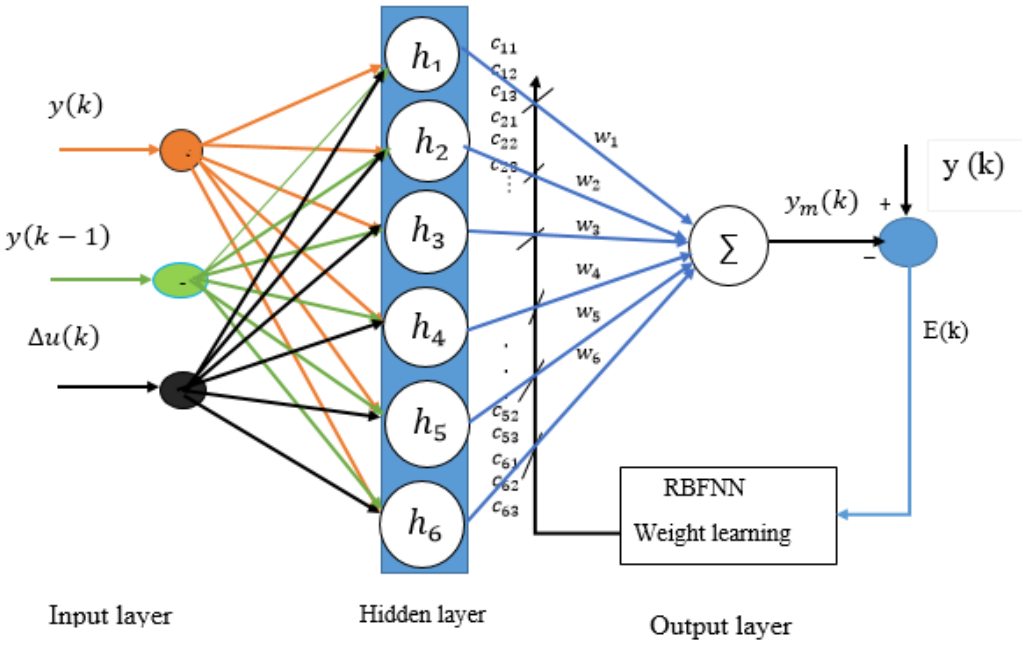


Figure 4-14 Architecture of the RBF neural network

In a RBF neural network, the Gaussian function is used as the activation function. The hidden neurons implement Gaussian function as the basis function and the elements of radial basis vector $h = [h_1, h_2, . h_j .. h_6]^T$ can be expressed by Gaussian function as follows:

$$h_j = exp\left[-\frac{\|X - C_j\|^2}{2{b_j}^2}\right], \{j = 1,2, \dots ,6\} \qquad (4.31)$$

where X is the input vector of the neural network as given by equation (4.30), $C_j = [c_{j1}, c_{j2}]^T$ is the input vector of the j^{th} node in the hidden layer, b_j is the width parameter of the j^{th} node in hidden layer and j is the number of neurons in hidden layer.

Thus the network output can be express as follows:

$$y_m(k) = w_1 h_1 + w_2 h_2 + \cdots + w_6 h_6 \qquad (4.32)$$

Where w_j is the weight from hidden layer to output layer.

C. Network Learning algorithm

The performance index function of the RBF neural network is set as follows:

$$e_m(k) = \frac{1}{2}[y(k) - y\_m\ (k)]^{\,2} \qquad (4.33)$$

Where,$y(k) = e_q$ is the adaption mechanism output signal at time k and $y_m(k)$, is the output of is the output of the neural network of time k . This process will take place and be continuously updated with the aim of finding the minimum error between $y(k)$ and $y_m(k)$. For the machine learning model, the Gradient Descent method is the most basic and easy to use.

As shown in figure 4.13, the [$E(k)$] is used to adjust the weights of the neural network .In order to minimize the error ,a gradient descent method, the iteration algorithm to calculate the variation of the weight of the output layer, the center node, and the node base is as follows [87].

$$\begin{cases} \Delta w_j(k) = -\eta \dfrac{\partial E(k)}{\partial w_j} = -\eta \dfrac{\partial E(k)}{\partial y_m} \dfrac{\partial y_m}{\partial w_j} = \eta\big[y(k) - y_m(k)\big] h_j \\ w_j(k) = w_j(k-1) + \Delta w_j(k) + \alpha\big[w_j(k-1) - w_j(k-2)\big] \end{cases} \qquad (4.34)$$

$$\begin{cases} \Delta b_j(k) = -\eta \dfrac{\partial E(k)}{\partial b_j} = \eta\big[y(k) - y_m(k)\big] w_j h_j \dfrac{\|X - C_j\|^2}{{b_j}^3} \\ b_j(k) = b_j(k-1) + \Delta b_j(k) + \alpha\big[b_j(k-1) - b_j(k-2)\big] \end{cases} \qquad (4.34)$$

$$\begin{cases} \Delta c_{ji}(k) = -\eta \dfrac{\partial E(k)}{\partial c_{ji}} = \eta\,\big[y(k) - y_m(k)\big] w_j \dfrac{x_i - c_{ji}}{{b_j}^2} \\ c_{ji}(k) = c_{ji}(k-1) + \Delta c_{ji}(k) + \alpha\big[c_{ji}(k-1) - c_{ji}(k-2)\big] \end{cases} \qquad (4.35)$$

Where, Δb_j is the change of the basis width parameter, Δc_{ji} is the change of hidden layer's input vector, $\eta \in (0,1)$is the learning rate, $\alpha \in (0,1)$ is the momentum factor .

For the adjustment of the weights of the neural network, its need a parameter called learning rate η. If η is too large, neural network will be unstable; but if η is too small, the convergence rate will be too slow. Therefore, the selection of the value of learning rate is crucial to the stability and convergence of the system.

And Jacobian matrix:

$$\frac{\partial y(k)}{\partial \Delta u(k)} \approx \frac{\partial y_m(k)}{\partial \Delta u(k)} = \sum_{j=1}^{m} w_j h_j \frac{c_{j1} - x_1}{{b_j}^2} \tag{4.36}$$

And $x_1 = \Delta u(k) = \widehat{\omega}_r\_est$ is the control increment which is defined as the first input of the neural network.

RBF neural network identifier through the input and output data of controlled object to identify the approximate model of the controlled object, and finally to make the output data of the neural network to approximate the output data of controlled object. The Jacobian generalizes the gradient of a scalar-valued function of multiple variables, which itself generalizes the derivative of a scalar-valued function of a single variable. The output response of neural network is $y_m(k)$ will approach the system output $y(k)$.

Therefore $\frac{\partial y(k)}{\partial \Delta u(k)}$, is closed to $\frac{\partial y_m(k)}{\partial \Delta u(k)}$

$$\frac{\partial y(k)}{\partial \Delta u(k)} \approx \frac{\partial y_m(k)}{\partial \Delta u(k)} = \frac{\partial y_m(k)}{\partial x_1} = \frac{\partial}{\partial x_1} \sum_{j=1}^{m} w_j e^{-\frac{\|X - C_j\|^2}{2{b_j}^2}} = \sum_{j=1}^{m} w_j h_j \frac{\partial}{\partial x_1} \left(-\frac{\|X - C_j\|^2}{2{b_j}^2} \right)$$

$$\frac{\partial y(k)}{\partial \Delta u(k)} = \sum_{j=1}^{m} w_j h_j \frac{\partial}{\partial x_1} \left(-\frac{X^T X - X^T C_j - {C_j}^T X + {C_j}^T C_j}{2{b_j}^2} \right)$$

$$\frac{\partial y(k)}{\partial \Delta u(k)} = \sum_{j=1}^{m} w_j h_j \left(-\frac{2x_1 - 2c_{j1}}{2{b_j}^2} \right) = \sum_{j=1}^{m} w_j h_j \frac{c_{j1} - \Delta u(k)}{{b_j}^2}$$

$$\frac{\partial y(k)}{\partial \Delta u(k)} = \sum_{j=1}^{m} w_j h_j \frac{c_{j1} - \Delta u(k)}{{b_j}^2} \tag{4.37}$$

Where $\frac{\partial y}{\partial \Delta u}$ the identification information for the Jacobian matrix of the controlled object and it can be obtained through the identification process of neural network.

D.Control Principle of RBF-PI

The PI controller can have excellent performance with proper parameters, otherwise the controller cannot achieve desired control requirement. Thus, the regulating the PI parameters properly is an important control and the designed network is able to tune k_p and k_i accurately in

different situations by using Jacobian matrix [89]. It has been known that the PI control performance is based on the value of PI parameters k_p and k_i.

The control error of PI controller is given as following:

$$e(k) = r(k) - y(k) = e_d(k) - e_q(k) \tag{4.38}$$

Where , $r(k) = e_d$ is the desired process value or set point of reference model and y(k)= e_q is aadaption mechanism model output signal.

The input of the PI controller in this control strategy, which are defined as:

$$\begin{cases} xc_1(k) = e(k) - e(k-1) \\ xc_2(k) = e(k) \end{cases} \tag{4.39}$$

Where, $xc_1(k)$ and $xc_2(k)$ are defined as the propositional and integral error respectively.

Then incremental PI controller output is the updating algorithm as:

$$\begin{cases} u(k) = u(k-1) + \Delta u(k) \\ \Delta u(k) = k_p xc_1(k) + k_i xc_2(k) \end{cases} \tag{4.40}$$

Where, k_p and k_i are the parameter of PI controller.

Thus, the updating algorithm for the adaptive PI controller based on RBF can be derived as:

$$\begin{cases} k_p(k) = k_p(k-1) + \Delta k_p \\ k_i(k) = k_i(k-1) + \Delta k_i \end{cases} \tag{4.41}$$

Neural network-tuning of indicators is as follows:

$$E(k) = \frac{1}{2} e(k)^2 \tag{4.42}$$

Then $k_p$and k_i is designed based on gradient descent method is applied with the chain rule to minimize the performance index function $E(k)$ as follows:

Then $\Delta k_{pi}(k)$ are the increment of the gain vector $k_{pi}(k)$, adjusted based on the negative gradient method is applied with the chain rule to minimize the performance index function $E(k)$ as follows:

$$\begin{cases} \Delta k_p = -\eta_p \dfrac{\partial E}{\partial k_p} = -\eta_{pi} \dfrac{\partial E}{\partial y}\dfrac{\partial y}{\partial \Delta u}\dfrac{\partial \Delta u}{\partial k_p} = \eta_p e(k) \dfrac{\partial y}{\partial \Delta u} xc_1(k) \\ \Delta k_i = -\eta_i \dfrac{\partial E}{\partial k_i} = -\eta \dfrac{\partial E}{\partial y}\dfrac{\partial y}{\partial \Delta u}\dfrac{\partial \Delta u}{\partial k_i} = \eta_i e(k) \dfrac{\partial y}{\partial \Delta u} xc_2(k) \end{cases} \quad (4.43)$$

where $\frac{\partial y}{\partial \Delta u}$ is the Jacobian matrix reflects the sensitivity of the output of the controlled object $y(k)$ to controller output Δu(k) is given by equation (4-43) and η_{pi} is the learning rate for the adaptive PI algorithm.

