Addis Ababa University

Addis Ababa Institute of Technology

School of Electrical and Computer Engineering

MRAS Based Sensorless Speed Control of Permanent Magnet Synchronous Motor Using Fuzzy Logic-PI Controller

A Thesis Submitted to the School of Graduate Studies of Addis Ababa University in Partial Fulfillment of the Requirements for the Degree of Master of Science in Control Engineering

By

Yimam Yimer

Advisor

Dr. Mengesha Mamo

November 20, 2023

Addis Ababa, Ethiopia

Addis Ababa University

Addis Ababa Institute of Technology

School of Electrical and Computer Engineering

This is to certify that the thesis prepared by Yimam Yimer, entitled "MRAS Based Sensorless Speed Control of Permanent Magnet Synchronous Motor Using Fuzzy Logic-PI Controller" submitted in partial fulfillment of the requirements for the degree of Master of Sciences in Control Engineering complies with the regulations of the university and meets the accepted standards with respect to originality and quality.

Approved by Board of Examiners

Name	Signature	Date
Dr. Bisrat Derebssa		
(School dean)	_____	_____
Dr. Lebsework Negash		
(Chairman)	_____	_____
Dr. Mengesha Mamo		
(Advisor)	_____	_____
Dr. Lebsework Negash		
Internal Examiner	_____	_____
Mr. Mesfin Tilahun		
External Examiner	_____	_____

Declaration

I, the undersigned, certify that this research work titled "MRAS Based Sensorless Speed Control of Permanent Magnet Synchronous Motor Using Fuzzy Logic-PI Controller" is my original work. The work has not been presented for the fulfillment of any degree in this or any other university, and all sources and materials used in the thesis from other sources are properly acknowledged.

Name of Student	**Signature**	**Submission Date**
Yimam Yimer	_____	November 20, 2023.

This thesis has been submitted for evaluation with my approval as a university advisor.

Name of Advisor	**Signature**
Dr. Mengesha Mamo	_____

Acknowledgments

I would like to express my sincere gratitude to my thesis advisor, Dr. Mengesha Mamo. His patience, motivation, constant support, and guidance were invaluable throughout my research. His guidance offered unwavering assistance from the beginning to the end of the research and thesis writing process.

My family members deserve heartfelt thanks for their unwavering encouragement, understanding, and support during my academic journey. Their belief in me has been a continuous source of motivation.

Finally, I extend my appreciation to all my friends and colleagues who provided assistance, shared insights, and offered encouragement during this research endeavor. This work wouldn't have been possible without the collective support and inspiration of everyone who contributed in various ways. Thank you all for being part of this journey.

Yimam Yimer

Abstract

The accurate control of Permanent Magnet Synchronous Motors (PMSMs) without the need for mechanical speed sensors is a significant challenge in various industrial applications. This thesis proposes a Model Reference Adaptive System (MRAS) based sensorless speed control technique for PMSMs using a fuzzy logic-PI controller. The MRAS estimator utilizes a reference and adjustable models to estimate the (rotor) speed of the motor. The fuzzy logic-PI controller combines the advantages of fuzzy logic control and proportional integral control to improve control performance and speed regulation. The study investigates the performance of Proportional Integral (PI) and Fuzzy Logic-Proportional Integral (FL-PI) controllers under various conditions, including load and no-load scenarios, parameter variations and the influence of disturbances like sudden load torque changes. The simulation results reveal that the fuzzy logic-PI controller reduced overshoot to 2.6% from 5% and eliminated steady state error under no-load conditions. It exhibited a 3% overshoot during sudden load changes, the PI controller's 3.9%. the Fl-PI controller consistently outperforms the PI controller in terms of overshoot, settling time and steady state error across different operating conditions. Moreover, the Fl-PI controller demonstrates robustness to parameter variations and effectively mitigates the impact of disturbances on motor speed and current. These findings highlight the superior performance and robustness of the Fl-PI controller, making it a promising choice for speed control in PMSM applications.

Keywords Permanent Magnet Synchronous Motor (PMSM), Model Reference Adaptive System (MRAS), Sensorless Speed, Fuzzy-PI Controller(Fl-PI).

Contents

List of Figures

List of Tables

List of Acronyms

AC	Alternating Current
EKF	Extended Kalman Filter
Fl-PI	Fuzzy PI Controller
FLC	Fuzzy Logic Controller
FOC	Field Oriented Control
Id	Direct Axis Current
Iq	Quadrature Axis Current
MRAS	Model Reference Adaptive System
PI	Proportional Integral
PSO	Particle Swarm Optimization
SMC	Sliding Mode Control
SMO	Sliding Mode Observer
SMPMSM	Surface Permanent Magnet Synchronous Motor
SVPWM	Space Vector Pulse Width Modulation

Chapter 1

Introduction

1.1 Background of the Study

A synchronous motor is a type of electric motor known for its unique operational characteristics. It operates at a constant speed that is precisely synchronized with the frequency of the alternating current power supply to which it is connected. Unlike induction motors, synchronous motors do not experience slip and maintain a constant speed regardless of the load [1]. The synchronous speed (ω_s) is (120 x f)/P, where f is the frequency and P is poles in the motor.

PMSM is a type of synchronous motor that uses permanent magnets in the rotor to generate a constant magnetic field. This magnetic field interacts with magnetic field produced by the stator windings, resulting in synchronous operation. The rotor of a motor contains permanent magnets that generate a constant magnetic field without the need for external excitation. This design eliminates the need for slip rings or brushes typically found in other types of synchronous motors, simplifying the motor's construction and reducing maintenance requirements. The motor operate at a fixed speed ratio with the frequency of the stator current. The rotor rotates at the same speed as the rotating magnetic field produced by the stator, resulting in synchronous operation. PMSMs are widely used in various applications due to their high efficiency, controllability and compact design [2] [3]. In speed dependent applications, sensors measure motor speed and provide feedback to the controller. However, sensors increase space and the risk of system failure. To reduce costs and maintenance, sensorless control uses mathematical estimators to determine rotor speed.
Dealing with permanent magnet synchronous motors in various applications is motivated by

several reasons, making them a popular choice for numerous industries and systems:

High Efficiency: PMSMs are known for their high efficiency, which is significantly higher than that of induction motors. The absence of rotor losses and the use of permanent magnets contribute to their improved efficiency, leading to energy savings and reduced costs.

Precise Speed Control: Permanent magnet synchronous motors offer stable speed control. Through advanced control techniques like the field oriented control, the motor's speed can be regulated, making them suitable for applications that require precise motion control.

High Power Density: PMSMs have high power to weight ratio and power to volume ratio, making them compact and suitable for applications space and weight are critical factors.

Sensorless Control: Permanent magnet synchronous motors can be operated without need for physical speed. Sensorless control techniques estimate the motor's speed and position using mathematical models and current, voltage measurements, reducing complexity and cost.

Application Diversity: PMSMs find applications in various industries, including automotive, industrial automation, robotics, renewable energy systems. Their versatility and efficiency make them suitable for a wide range of modern technologies.

High Torque Density: PMSMs offer a high torque to inertia ratio, providing fast acceleration and deceleration capabilities. This characteristic is essential in applications requiring quick and precise motion control.

Reduced Maintenance: The absence of brushes in motor and fewer mechanical parts result in reduced maintenance requirements, leading to increased reliability of the motor.

Environmental Benefits: Permanent magnet synchronous motors are environmentally friendly due to their high efficiency and potential for regenerative braking, contributing to reduced energy consumption and lower greenhouse gas emissions.

1.2 Problem Statement

Permanent Magnet Synchronous Motors are widely used in various industries due to their high efficiency and precise control capabilities. For many applications, sensorless speed control techniques have become increasingly attractive as they eliminate the need for physical speed sensors, reducing system complexity and cost. Among these techniques, the model reference adaptive system has shown effective performance in estimating motor speed without the need for sensors.

The aim of this study is to develop and implement MRAS speed estimator for sensorless speed control of PMSM. The primary focus will be on achieving regulate speed estimation while maintaining a stable and efficient control of the motor. To achieve this, a Proportional Integral (PI) controller will be utilized for speed and current control. The tuning of PI controller is crucial step in ensuring optimal performance and response characteristics of the system. Traditional methods for PI controller tuning may not fully capture the dynamic characteristics of the motor and the control system. To address this issue, fuzzy logic controller will be employed to tune PI controller parameters. Fuzzy logic offers robust and adaptive approach to parameter tuning, considering the nonlinearities and uncertainties present in control system. The FLC will dynamically adjust the PI controller gains to optimize the speed control performance under varying load conditions, disturbances, and operating points.