Thus, the updating algorithm for the adaptive PI controller based on RBF can be derived as:

$$\begin{cases} \Delta k_p = -\eta_p \dfrac{\partial E}{\partial k_p} = -\eta_{pi} \dfrac{\partial E}{\partial y}\dfrac{\partial y}{\partial \Delta u}\dfrac{\partial \Delta u}{\partial k_p} = \eta_p e(k) xc_1(k) \sum_{j=1}^{m} w_j h_j \dfrac{C_{j1} - \Delta u(k)}{{b_j}^2} \\ \Delta k_i = -\eta_i \dfrac{\partial E}{\partial k_i} = \eta_{pi} e(k) \dfrac{\partial y}{\partial \Delta u}\dfrac{\partial \Delta u}{\partial k_i} = \eta_i e(k) xc_2(k) \sum_{j=1}^{m} w_j h_j \dfrac{C_{j1} - \Delta u(k)}{{b_j}^2} \end{cases} \quad (4.44)$$

The parameters of PI are updated based on the following rule:

$$k_{pi}(k) = \begin{cases} k_{pi}(k-1) & if \ \left\| \dfrac{\partial y}{\partial \Delta u} \right\| \geq \gamma \\ k_{pi}(k-1) + \Delta k_{pi}(k), & otherwise \end{cases} \quad (4.45)$$

$\gamma = 0.0085$ is a threshold, by which the tuning Equation (4.45) can prevent PI gain parameters from being increased rapidly due to the effect of external strong interference.

$\Delta k_{pi}(k)$ is the increment of the gain vector $k_{pi}(k)$, adjusted based on the negative gradient method as follows

$$\begin{cases} \Delta k_p = -\eta_{pi} \dfrac{\partial E}{\partial k_p} = -\eta_p \dfrac{\partial E}{\partial y}\dfrac{\partial y}{\partial \Delta u}\dfrac{\partial \Delta u}{\partial k_p} = \eta_p e(k) xc_1(k) \sum_{j=1}^{m} w_j h_j \dfrac{C_{j1} - \Delta u(k)}{{b_j}^2} \\ \Delta k_i = -\eta_{pi} \dfrac{\partial E}{\partial k_i} = \eta_i e(k) \dfrac{\partial y}{\partial \Delta u}\dfrac{\partial \Delta u}{\partial k_i} = \eta_i e(k) xc_2(k) \sum_{j=1}^{m} w_j h_j \dfrac{C_{j1} - \Delta u(k)}{{b_j}^2} \end{cases} \quad (4.46)$$

Therefore, the optimum PI parameters can be gained and then provided to the PI controller in order to get the desired control performance.

For most systems, its signal is definite $\frac{\partial y}{\partial \Delta u}$ is replace with sgn($\frac{\partial y}{\partial \Delta u}$) , and learning speed η is used to equalize the calculate error [90].

4.6.3. RBFNN Training Algorithm

Training of a RBF network involves the determination of the number of centers C, the widths of the receptive fields B, and the connection weights to the output layer W. The standard technique used to train an RBF network is the hybrid approach, which is a two-stage learning strategy. At first stage, some unsupervised training procedures are used to adjust the parameters of the radial basis functions (centers and widths). One may select centers by random selection of fixed centers, self-organized selection e.g. using k-means clustering algorithm and supervised selection, among others. Indeed, the choice fixed-center selection method. The fixed centers algorithm adapts only the weights between the hidden and output layers.Centers can be initialized in an unsupervised training step prior to training the weights between hidden units (radial basis) and output units.

In the second stage, the weights of the output layer are adapted applying the supervised training algorithms. It defines a cost function to be rninimized using an error-correction learning scheme such as a gradient-descent method, least-squares method, orthogonal least squares [91]- [92]. Gradient descent can be used to adjust weights, centers, and widths.

- The steps of the proposed RBFNNPI controller strategy are as follows

Step1: Initializing the network parameter, include number of nodes in input and hidden layer, learning rate, the base width and the weight vector.

Step2: Sampling to get input r(k) and output calculating error in terms of Equation(4.38).

Step3: Calculating network output y_m, adjustment center vector [C], base width vector [B], and Weight vector [W] from Eqn. (4.34) -(4.36) , the output [u]of controller according to Eqn.(4.40) and Jacobian matrix in terms of Eqn. (4.37) to obtain network identification information.

Step4: Adjusting parameter of the regulator in terms of Eqn. (4.43).

Step5: Return to Step 2 and repeat the calculations until e_m is less than the desired error.

The training algorithm for RBF is given as follows:

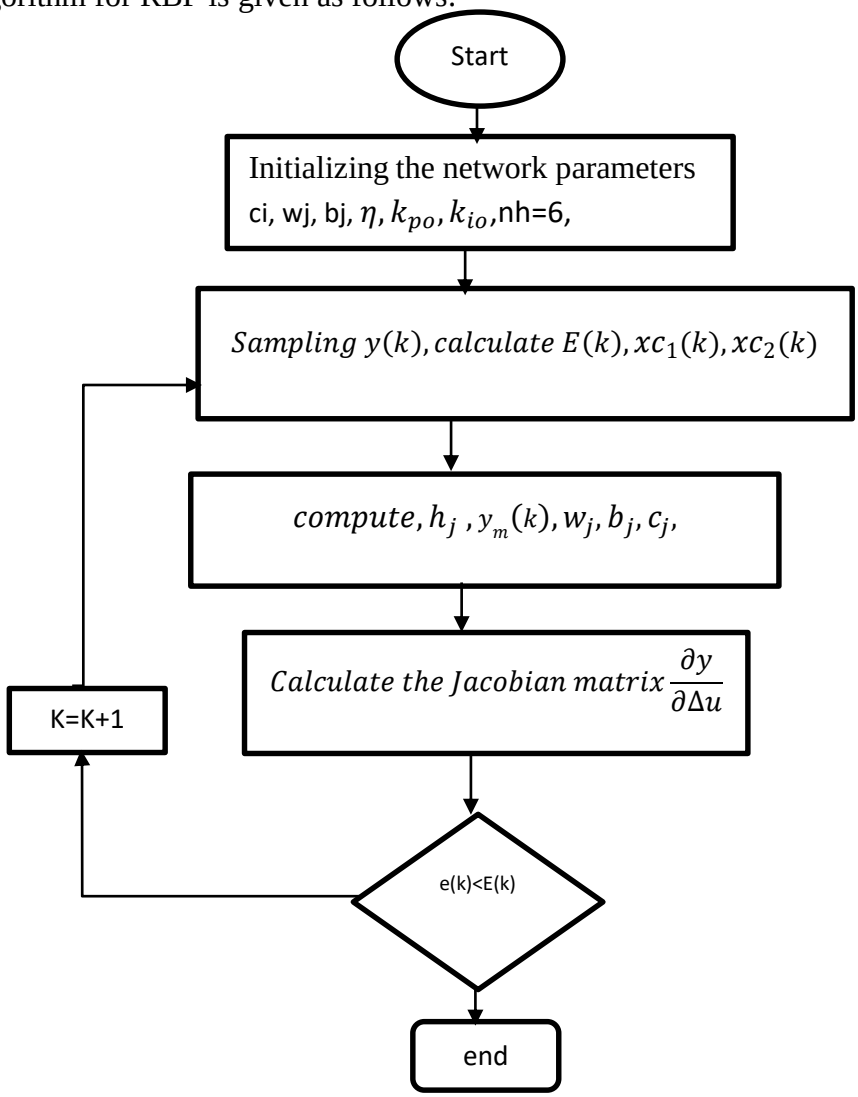


Figure 4.15. Flow chart of RBF neural network algorithm

All the training algorithm is given as follows:

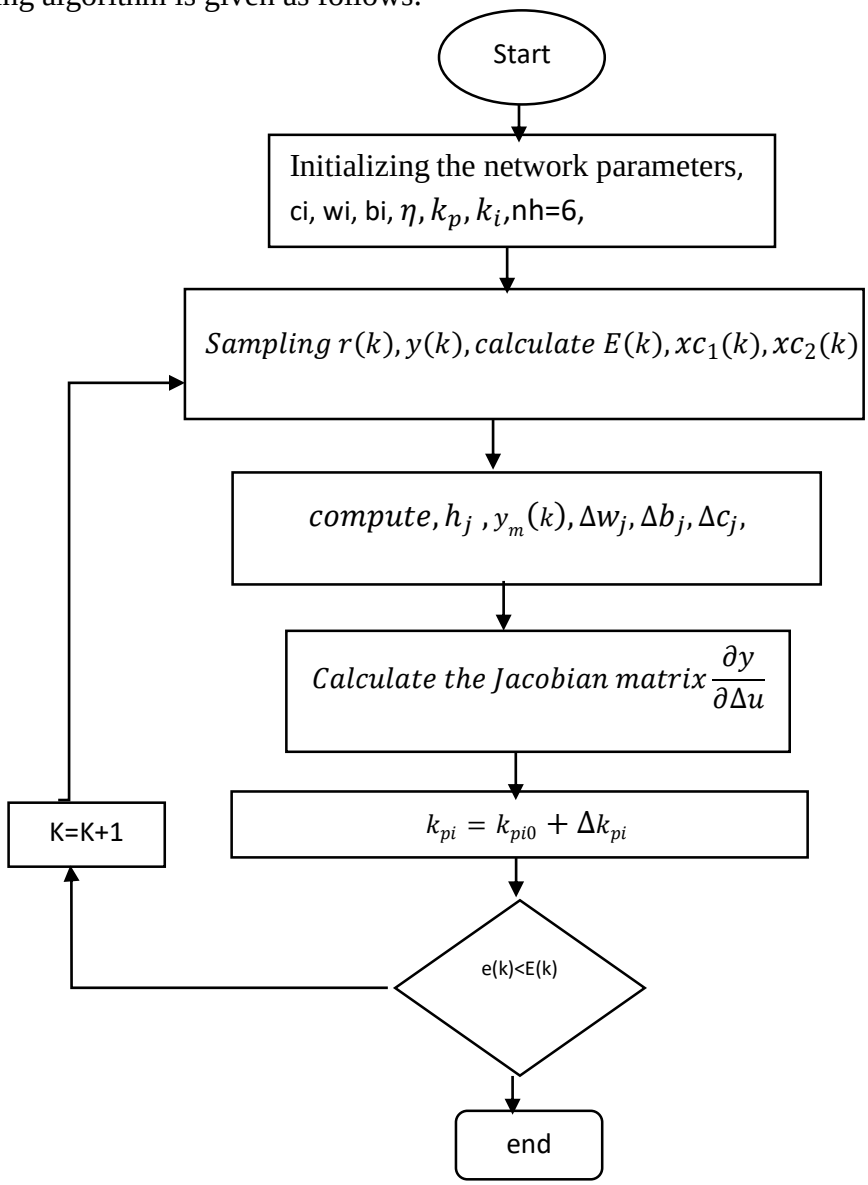


Figure 4.16. Flow chart of RBFNN-PI controller

The controller block diagram was implemented according to fig.4.14 with the RBF-NN controller parameters that are given in Table 6.

Table 6: The network parameter

Symbols	Meaning	Value
c_{ji}	Initial center matrix of Gaussian functions	$\begin{bmatrix} 30 & 30 & 30 & 30 & 30 & 30 \\ 30 & 30 & 30 & 30 & 30 & 30 \\ 30 & 30 & 30 & 30 & 30 & 30 \end{bmatrix}$
b_j	Initial widths vector of Gaussian functions	$[20 \quad 20 \quad 20 \quad 20 \quad 20 \quad 20]^T$
w	Initial weight vectors	$[10 \quad 10 \quad 10 \quad 10 \quad 10 \quad 10]^T$
$[k_p \quad k_i]^T$	Initial $k_p \quad k_i$	$[0.00026 \quad 0.032]^T$
i	Number network inputs	3
j	Number of hidden neurons	6

The training data points generate form simulation part input and out put behavior of Adaptition mechanism under similar refernce inputs and the architecture comprises three neurons in the input layer, six on the hidden layer and two neurons linear activation function are used in the output layer. In order to describe the proposed design from tuning controller system in 4.6.2 section. First, choosing the initial parameter values that are suitable for the system, simulating for training using the MATLAB code shown in Appendix B.

CHAPTER FIVE

5. Simulation Results and Discussion

5.1 Introduction

In this thesis, In order to compare and verify the performance of the proposed control are applied to the model of SVPWM inverter-fed IPMSM drive system is built in MATLAB/SIMULINK platform. Then build the adjustable model and adaptive law. The parameters that are used for simulating the IPMSM drive system placed in appendix A of Table A I. Simulation studies are carried out with a 5-kHz control frequency.

5.2. Simulink Model of PMSM Drive System

The overall Matlab/Simulink model of the PMSM drive and Control system is shown in Fig. 5.1 The overall system simulation block includes different sub -functional blocks: PI controller blocks, voltage decoupling block ,coordinate transformation blocks, MRAS block and SVPWM block are main sub-functional blocks.

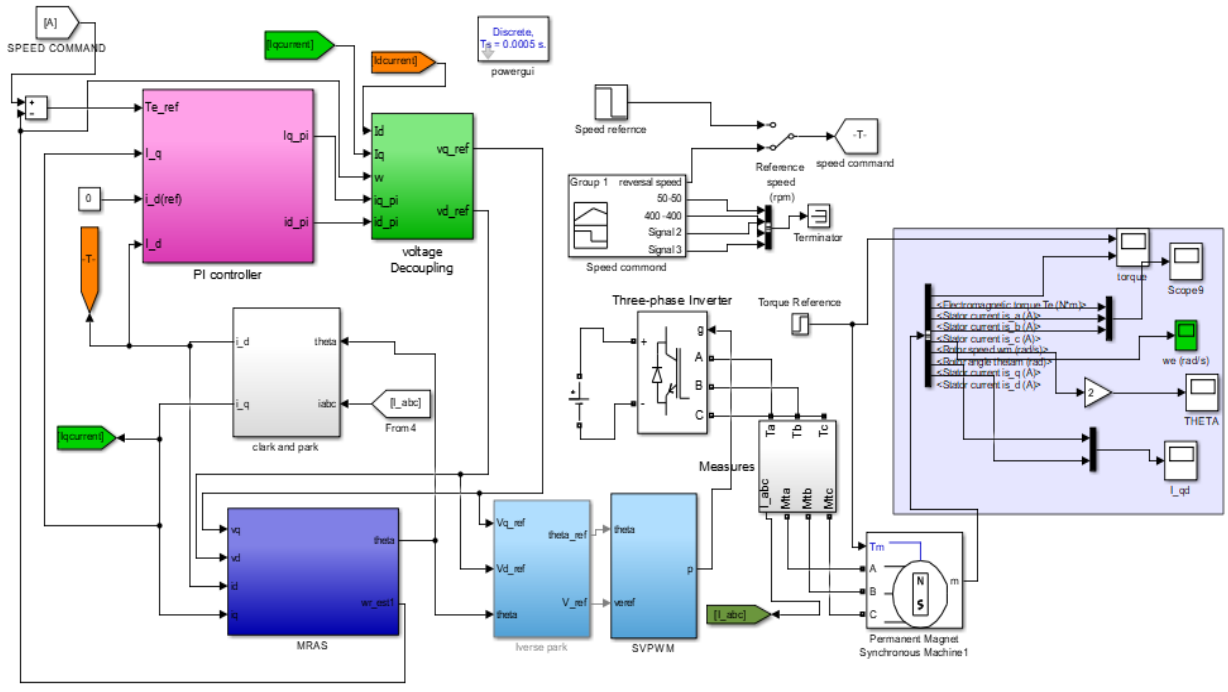


Figure 5.1: Overall observer based speed control of IPMSM system simulation block diagram

From the overall block diagram the subsystem underneath the mask, take values of the three phase currents to compute the direct axis and quadrature axis currents the blocks as shown below:

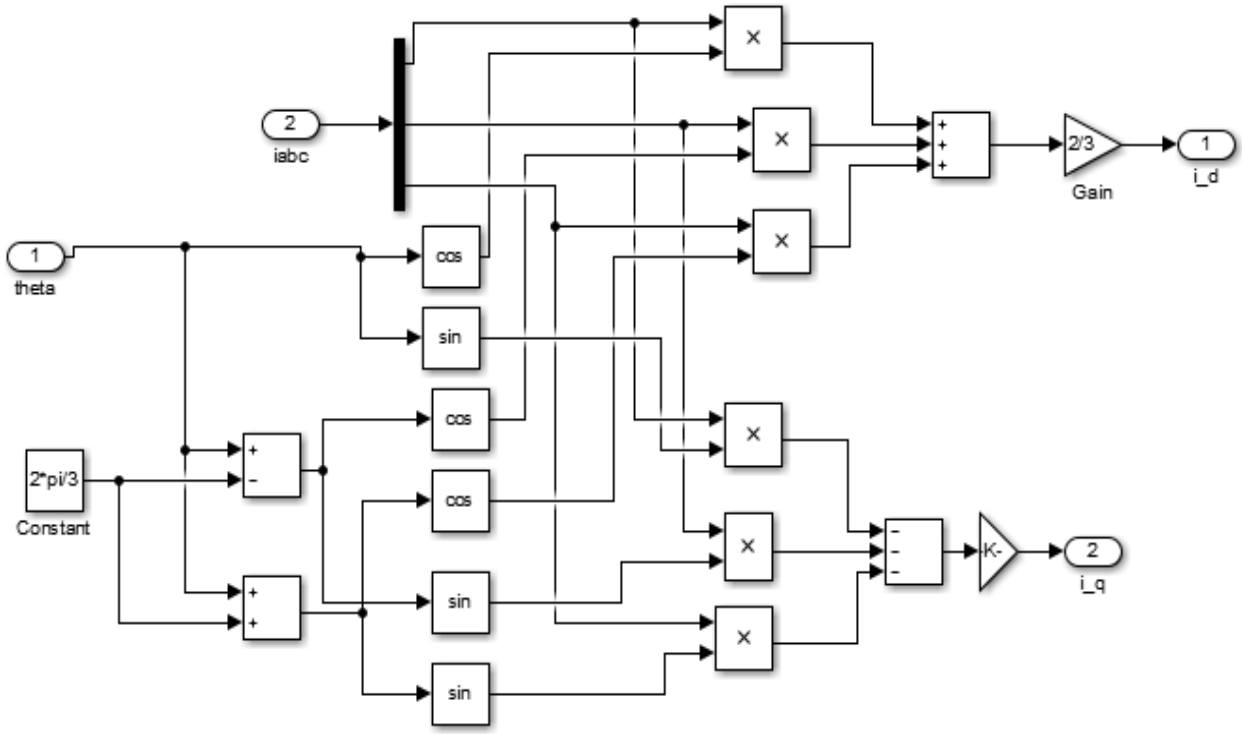


Figure 5.2 simulation block diagram performing inverse Clarke and Park transformation

In the final step of SVPWM simulation model, to generate the PWM pulse, by comparing the vector switching points wave with isosceles triangular wave is produced by repeating sequence, for each IGBT device. PWM1, PWM3 and PWM5, carry on NOT, to get PWM2, PWM4, PWM6.The Simulink model and waveform vector switching points shown in fig 5.3 and 5.4.

.

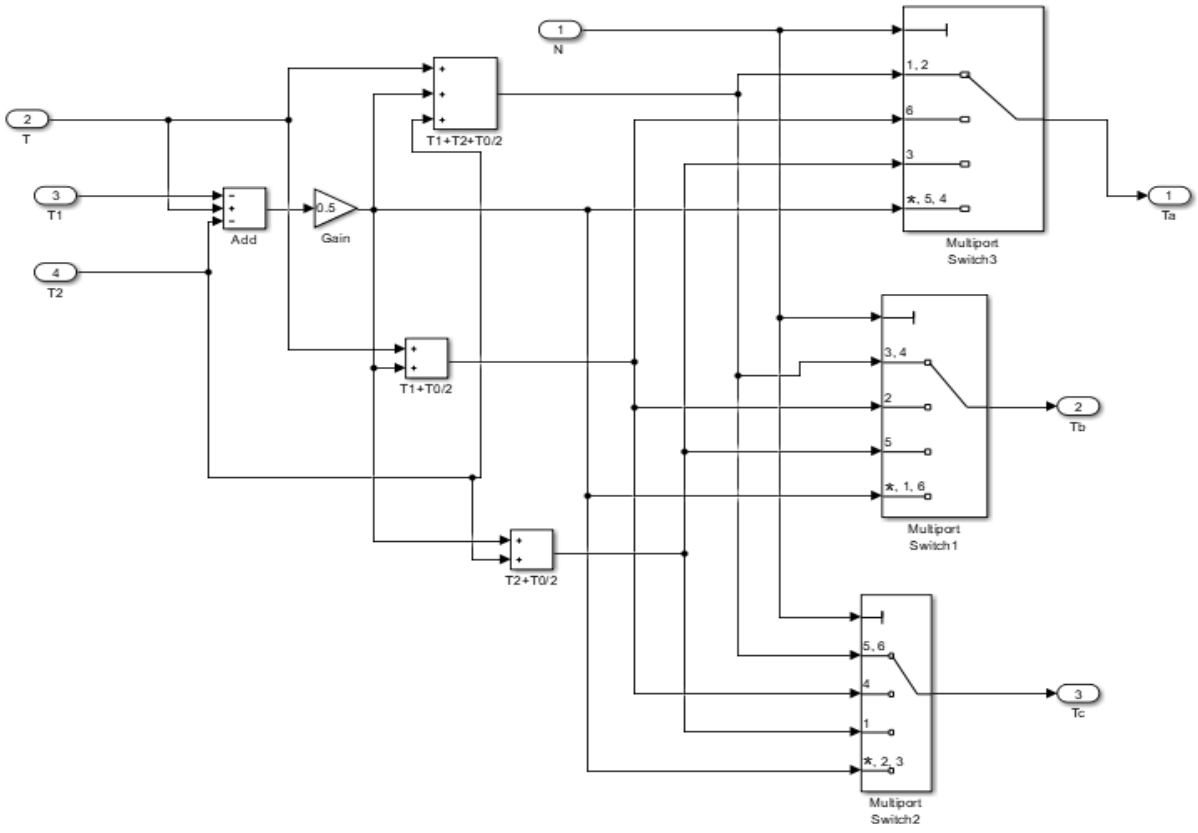


Figure 5.3: Simulink model vector switching points

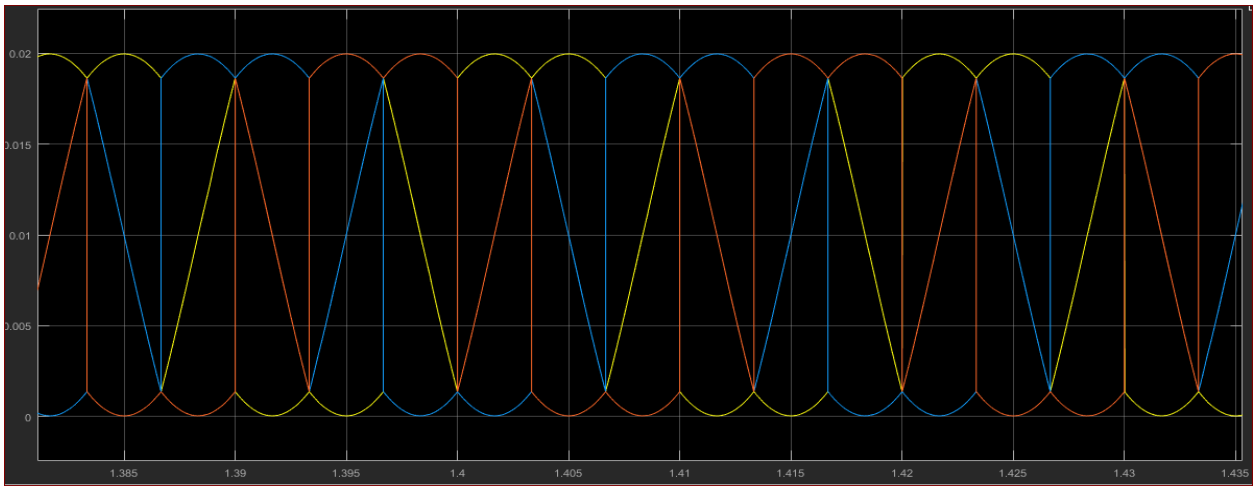


Figure 5.4: The waveform results of vector switching points

In Figure 5.5 it is clear that stator current based MRAS speed estimator is dependent on IPMSM parameters.