Therefore, the problem addressed in this study is the development of an effective MRAS based sensorless speed control technique for PMSMs using a fuzzy logic-PI controller. The technique aims to overcome the challenges associated with speed estimation and robust control under varying operating conditions. By addressing these challenges, the proposed control system will provide a reliable and cost effective solution for achieving sensorless speed control in PMSMs, eliminating the need for additional speed sensors and reducing system complexity.

1.3 Objectives of the Thesis

1.3.1 General Objective

The general objective of this study is to develop Model Reference Adaptive System (MRAS) based sensorless speed control technique for Permanent Magnet Synchronous Motors (PMSMs) using a fuzzy logic-PI controller. The aim is to achieve accurate and robust speed control of the PMSM without the need for additional speed sensors.

1.3.2 Specific Objectives

- To review existing sensorless speed control literature for PMSMs.
- To develop mathematical model of motor.
- To design FL-PI controller.
- To estimate the speed of PMSM using MRAS.
- To design and analyze speed control system for the motor using fuzzy logic-PI controller.
- To simulate and validate the performance of the sensorless speed control system through MATLAB® under various operating conditions and load torque disturbances.
- To compare and evaluate the performance of the fuzzy logic-PI controller with the conventional PI controller.

1.4 Methodology

This section presents the approach to achieve sensorless speed control of permanent magnet synchronous motors using MRAS and fuzzy logic-PI controller.

Literature Review

A comprehensive review of existing literature will be conducted to gather information on sensorless speed control techniques for PMSMs. The focus on MRAS based methods.

Mathematical Model Development

The model of the PMSM system will be developed, including its electrical equations and mechanical dynamics. The model will be validated through comparisons with known PMSM characteristics and experimental data.

MRAS Estimator Design

The MRAS estimator will be designed for sensorless speed estimation of motor. The reference and adjustable models will be carefully selected, considering the system's dynamics and parameter variations.

Fuzzy Logic-PI Controller Design

Fuzzy logic-PI controller will be designed for speed regulation of the PMSM. The PI gains will be optimized to achieve stable and accurate speed control.

Simulation and Validation

The proposed sensorless speed control system will be simulated using computer-based simulations. The system's performance will be evaluated under various operating conditions and load torque disturbances.

Recommendations

Recommendation outlines future research pathways and potential areas for refining sensorless speed control in PMSMs

1.5 Contribution of the Research Study

The research study on MRAS based sensorless speed control of permanent magnet synchronous motor using fuzzy logic-PI controller contributes to the field of sensorless speed control for permanent magnet synchronous motors by integrating model reference adaptive system estimation with fuzzy logic-PI controller.
The principal contributions of this thesis involve:

- Enhanced Sensorless Control and speed estimation accuracy and robustness:
- Improved Performance by designing of the fuzzy logic-PI controller:
- Evaluation and comparison of performance:

Overall, the research study contributes to the advancement of sensorless speed control for PMSMs by proposing MRAS estimation with fuzzy logic-PI controller. The integration speed estimator and controller enhances the accuracy, robustness and regulate speed of motor.

1.6 Structure of the Thesis

The thesis organized into the following chapters:

1. **Chapter 1: Introduction** This chapter provides overview of research study, including background, problem statement, research objectives and the organization of thesis.

2. **Chapter 2: Literature Review** In this chapter, a comprehensive review of existing literature is presented. The review focuse on speed control techniques for permanent magnet synchronous motors, MRAS based approaches, the chapter provides a critical analysis of previous research and identifies the research gaps.

3. **Chapter 3: System Modeling** This chapter covers the modeling of the PMSM, encompassing electrical equations and the motor's mechanical dynamics.

4. **Chapter 4: Fuzzy Logic-PI Controller Design** In this chapter, the design of the fuzzy logic-PI controller for speed regulation of the PMSM is presented. Explores PI controller gain tuning and includes the design of current controllers. Discusses and design MRAS estimator.

5. **Chapter 5: Simulation Results and Discussion** The chapter describes system integration and presents simulations assessing control system performance under varied conditions, considering load torque disturbances and parameter uncertainties. Analyzes results, focusing on stability, accuracy and robustness.