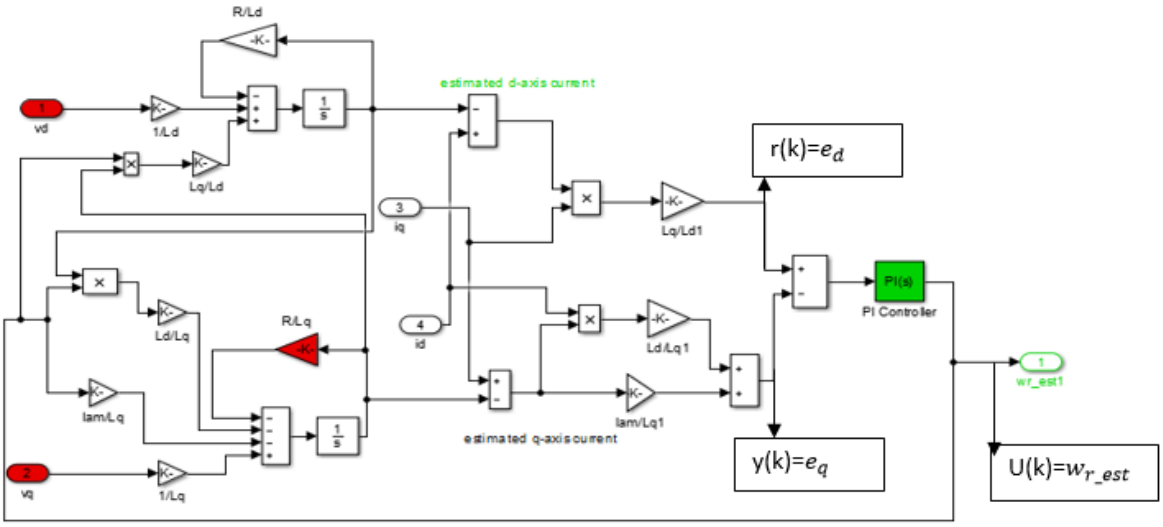


Figure 5. 5: Complete Matlab/Simulink model of MRAS speed estimator.

5.3 Simulation Results

The overall MATLAB/Simulink model of the PMSM drive allows simulating the behavior of the machine using sensor less speed feedback and adaptive neural observer ,for PMSM speed control algorithm ,under different operating modes.

Case A. Variable Speed and Load torque condition

In order to test response,the step response of the proposed controllers,the reference speed is set to 800 rpm and decrease into 300 r/min at 1sec and its effectiveness is compared with the MRAS and pi- neural adaptive controller. The system is loaded with a load of 2 N·m at start the operation, and it is reduced to 1 N.m at 0.5 s up to 1.5sec then increase into 2N.m.

The reference speed change from 800 r/min to 300 r/min is shown in figure 5.6.The result shows that the pi-neural and without pi-neural estimated speed takes 0.09 second to follow the reference speed with good accuracy steady state response.When Speed decrease at time of 1 sec minimized overshoot and remains steady after 1.35 second in neural adaptive controller.

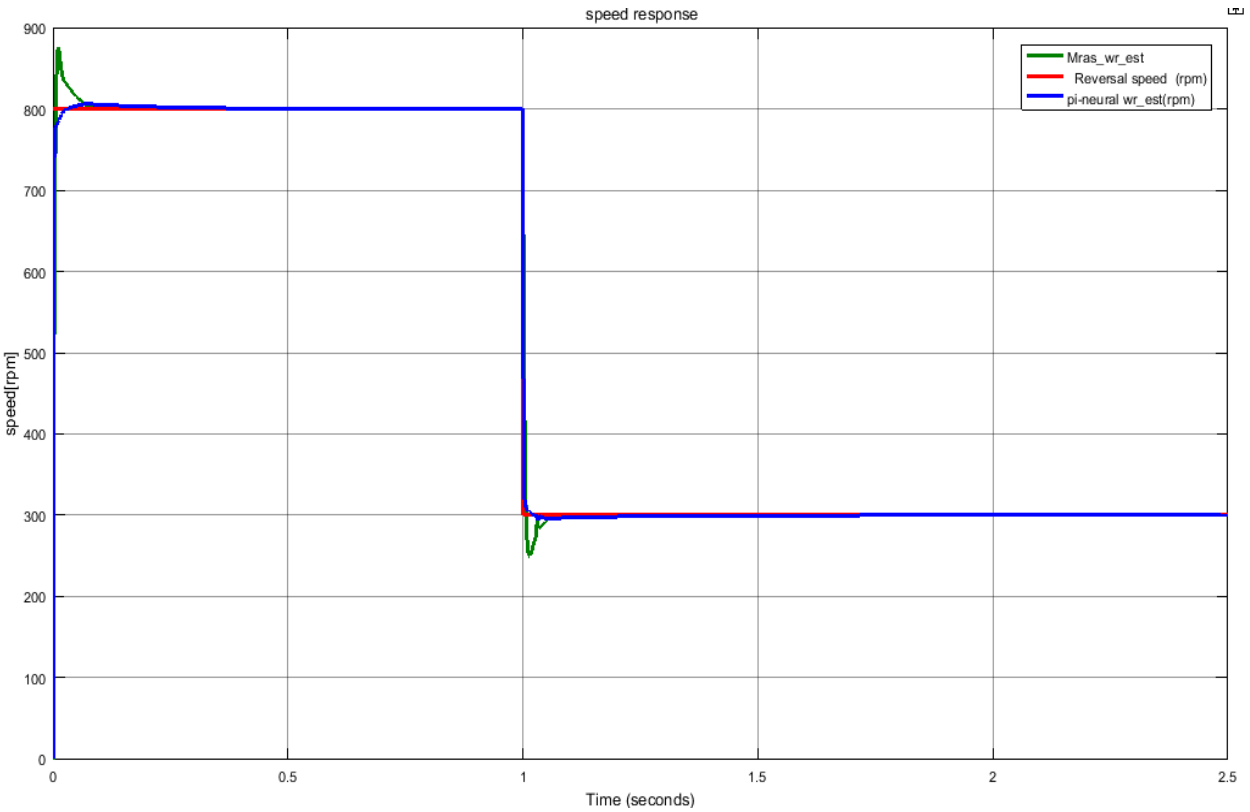


Figure.5.6 Speed response of the IPMSM drive system for variable speed and load torque

❖ Three phase currents

At time starting the reference speed of 800 rpm is given the three-phase currents generated are sinusoidal with minimum distortions for neural observer in the figure 5.7 and starting with more distortions see on MRAS response in figure 5.8.When constant reference speed and a step change in load torque is given at 0.5sec,the waveform of the three -phase currents shows the change its magnitude with change of the torque in fig.5.7 and the motor continues to run at the given reference speed up to 2.5sec. At high speed, when the load is reduced to 1N.m at 0.5 s, the waveform of the current is stable at 0.65A after a reaction late time of about 0.06 sec. At low speed, when load is increase, the waveform of the current is stable at1.245 A after 1.5 sec

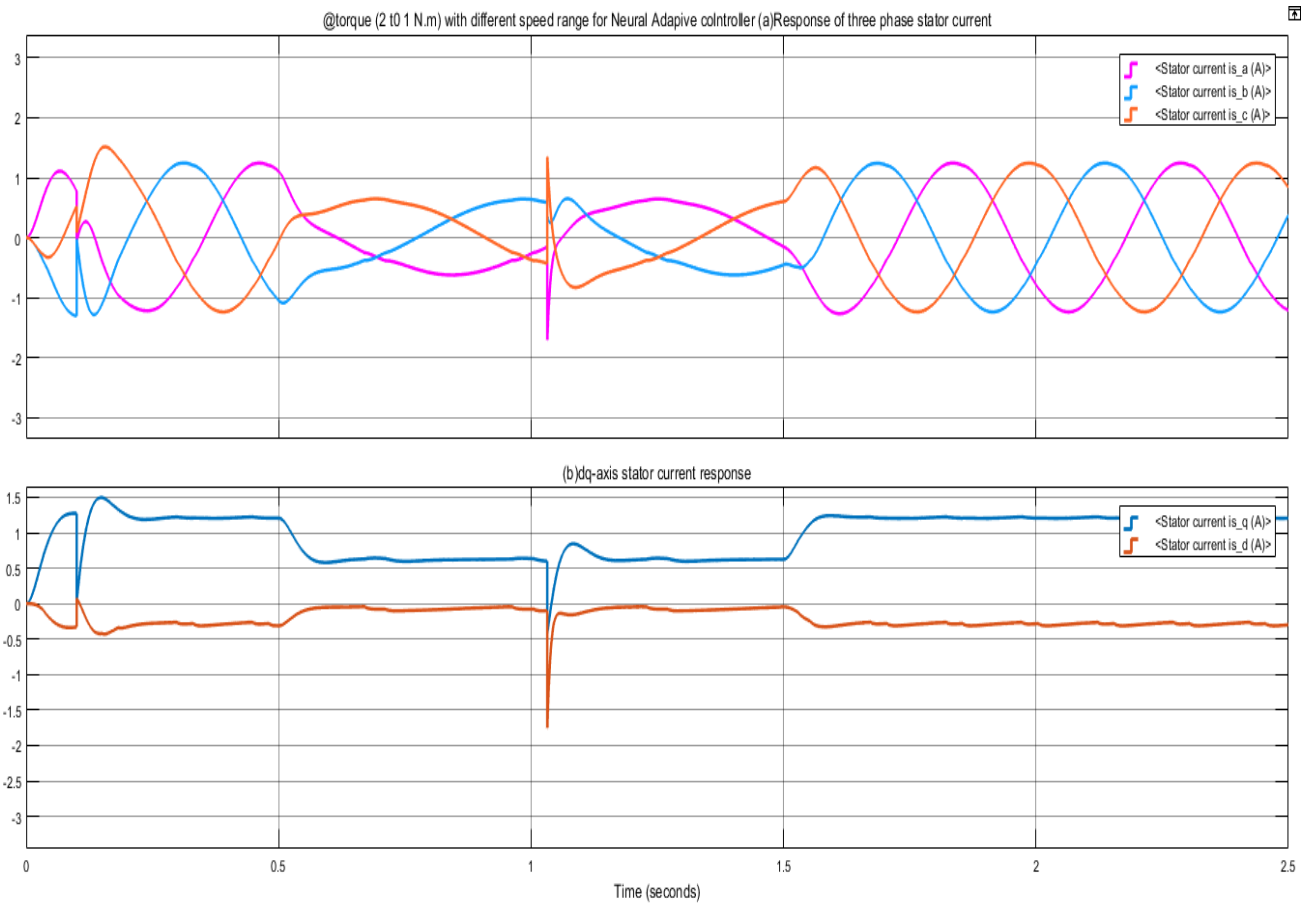


Figure.5.7 Current response of the IPMSM drive using MRAS with PI-Neural Adaptive

❖ d-q-axis stator currents

Figure 5.7 and figure 5.8 the d-q axis of the stator current response at variable load and speed. The d-q axis of the stator current follows the reference and actual torque wave's form.