6. **Chapter 6: Conclusion and Future Work** The final chapter summarizes the key findings, discusses study implications and identifies future research directions and areas for improvement in sensorless speed control for PMSMs.

Chapter 2

Literature Review

2.1 Theoretical Background

A Permanent Magnet Synchronous Motor (PMSM) is an electric motor characterized by its utilization of permanent magnets embedded in the rotor and the synchronization of rotor speed with the frequency of the stator's alternating current. PMSM is a type of electric motor used in industrial and automotive applications due to its high efficiency, power density, and controllability [3]. The field of sensorless speed control in permanent magnet synchronous motor has remarkable advancements in recent years, driven by the integration of innovative techniques like model reference adaptive systems and fuzzy logic controllers. This literature review aims to explore the collaboration between MRAS and fuzzy logic in achieving accurate speed control without the need for direct speed sensors in PMSMs [4]. The ability to ensure precise speed regulation holds pivotal importance across diverse applications, ranging from industrial automation to electric vehicles. Permanent magnet synchronous motors get attention due to their high efficiency, compact size, and reliable operation. Effective speed control in these motors is a critical factor influencing their performance, efficiency, and overall operational characteristics. Conventionally, speed control has been achieved using Proportional Integral (PI) controllers [5]. However, such controllers often struggle to handle the nonlinear and time-varying nature of PMSMs, especially under varying load conditions and disturbances.

The adoption of MRAS and fuzzy logic in sensorless speed control addresses these challenges by providing adaptive and robust control strategies. MRAS estimates motor speed using mathematical models for dynamic adjustments, while fuzzy logic controller fine-tunes PI controller parameters in various conditions.

2.1.1 Construction of Permanent Magnet Synchronous Motor

The construction of Permanent Magnet Synchronous Motor (PMSM) consists of two main components the stator and the rotor.

The stator is the stationary part of the motor and is typically made of laminated steel sheets. Typically, the stator windings consist of three phases and connected to the power supply to create a rotating magnetic field [6]. The rotor is the rotating part of the motor and is mounted on the motor shaft. It contains permanent magnets, often made of ferrite materials, which generate a constant magnetic field. The rotor can be either surface mounted or embedded (buried) magnets, depending on the motor's design.

2.1.2 Working Principle of PMSM

The working principle of a Permanent Magnet Synchronous Motor (PMSM) is based on the interaction between the stator's rotating magnetic field and the rotor's permanent magnet field. The motor operates as follows [7]:

Stator Magnetic Field Generation: When 3 phase AC currents are supplied to stator windings, magnetic field is created in the air gap between stator and rotor. The rotating magnetic field is generated due to the phase shift between the three stator windings, typically 120 degrees apart.

Rotor Alignment: The constant magnetic field generated by the rotor's permanent magnets interacts with the rotating magnetic field generated by the stator. As a result, the rotor experiences torque, causing it to align itself with the rotating stator field. The rotor aligns its magnetic axis with respect to rotating stator field of magnetic axis.

Synchronous Operation: The PMSM operates in synchronism with the supply frequency. The rotational speed at which motor rotates is directly related to the frequency of the voltage it receives and is determined by number of pole pairs present in motor. The synchronous speed can be calculated using the formula: $N_s = \frac{120 \times f}{P}$, where N_s is the synchronous speed in RPM, f is the supply frequency in Hz, and P is the number of pole pairs.

Torque Generation: Once the rotor aligns with the rotating stator field, torque is generated as a result of magnetic attraction and repulsion between stator and rotor fields. This produced torque drives the motor shaft's rotation.

2.1.3 Types of Permanent Magnet Synchronous Motors (PMSM)

Permanent magnet synchronous motors come in various configurations, each designed for specific applications. The two main types are:

Surface Permanent Magnet (SPM) Motors: In SPM motors, the permanent magnets are mounted on the rotor's surface, facing the stator. These motors suit high torque and efficiency demands. They are widely used in industrial machinery, robotics, and electric vehicles [3] [8].

Interior Permanent Magnet (IPM) Motors: In IPM motors, the permanent magnets are embedded within the rotor core, also known as the rotor back iron. IPM motors are often preferred for requiring high torque at low speeds, including electric and hybrid vehicles, air conditioning systems and industrial pumps, among others. .

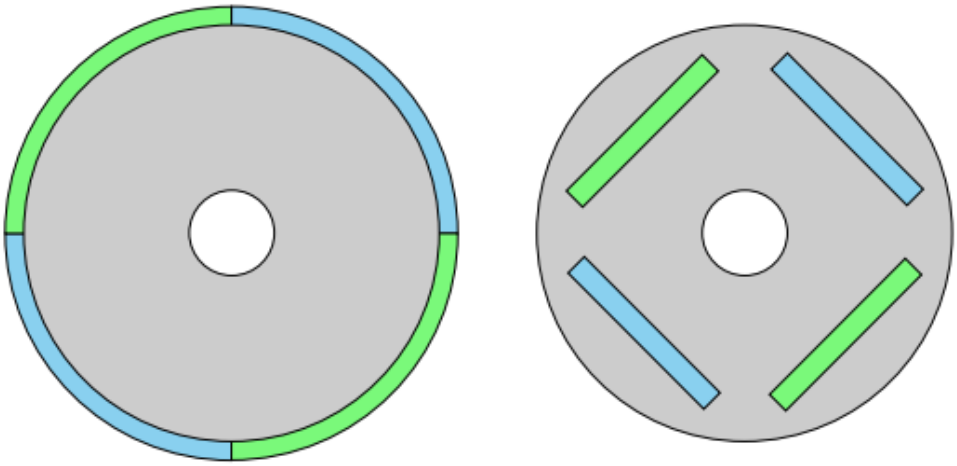

Figure 2.1: The Configuration for both SPMSM and IPMSM [9]

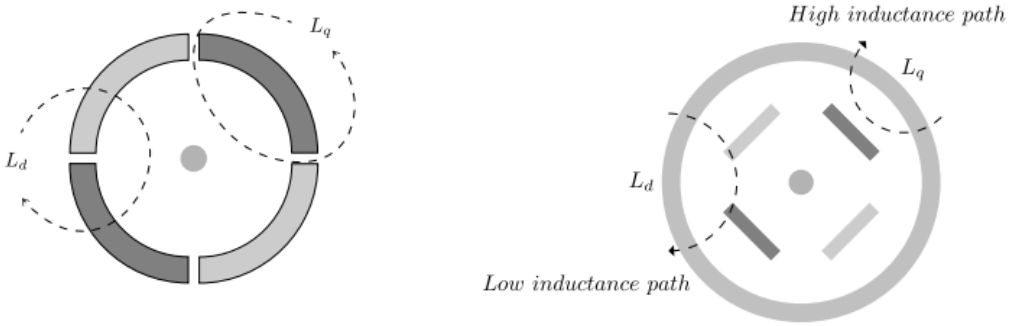

Figure 2.2: Distribution of Magnet in the Rotor [9]

As shown in figure 2.1 and 2.2 to left SPMSM the magnets are uniformly distributed on the outer surface and the stator inductance (Ls =Ld=Lq) not dependent on rotor's position.

2.1.4 Proportional Integral controller

The Proportional Integral (PI) controller is widely employed feedback control mechanism extensively used in the control of PMSM. It operates within a feedback loop to regulate the motor's speed, current and other parameters according to desired set points. The proportional term of the PI controller computes an output based on the current error, which represents the difference between the desired value (setpoint) and the actual value (measured value, such as motor speed). Larger error results in a stronger corrective action. The integral term of the PI controller considers the accumulation of past errors over time. It effectively eliminates any steady state error that might persist due to factors like system biases, disturbances, or model inaccuracies. In the context of PMSM control, the integral term contributes to gradually reducing persistent deviations from the desired speed, ensuring accurate and stable speed regulation.

The PI controller role in a PMSM control system involves continuous evaluation of the error between desired and actual motor speed. It employs a combination of proportional and integral terms to adjust control signals, ensuring that the PMSM accurately and efficiently maintains the desired speed, even in the presence of disturbances or uncertainties.

2.1.5 Fuzzy Logic Controller

In the domain of controlling PMSMs, integration of a Fuzzy Logic Controller (FLC) introduces intelligent and effective approach for tuning proportional-integral controller gains. FLC leverages linguistic rules and input data to dynamically adjust the PI controller parameters, optimizing motor performance in diverse operating conditions [10]. Specifically focusing on PI controller gain tuning, the FLC operates by analyzing error and its rate of change, interpreting these inputs using linguistic rules, and generating appropriate adjustments to the gains. This adaptive mechanism enhances the PMSM's control precision, responsiveness and stability.