As constant speed with load change the magnitude wave form of $i_q$current is stable at 0.6A and 1.2A ,the magnitude of i_d current is stable negative 0.1A and 0.3A with given difference time. The $i_q$stator current follows actual torque and $i_q$current follows the reference torque load with some fluctuation at sudden increase and decrease reference speed for MRAS shown in figure 5.8 and good follower for MRAS with pi- neural adaptive .

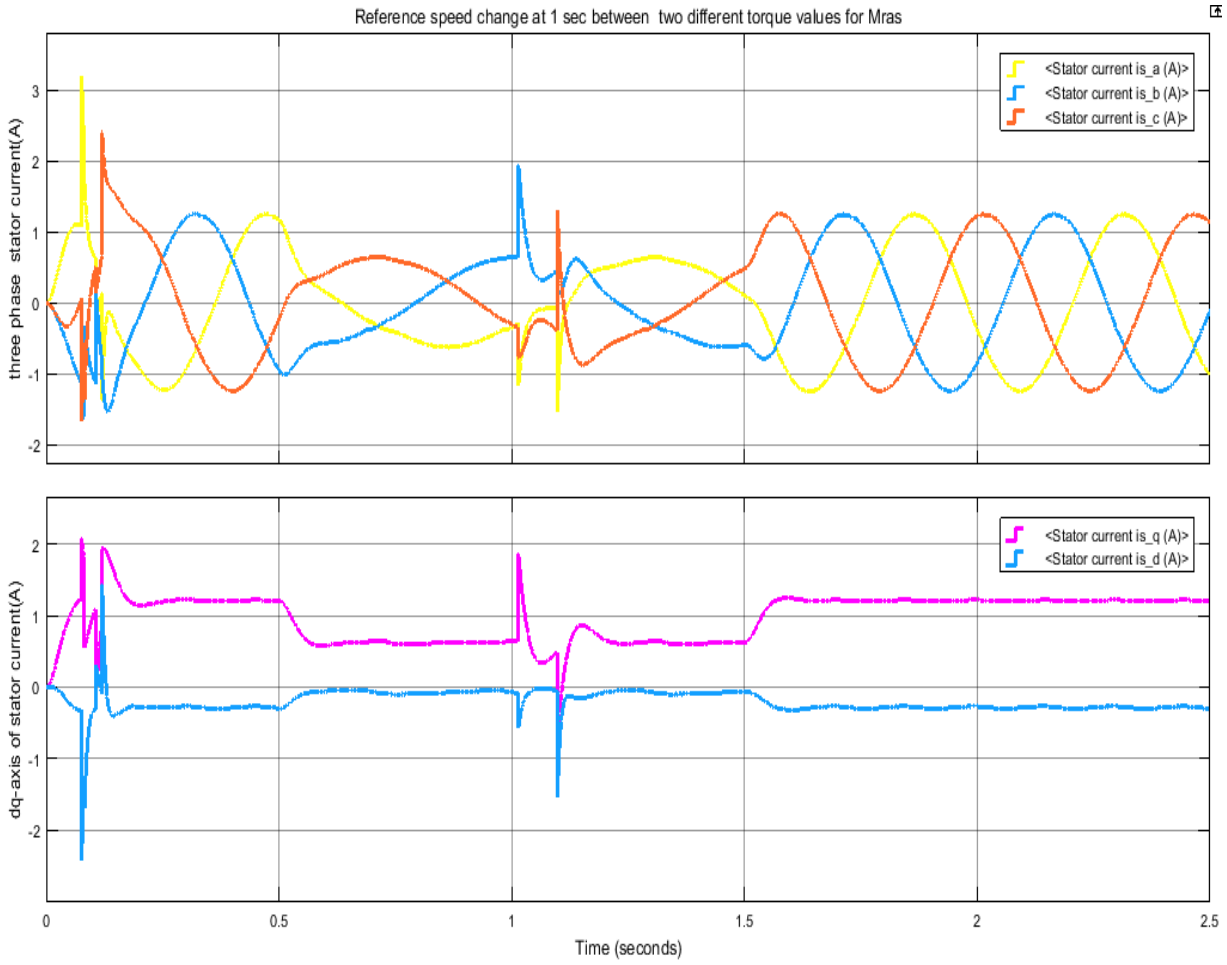


Figure.5.8 Three phase and dq-axis stator current response of the IPMSM drive using MRAS

❖ Load torque response

The electromagnetic torque developed for both MRAS and pi- neural Adaptive models are shown in Figure 5.9.Electromagnetic torque response when reference speed changes from 800 to 300 r/min at t =1 s under the condition that load torque is change from 2 to 1 N.m at 0.5 sec and 2 to1N.m at 1.5 sec, At high speed under MRAS estimator, electromagnetic torque is more ripples and at low speed it have good starting response, stable after 1.5sec and other one is stable at 1.2sec.

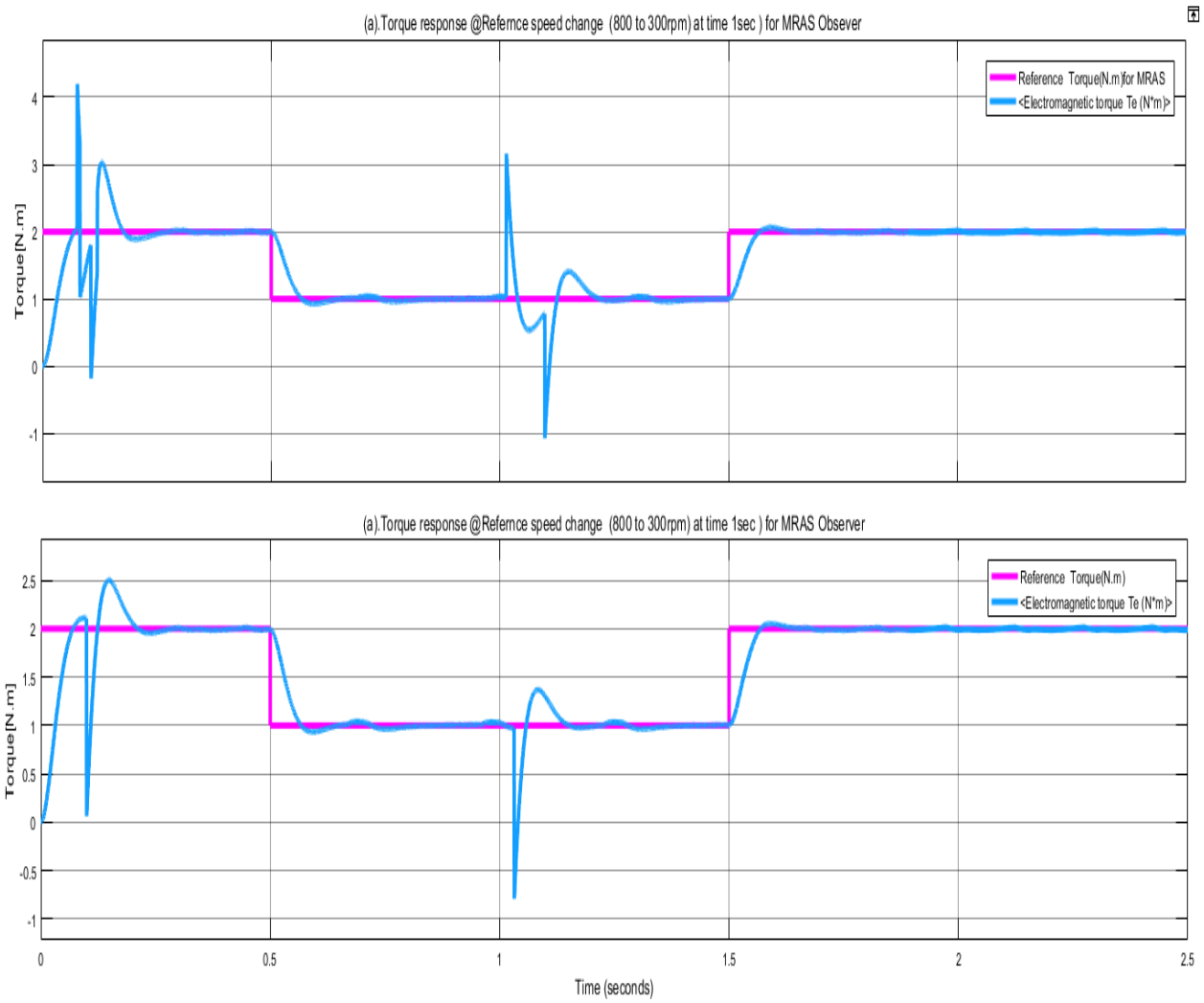


Figure 5.9 .Electromagnetic torque of response when variation speed at 1sec

Case B:-Reversal speed at constant load

Simulation results of the proposed system are shown in Figure 5.6 through Figure.5.10 shows the responses of stator phase, d-q- axis currents, and speed waveforms for both MRAS and pi- neural adaptive modes. Figure 5.11 shows the two method estimation of motor speed for reversal speed in the speed command from 600 rad/s to -600rad/s in the place of constant load. Simulation results are given when the motor is run under the load torque condition at rated of 1 N.m with reversal speed of ±600 rpm.Initially, the drive is operated with the speed command of 600 rad/secup to 1 sec and the speed command is gradually reduced to –600 rad/secattained steady state after 1.2 sec .

Figure 5.11 shows 3-ϕ stator current response of the permanent magnet synchronous motor for reversal speed adaptive neural observer at 1Nm load is applied.When the speed is de-accelerated from 600rad/sec and attained steady speed of -600rad/sec, the direction of current is changing or the three phases are reversed.Also,it has good steady state and transient response.

Figure 5.12 shows the generated electromagnetic torque. When the speed is decelerated, the torque values are negative. At this stage the motor act as a generator. The pi-neural adaptive speed estimation scheme, performance better under steady state with negligible error and the estimated speed closely matches the actual speed .The performance model reference adaptive system after declined reference it have poor operation at given load. The corresponding three phase stator current is a small fluctuation in the waveform shown in figure 5.13.