Incorporating FLC within motor control systems to adjust PI controller gains represents an advanced and efficient method for optimizing motor performance [4]. By harnessing linguistic rules and real time error analysis, FLC improves the flexibility and stability of the control system, ultimately contributing to superior PMSM operation and control under varying conditions.

2.2 Particle Swarm Optimization

In this thesis, Particle Swarm Optimization (PSO) is employed to determine scaling factors for a fuzzy logic controller and to fine tune the parameters of a proportional integral controller within adaptable mechanism for speed control of PMSMs. PSO simulates the social behavior of particles within a multidimensional space. It adjusts the scaling factors of the FLC to minimize a defined cost function, thus enabling an accurate representation of the system's complexities. Moreover, PSO optimizes the parameters of the PI controller, facilitating precise speed regulation of PMSMs by dynamically adapting control actions based on real time feedback. This integrated approach enhances efficiency, accuracy, and adaptability across a range of conditions.

Concise step by step guide for using particle swarm optimization to determine scaling factors and for tuning a PI controller

1. **Define Objective:** Specify the control problem and desired improvement in performance.
2. **Representation:** Model particles as scaling factor vectors for the PI controller.
3. **Initialization:** Create a particle population with random scaling factors.
4. **Fitness Function:** Develop a fitness function based on control metrics (e.g., overshoot, settling time).
5. **PSO Steps:** Update velocities, positions, personal bests, and neighborhood bests.
6. **Constraints:** Define valid ranges for scaling factors.
7. **Termination:** Set stopping criteria (iterations, fitness threshold).
8. **Implementation:** Code the PSO algorithm (MATLAB).
9. **Execution:** Run PSO to find optimized scaling factors.
10. **Validation:** Test optimized factors in the control system, compare results.

2.2.1 Sensorless Speed Control Techniques

Various sensorless speed control techniques have been proposed for PMSMs, including model based and sensor based approaches. The literature review highlights the importance of sensorless control techniques in reducing system complexity, cost and maintenance requirements. Model based techniques, such as the model reference adaptive system method, have gained significant attention due to their ability to estimate speed of rotor without the need of sensors. The MRAS technique utilizes a reference and adjustable models to estimate rotor speed based on the motor's electrical and mechanical characteristics.

2.2.2 Overview of the Existing Literature Review

Sensorless control of PMSMs goal of eliminating the requirement for physical sensors to measure rotor position and speed or current. This approach leverages advanced estimation algorithms and signal processing techniques to infer the motor's state variables indirectly. Various sensorless control approach's, such as back electromotive force estimation, observer based methods, EKF, Sliding Mode Observer (SMO) and MRAS have been explored in this literature. In recent times, multiple academic publications have proposed a variety of speed estimation methods with the intention of enabling sensorless operation of PMSM drives.

The Authors Genduso, Fabio and Miceli, Rosario and Rando, Cosimo and Galluzzo, Giuseppe Ricco [11] and Nahid-Mobarakeh, Babak and Sargos, Franois-Michel [12] presented Sensorless control method based on emf approach is likely the predominant technique for achieving Control without sensor in PMSMs. Straightforward and uncomplicated to compute, yet it exhibits excellent performance in high speed control scenarios. However, a recognized drawback of the back electromotive force approach for estimating rotor speed of motor is its dependency on motor parameters and operating conditions. This method relies on measuring the back electromotive force induced in the motor windings, which is directly corresponding to speed. However, variations in motor parameters such as resistance, inductance, and magnet flux can introduce inaccuracies in the speed estimation. Additionally, it fails to operate during standstill, evident from a weak or absent signal [13] [14]. Its accuracy is also affected by noise and measurement errors, making it less reliable in noisy environments. Consequently, the Back-EMF method's accuracy and effectiveness heavily depend on the accuracy of motor parameter values and may require compensation techniques to enhance performance across a range of operating conditions. In this thesis proposed that MRAS

based on stator current can be used to determine the speed of motor. This approach is insensitive to changes in parameters and has the ability to estimate over a wide range of speeds, in contrast to the back EMF method.