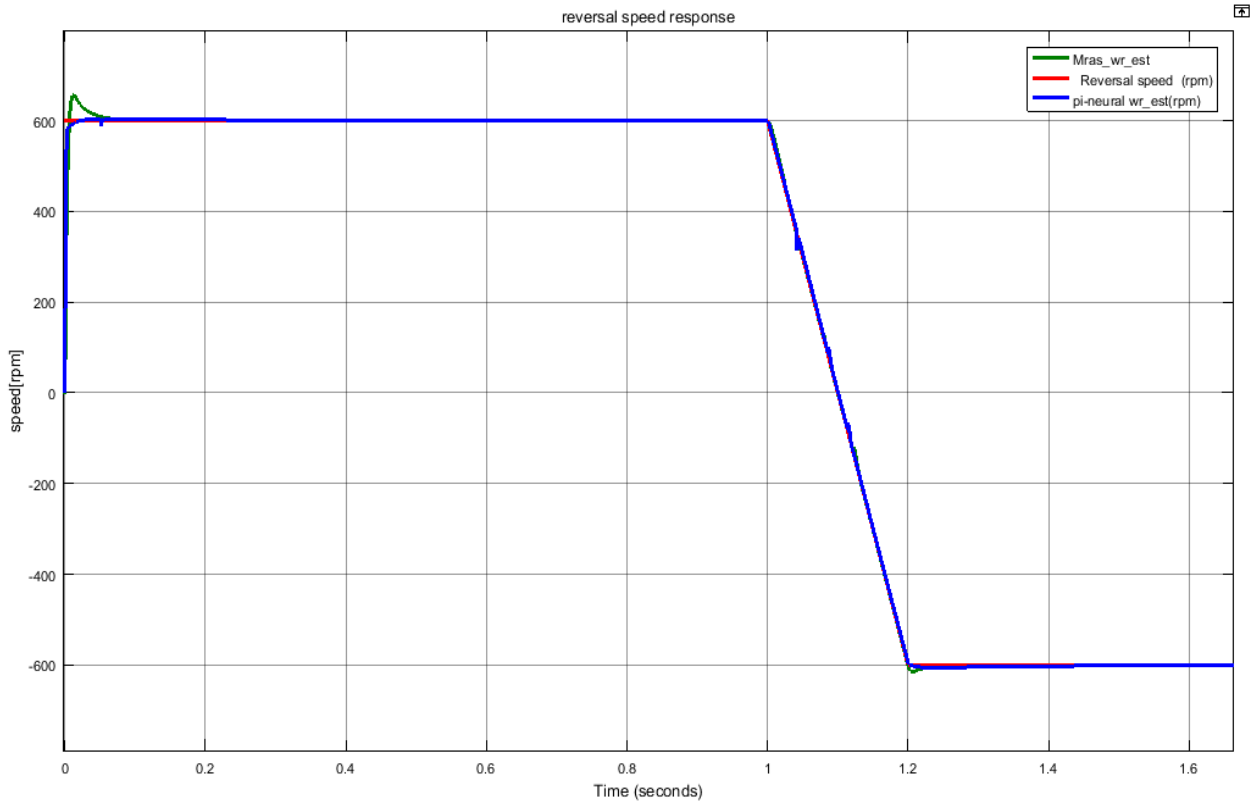


Figure 5.10: Speed response of IPMSM for reversal speed for Neural Adaptive controller

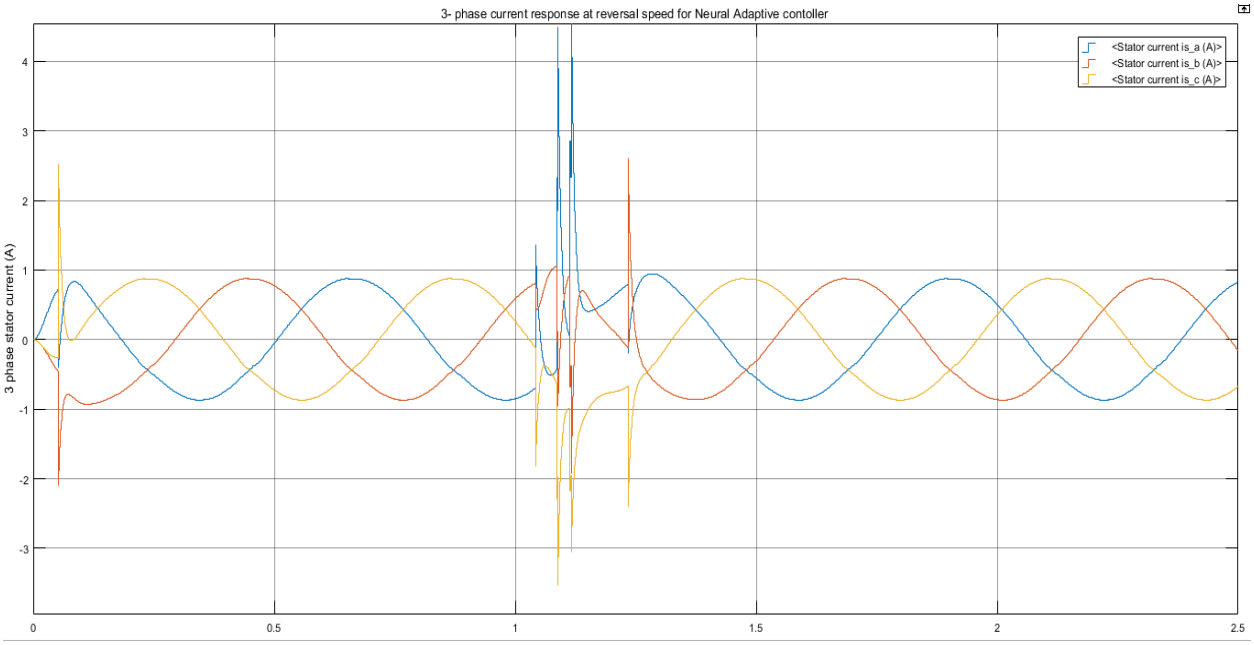


Figure 5.11: 3-ϕ stator current response for reversal speed with neural adaptive

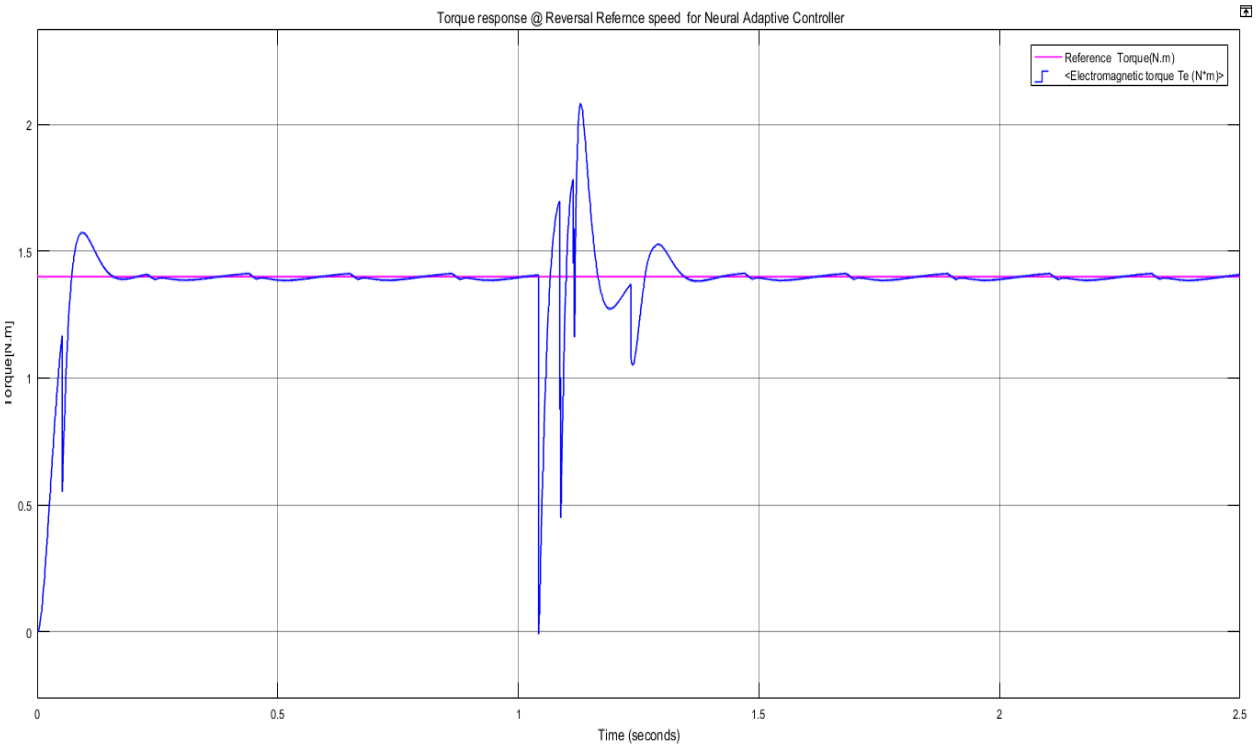


Figure 5.12: Electromagnetic torque of IPMSM when reversal speed under Neural Adaptive

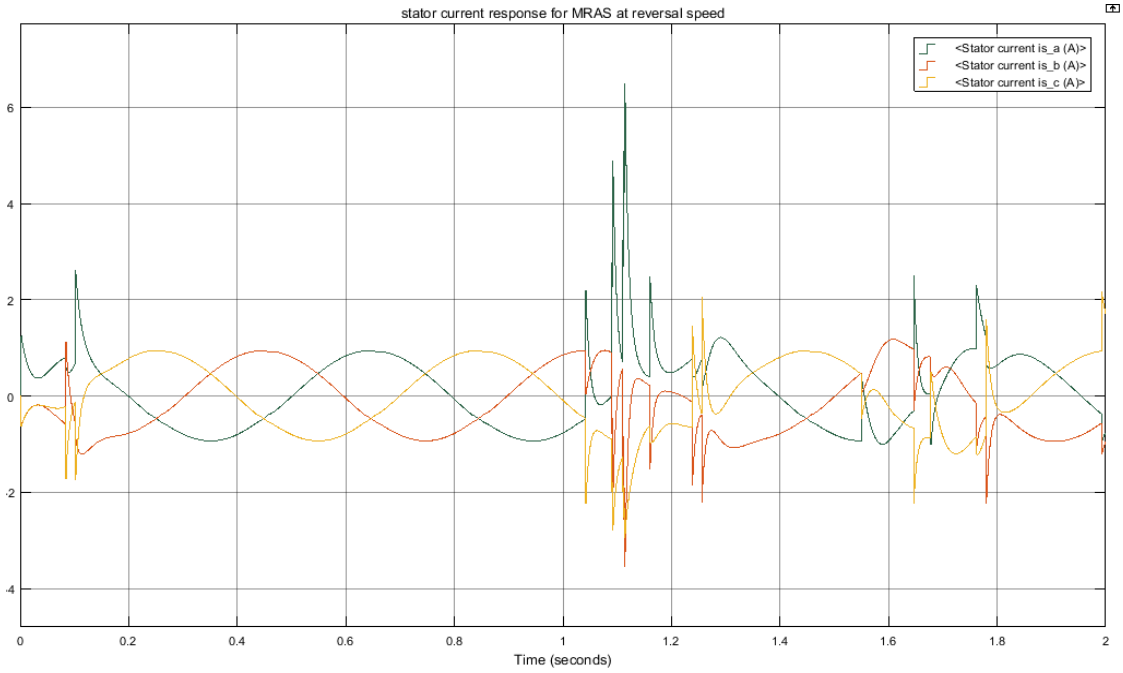


Figure 5.13: 3-ϕ stator current response of the IPMSM for reversal speed with mras

CHAPTER SIX

6. Conclusion and Future works

6.1 Conclusion

In this thesis, MRAS – model is built according to the stator current d – q- axis equations, for FOC of IPMSM drives. This model to estimate the rotor speed ,where output of the reference model is take directly from measurable currents and disturbance signal of stator voltage takes from decoupling controller used for input of the Adjustable model. The adaptive law of speed estimation is constructed according to the Popov stability.

The simulation results show that, the proposed a pi-radial basic function based MRAS method ,to provide better response when compared to conventional pi controller in terms of rise time, settling time, and torque ripple. It can be claimed that the proposed controller is highly successful in speed tracking under variation load conditions and reversal speed references.

6.2 Future works

Recommendations for future works are listed as follows:

- Before training radial basis neural network, to determine the center by k-means or other clustering algorithm for selecting the adequate set of centers.
- Add other neural network with radial basis function like back propagation neutral network or express by s-function.

Reference

[1] P. P. a. R. Krishnan, "Application Characteristics of Permanent Magnet Synchronous and Brushless dc Motors for Servo Drives," *IEEE Transactions on Industry Applications,* vol. 21, September/October 1991..

[2] M. ˇ. -V. H. -M. Franko, "Permanent Magnets Synchronous Motor Control Theory," *Journal of Electrical Engineering,,* vol. 58, 2007.

[3] L. Y. a. Z. Hao, "Sensorless Control of PMSM," in *IEEE Vehicle Power and Propulsion Conference (VPPC)*, Harbin, China, September 3-5, 2008.

[4] B. S. B. P. a. D. S. Singh, "A review of sensor reduction techniques in PMSM," *Proc. Int. J. Power Energy Syst,* vol. 29, pp. 10-18.

[5] Q. H. Y.Y.Li, " Improved chaotic ant swarm algorithm," *Chinese Journal of Scientific Instrument,* vol. 4, no. 30, pp. 733-737., 2009.

[6] Z. H.Q.ZHI, " Comparative Study on BP Network and RBF Network in Function Approximation," *Bulletin of Science and Technology,* vol. 2, no. 21, pp. 193-1966, 2005.

[7] H.Xia, "A fast identification algorithm for box-cox transformation based radial basis function neural network," *IEEE Transactions on Neural Networks,* pp. 1064-1069, 2009.

[8] V. L. U. a. S. K. T, "Artificial Neural Networks based Speed control of Permanent Magnet Synchronous Motor ,," in *International Conference on Soft Computing*.

[9] F. Poltschak and W.Amrhein, ""“Influence of winding layout of PMSM on inductance and its impact on suitability for sensorless vector control," *International Symposium on Power Electronics, ElectricalDrives,* pp. 1002-1007, June 2010.

[10] S.Shinnaka, "A new sensorless vector control method using high-order filters with speed-varying bandwidth for permanent magnet synchronous motors," *”, Electronics and Communications,* vol. 93, pp. 1-14.

[11] L. W. Z. J. e. a. F. Shicai, " A hybrid speed sensorless control strategy for PMSM Based on MRAS and Fuzzy Control,," in *Power Electronics and Motion Control Conference*, 2012.

[12] R. Y. Z. Boze, "The Research on Permanent Magnetic Synchronous Motor Vector Control Based on MRAS," in *Applied Mechanics and Materials*, Vols. 719-720, 2015, pp. 381-387..

[13] K. M. D. S. K. Pradeep, "SensorLess Speed Control of PMSM using SVPWMTechnique Based on MRAS Method for VariousSpeed and Load Variations," in *Lecture Notes in Engineering and Computer Science*, vol. 2217, 2015, pp. 375-380..

[14] W. P. H. B. Z. Hongshuai, "Rotor Position Measurement for High-speed Permanent Magnet Synchronous Motors Based on Fuzzy PI MRAS," *Proceedings of the CSEE,* vol. 32, no. 12, pp. 1889-1896., 2014.

[15] L. Kan, "Research on the Online Multi-parameter Identification of Permanent Magnet Synchrounous Machines," *doctoral dissertation Hunan University,,* 2011.

[16] Z. Q. Z. Z. Z. J. L. Kan, "Comparison of two novel MRAS basedstrategies for identifying parameters inpermanent magnet synchronous motors," *International Journal of Automation andComputing,* vol. 7, no. 4, pp. 516-524., 2010.

[17] F. Z. L. B. Y. Jianguo, "Study on Sensorless Control Model Based on a Model Reference Adaptive System with Parameter Optimization,," in *Applied Mechanics and Materials*, Vols. 121-126, 2011, pp. 4770-4778.

[18] M. A. D. Zaltni, "Synchronous Motor Observability Study and a New Robust MRAS Speed Observer with an Improved Zero-speed Position Estimation Design for Surface PMSM," *WSEAS Transactions on Systems and Control,* vol. 5, no. 7, pp. 567-580, 2010.

[19] J. D.-L. A. G. e. a. M.A. Hamida, "An Adaptive Interconnected Observer for Sensorless Control of PM Synchronous Motors With Online Parameter Identification, I," *IEEE Transactions on Industrial Electronics,* vol. 60, no. 2, pp. 739-748., 2013.

[20] D. fita, "FIELD WEAKENING CONTROL OF PMSM," *ADDIS ABABA UNIVERSITY ,* 2005.

[21] a. F. N. M. S. Merzoug, "Comparison of Field-Oriented Control and Direct Torque Control for PMSM," *World Academy of Science, Engineering and Technology,* 21 2008..

[22] C.-M. Ong, Dynamic Simulation of Electrical Machinery: Using Matlab/Simulink, New Jersey: Prentice Hall, 1998.

[23] G. R. S. a. M. A. R. T. Sebastian, "Design considerations for variable speed permanent magnet motors," in *Proc. International Conf: on Electrical Machines*, 1986.

[24] K. B. Bimal, Modern Power Electronics and AC Drives,, Beijing: Machine Press, 2006.

[25] G. O. Bilal, "Sensorless vector control of a Permanent magnet synchronuous motor with fuzzy logic observer.," *Electrical Engineering,* vol. 88, no. 6, pp. 395-402, 2006.

[26] Pragasen Pillay and Ramu Krishnan, "Application Characteristics of Permanent Magnet Synchronous and Brushless dc Motors for Servo Drives," *IEEE Transactions on Industry Applications,* vol. 21, no. 5, September/October 1991..

[27] K. Khamesra, "PERFORMANCE INVESTIGATION OF INTELLIGENT POWER MODULE BASED PMSM DRIVE," Maharana Pratap University of Agriculture and Technology, Udaipur , 2013.

[28] www.microsemi.com/soc/company/contact/default.aspx, "Field Oriented Control of Permanent Magnet Synchronous Motors User's Guide," *Microsemi.,* 2019.

[29] S.-K. Sul, Control of Electric Machine Drive Systems, 1st, Ed., New Jersey: John Wiley & Sons, Inc.,Hoboken,, 2011.

[30] B. Bose, " A high-performance inverter-fed drive system of an interior permanent Magnet synchronous machine," *IEEE Transactions on Industry Applications,* vol. 24, no. 6, p. 987–997.

[31] G. L. D. Dmitry V. Lukichev, "Features of Tuning Strategy for Field Oriented Control of PMSM Position Drive System with Two-mass Load".

[32] R. Krishnan, Permanent Magnet Synchronous and Brushless DCMotor Drives,, Taylor & Francis Group, 2010.

[33] A. Cimpoeru, Encoderless Vector Control of PMSG for Wind Turbine Applications, Conducted by group PED , 2010.

[34] H. M.P.Kazmierkowski, Automatic Control of Converter-Fed Drive, ,Warszawa , 994.

[35] .Ogata, Modern Control Engineering,, 3 ed., 1997.

[36] M. S. J. W. Umland, "Magnitude and Symmetric Optimum Criterion for the Design of Linear Control Systems," *IEEE Transaction on Industry Applications,,* pp. 489-497, may/june 1990.

[37] G. PED-9, Control of a variable speed variable pitch wind turbine with full scale power converter,, Aalborg University, , December 2007.

[38] M. Liserre, "Innovative control techniques of power converters for industrial automation," Italy, Dec. 2001., Dec. 2001..

[39] S. P. l, "High Speed Field Oriented Control ‘’," *‘ Institute of Energy Technology Aalborg University,* June 2009.

[40] V. R. S. a. S. N. Vukosavic, "Space vector PWM voltage control with optimized switching strategy," *in Proc. IEEE Industry Applications Soc Annu. Meeting ,* pp. 1025-1033., 1992 .

[41] H. C. S. a. G. V. S. W.vanderBroek, " Analysis and realization of PWM based on voltage space vectors," *IEEE Trans. Ind. Applicat. ,* vol. 24, no. 1, p. 142–150 , 1988.

[42] J..Holtz, "Pulse-width-modulation–Asurvey," *IEEE Trans. Ind. Electr.,,* vol. 39, pp. 410-419, Dec.1992..

[43] K. V. D. a. K. Pradeep. H. Yakkundi, "Design and Simulation of Space Vector PWM for Three-Phase Induction Motor," *IOSR-JEEE,* vol. 9, no. 3, pp. 01-08, MAY-JUN 2014.

[44] A. S. a. A. M. K. A. Maamoun, "Space-vector PWMinverter feeding a small induction motor," ,," in *inProc. IEEE Int. Conf. on Mechatronics*, Komamoto, Japan,, May 2007.

[45] H. S. H. a. S. Van derBroeck, "Analysis and realization of pulse width modulator based on voltage space vectors," *IEEETrans. Ind. Elec.,* vol. 24, no. 1, pp. 142-150., 1988.

[46] J. J. Y. G. a. J. Z. Z. Wang, " SVPWM techniques and applications in HTSPMSM machines control," *Journal of Electronic Science and Technology of China,* vol. 6, no. 2, pp. 191-197, June 2008 .

[47] A. Y. a. Y. A. A. Saadoun, "Modeling and simulation of DSP controlled SVPWM three - phase VSI," *Journal of Applied Sciences,* vol. 7, no. 7, pp. 989-994, 2007.

[48] M. A. S. M. P. D. G. a. A. M. N. L. P. F. Seixas, "A space-vector PWM method for three-level voltage source inverters," *in Proc. IEEE APEC,* vol. 1, p. 549–555, 200.

[49] H. H. a. Z. L. W. Yao, "Comparisons of space-vector modulation and carrier-based modulation of multilevel inverter," *IEEE Transpower Electron,* vol. 23, no. 1, pp. 45-51, jan.2008.

[50] N. a. D. Boroyevich, "A fast space-vector modulation algorithm for multilevel three-phase converters," *IEEE Trans. Ind.Appl,* vol. 57, no. 2, p. 637–641, Mar./Apr. 2001.

[51] S. K. a. K. G. N.Tekwani, "A dual five-level induction motor drive with common-mode voltage elimination and DC-link capacitor voltage balancing using only the switching-state redundancy," *IEEE Trans. Ind. Electron.,* vol. 54, no. 5, p. 2600–2608, Oct. 2007..

[52] K. K. M. a. L. U. V. T. Somasekhar K. Gopakumar M. R. Baiju, "A multilevel inverter system for an induction motor with openend windings," *IEEE Trans. Ind. Electron,* vol. 52, no. 3, p. 824–836, Jun. 2005.12.

[53] D. K. G. a. H. S. Renu Sharma, "Analysis of Space Vector PWM for Three Phase Inverter and Comparison with SPWM," *International Journal of Advanced Research in IEEE ,* vol. 5, no. 1, January 2016.

[54] F. R. Y. a. R. A. Mahmood, "Design New control System for Brushless DC motor Using SVPWM," *International Journal of Applied Engineering Research ISSN ,* vol. 13, no. 1, pp. 582-589, 2018.

[55] N. DO., " Space Vector Modulation," in *The 27th Annual Conference of the IEEE Industrial Electronics Society* , 2,Dec,2001.

[56] M. Balaji, speed control of bldc motordrive using svpwm inverter, ,Rourkela, Odisha: National Institute of Technology Rourkela, 2015.

[57] P. A. M. ,. P. J. a. S. S. K. V. Kumar, "Simulation And Comparison Of SPWM and SVPWM Control For Three Phase Inverter," *ARPN Journal Of Engineering Andapplied Sciences,* vol. 5, no. 7, pp. 61-74, 2010.

[58] A. L. S. a. A. A. Obed, " Speed Control of Brushless DC Motor based on Fractional Order PIDController," *International Journal of Computer Applications ,* vol. 95, no. 4, p. 0975 – 8887, 2014.

[59] N. a. PratibhaTiwari, "Analysis of Sensor less Permanent Synchronous Motor Using PI and Adaptive Control Scheme," *IJSETR,* vol. 4, April 2015.

[60] Y. L. a. H. Zhu, "Sensorless control of permanent magnet synchronous motor - a survey," in *in Vehicle Power and Propulsion Conference*, 2008.

[61] S. G. a. S. S. H. Mugdha B, "MRA Technique for Sensor less Speed Control of Induction Motor Using MATLAB \SIMULINK," *IJETCSE,* vol. 14, no. 2, APRIL ,2015.

[62] X. X. a. Y. Li, " A sensor less control based on MRAS method in interior permanent-magnet machine drive," in *in Proc. Int. Conf. Power Electron. Drive Sys*, 2005.

[63] Y. L. a. Y. Li, " Sensorless control of PM synchronous motors based on MRAS method and initial position estimation," in *inProc. Int. Conf.Elect. Mach. Syst*, 2003.

[64] C.Schauder, " Adaptive speed identification for vector control of inductionmotors without rotational transducers," *IEEE Transactions on Industrial Applications ,* vol. 28, no. 2, p. 1054–1061, 1992.

[65] H. M. D. D. Akrad. A, "Design of a Fault-Tolerant Controller Based on Observers for a PMSM Drive.," *Industrial Electronics, IEEETransactions on,* vol. 58, no. 4, pp. 1416-1427, April 2011. .

[66] F. Z. P. a. T.Fukao, "Robust speed identification for speed sensorless vector control of induction motors," *IEEE Trans. Industry Application,* vol. 30, no. 5, pp. 1234–1240,, Sep./Oct. 1994..

[67] V. P, Sensor less Vector and Direct Torque Control, New York: Oxford Univ. Press, 1998.

[68] R. T. a. H. B. S.Meziane, "Speed Sensorless Direct Torque Control and Stator Resistance Estimator of Induction Motor Based MRAS Method," *International Journal of Applied Engineering Research (IJAER),* vol. 3, no. 6, pp. 733-747 , 2008.

[69] Y.P. Landau, "A hyperstability criterion for model reference adaptive control system," *IEEE Transactions on Automatic Control 14 (5) (1969) 552–555,* vol. 14, no. 5, p. 552–555, 1969.

[70] Z. H. a. L. X. W. Lipeng, "Robust sensorless of ADRC controlled PMSM based on MRAS with stator resistance identification," in *in Control Conference(CCC)*, Chinese, 2011.

[71] W. X. Z. F. G. X. a. Z. P. Z. Xingming, "Wide-Speed-Range Sensorless Control of Interior PMSM Based on MRAS," in *in International Conference on*.

[72] P. B. P. Brandstetter, "Applications of Artificial Neural Networks in Control of DC Drive," in *CISIS'12 - ICEUTE'12 - SOCO'12* , 2013.

[73] L. W. v. L. M Zhang M., "Adaptive Control Strategy Based on RBF Neural Network Identification," in *Neural Networks and Brain*, 2005.

[74] C.-F. Z. Y.-Y. J. Wang.J.-J., "Self-Adaptive RBF Neural Network PID Control in Exhaust Temperature of Micro Gas Turbine.," *International Conference on Machine Learning and Cybernetics,* vol. 4, pp. 12-15, 2008.

[75] S. A. K. a. M. Bartere, "A Simple Neural Network Approach to Implementing Technical Institute Student Feedback System in Spreadsheet," *International journal of advanced research in computer science ,* vol. 1, no. 4, Nov-Dec 2010.

[76] S. M.-I. I. I. a. F. M. J. Zibert, "Bilingual Speech Recognition of Slovenian and Croatian Weather Forecasts," in *in Proceedings of the EURASIP* , croatia, July,2000.

[77] S. Haykin, Neural Network, A Comprehensive Foundation., New York: Macmillan College Publishing Company, 1994.

[78] J. E. Q. Mendez', "Stator Winding Fault Detection for a PmsmUsing Fuzzy Logic Classifier and NeuralNetwork Model Identification''," Florida State University Libraries, March 27,2008.

[79] G. Cybenko, " Approximation by superposition of a Sigmoidal Function,," *Mathematics of Control, Signal and Systems,* vol. 2, no. 4, pp. 303-314, 1989.

[80] C. Leondes, Intelligent Systems: Fuzzy Systems, Neural Networks and Expert System., Florida: CRC., 2003.

[81] L. W. Z. Y. e. a. Yi Yanping, "Study on the Surrogate Model of Groundwater Numerical Simulation Model Based on Radial Basis Function Neural Network," *Research of Soil and Water Conservation,* vol. 19, no. 4, pp. 265-269., 2012.

[82] Z. Defeng., Neural Network Application Design of Matlab, .Beijing: China Machine Press, 2009.

[83] S. V. T. Y. C. S. E lanayar, "Radial Basis Unction Neural Network for Approximation and Estimation of Nonlinear Stochastic Dynamic Systems," *IEEE Transaction on Neural Network,* vol. 5, pp. 584-603., 1994.

[84] O. J.L, Introduction to Radial Basis Function Networks., University of Edinburg, 1996.

[85] J.K.Liu, "MATLAB Simulation with Advanced PIDControl," Publishing House of Electronics Industry, Beijing, 2011.

[86] C. X. G. Kong Xiangsong, "PID Controller Design Based on Radial Basis Function Neural Networks for the Steam Generator Level Control," *CYBERNETICS AND INFORMATION TECHNOLOGIES,* vol. 16, no. 5.

[87] Qiaofu, "Study on Adaptive PID Control Algorithm Based on RBF Neural Network," *Journal of applied science and engineering innovation,* vol. 1, no. 2, 2014.

[88] K. a. D. S. Hunt, " Neural Networks for Nonlinear Internal Model Control," *IEE Proceeding,* vol. 138, no. 5, pp. 431-438., 1991.

[89] S. P. a. G. Sakthivel, "Auto Tuning of PI Controller Using Radial Basis Function Neural Network for a Spherical Tank Process," *World Applied Sciences Journal ,* vol. 34, no. 11, pp. 1570-1576, 2016.

[90] S. K. G. Cui X., "Direct Control and Coordination Using Neural Networks," *IEEE Transactions on Systems, Man and Cybernetics,* vol. 23, no. 3, pp. 686-697, 1992.

[91] N. a. M. Z. Vuković, " A growing and pruning sequential learning algorithm of hyper basis function neural network for function approximation," *IEEE,* vol. 46, pp. 210-226, 2013.

[92] G.-B. P. S. a. N. S. Huang, "A generalized growing and pruning RBF (GGAP-RBF) neural network for function approximation," *IEEE Transactions on,* vol. 16, no. 1, pp. 57-67, 2005.

[93] N. R. a. B. G. Kalyan Kumar Halder, "PositionSensorless Control for an Interior PMSM SVM Drive with ANN Based Stator Flux Estimator," *International Journal of Computer and Electrical Engineering,,* vol. 2, no. 3, June, 2010.

Appendix

I. Appendix: A

Table A1: The IPMSM characteristic parameter [93]

Rating: 3-phase, 1 hp, 300 V, 3A, 50 Hz,

Name of the Parameter	Rating of the Parameter
Stator resistance (R_s)	5.8 Ω
Inductance d-axis (L_d)	0.0448 H
Inductance q-axis (L_q)	0.1024 H
Magnetic flux constant	0.533 Wb
Moment of inertia (J)	0.0087 Kg·m
Friction coefficient(B)	0.0008 Nm/rad/sec
Poles (P)	4
Rated torque (T_L)	6 N-m
Speed (N)	1500 r.p.m

II. Appendix: B

```
% PI control based on RBF neural network
clear all;
close all;
% smodel = 'gashukpi'; %neural network based speed estimator of
% %pmsm MATHLAB Simulink file name
% open(smodel);
xite=0.27;alfa=0.06;belte=0.03;
x=[0,0,0]';
ci=5*rand(3,6);bi=450*ones(6,1);w=80*ones(6,1); h=[0,0,0,0,0,0]';
ci_1=ci;ci_3=ci_1;ci_2=ci_1;  bi_1=bi;bi_2=bi_1;bi_3=bi_2;
w_1=w;w_2=w_1;w_3=w_1;
u_1=0;xc=[0,0]';E=0;
error_1=0;error=0;
kp0=0.00026;        ki0=0.032;     kp_1=kp0;ki_1=ki0;
xitekp=0.20;xiteki=0.2;
 datavolt= xlsread('C:Adaptive mechanism.xlsx','eqeq1wrest','A1:C150');
datacurrent= xlsread('C:Adaptive mechanism.xlsx','ed=r','A1:A150');
input_train= datavolt(:,1:3);
output_train= datacurrent(:,1);
inputtrain= datavolt(:,1);
```

```
inputs= input_train;
r=output_train';
y=inputtrain;
samples = size(y,1);
ts=0.0001;
for k=1:samples
time(k)=k*ts;
for j=1:1:6
h(j)=exp(-norm(inputs(k,:)'-ci(:,j))^2/(2*bi(j)*bi(j)));
end
ym(k)=w'*h;
d_w=0*w;              d_bi=0*bi;      d_ci=0*ci;
E(k)= y(k) - ym(k);
for j=1:1:6
d_w(j)=xite*E(k)*h(j);
d_bi(j)=xite*E(k)*w(j)*h(j)*(bi(j)^-3)*norm(inputs(k,:)'-ci(:,j))^2;
for i=1:1:3
d_ci(i,j)=xite*E(k)*w(j)*h(j)*(inputs(k,i)-ci(i,j))*(bi(j)^-2);
end
end
w=w_1+d_w+alfa*(w_1-w_2)+belte*(w_2-w_3);
bi=bi_1+ d_bi+alfa*(bi_1-bi_2)+belte*(bi_2-bi_3);
ci=ci_1+d_ci+alfa*(ci_1-ci_2)+belte*(ci_2-ci_3);

%%%%%%%%%%%%%%%%%%Jacobian%%%%%%%%%%%%%%%%%%%%
yu=0;
for  j=1:1:6
yu=yu+w(j)*h(j)*(-x(1)+ci(1,j))/bi(j)^2; end
dyu(k)=yu;
error(k)=r(k)-y(k);

if abs(error(k))<=0.0085
        xc(2)=xc(2)+error(k)*ts;
    else xc(2)=0; end

kp(k)=kp_1+xitekp*error(k)*dyu(k)*xc(1);
ki(k)=ki_1+xiteki*error(k)*dyu(k)*xc(2);
if kp(k)<0  kp(k)=0;end
if ki(k)<0  ki(k)=0;  end
du(k)=kp(k)*xc(1)+ki(k)*xc(2);
u(k)=u_1+du(k);
%Return of parameters
x(1)=du(k);       u_1=u(k);
ci_3=ci_2;ci_2=ci_1;ci_1=ci;
bi_3=bi_2;bi_2=bi_1;bi_1=bi;
w_3=w_2;w_2=w_1;w_1=w;
xc(1)=error(k)-error_1;                %Calculating P
xc(2)=error(k);                        %Calculating I
error_1=error(k);
kp_1=kp(k);     ki_1=ki(k);
end
```