In 2023 Lu, Jianning , Wenbin , Zhang, Junshuai and Su, Jiaqi [15], In 2022 Li, Yanhao and Lan, Zhiyong and Su, Xiaoyang and Dai, Shanqi [16], In 2021 Allaoui, Samia and Laamari, Yahia and Chafaa, Kheireddine and Saad, Salah [17] and In 2019 Talib [18] introduced the extended kalman filter method is an alternative approach for estimate rotor speed and position of PMSMs. However, drawbacks associated with the EKF approach for estimate speed of motor computational complexity, The EKF method involves multiple matrix multiplications and recursive calculations during each iteration of the estimation process. This leads to a significant computational load, making it computationally expensive and resource intensive, especially for real time applications. The processor requirements are higher compared to MRAS speed estimation methods. High sensitivity to initial conditions, The EKF's accuracy and stability are highly dependent on the initial conditions and parameter tuning. Incorrect initial conditions or poor tuning can lead to divergence or inaccurate speed estimates. Ensuring proper initialization and tuning can be challenging and time consuming, assumption of gaussian noise, EKF assumes that noise in the system follows a zero mean gaussian distribution. However, real world systems may have non gaussian components. This assumption can lead to sub optimal performance if the noise characteristics deviate significantly from the assumed model, model inaccuracies while EKF attempts to account for model inaccuracies, it may struggle with complex nonlinearities and uncertainties present in PMSM systems. If the model used for the filter does not accurately represent the motor's behavior, the estimation accuracy can be compromised, limited adaptability EKF relies on a fixed model structure throughout the estimation process. If the system undergoes significant changes or variations in its dynamics, the EKF might struggle to adapt effectively, resulting in degraded performance, resource intensive in addition to computational demands, EKF requires knowledge of system parameters and noise characteristics. Accurate modeling and parameter identification are essential for optimal EKF performance, which can be time consuming [19]. So MRAS has advantages over EKF for speed estimation due to its simplicity, lower computational complexity, robustness to parameter variations and lower processor requirements.

In the literature, Bist, Abhishek, Jadhav, Sadhana (2020) [20] and Kim, Son, Jubum and Lee (2010) [21] introduced Control method using SMO. A commonly recognized drawback

associated with SMO [22], [23]. is the occurrence of chattering behavior, which involves rapid and high-frequency switching of control actions. This can lead to mechanical stress, acoustic noise, and instability in the system. The method also requires complex tuning of parameters to balance robustness and convergence speed and it is sensitive to noise and disturbances. The discontinuous control actions can result in non-smooth speed estimates, and the method may struggle with adapting to parameter variations or nonlinearities. Additionally, implementing sliding-mode techniques can be challenging due to hardware and timing constraints. In summary, the literature review underscores MRAS as superior to the sliding-mode observer for speed estimation in PMSMs. MRAS's adaptability, noise resistance and simpler tuning render it a preferable option for accurate and robust speed estimation.

In 2017, the work of Siraj Hajo and a study by Mahlet Legesse in 2011 [24] [25] respectively presented Control approach utilizing a fuzzy logic-PI controller. The PI controller's performance is influenced by changes in system parameters caused by internal and external conditions. To address this challenge, a solution was proposed involving utilizing a fuzzy logic controller approach for online tuning of PI controller parameters through sensors. This method aimed to enhance performance over a wide speed range and in the presence of varying load disturbances. However, it's worth noting that the inclusion of a speed sensor can lead to increased system costs.

Hussain, Shoeb and Bazaz presented a study titled [26]. The research introduces neural network observer to estimate rotor speed through current and voltage components. The neural network estimator ensures better speed observer while minimizing computational complexity. The incorporation of a Sliding Mode Control (SMC) controller enhances drive performance across varying load conditions. Although the system performs well under load variations and exhibits robust speed estimation despite parameter uncertainty, the learning rate of the neural network leads to prolonged processing times. Furthermore, the implementation of SMC introduces a chattering issue in the speed controller output [20].

Authors	Title	Key Findings	Gaps
Genduso,Fabio and Miceli [11] Babak, Franois-Michel [12]	High-performance PMSM control through back EMF sensorless algorithm	• Speed estimation is straightforward • Demonstrates excellent performance in high-speed control	• Ineffective at low and standstill speeds • Sensitive to alterations in parameter values
Lu,Jianning, Wenbin [15] Li,Yanhao [16] Allaoui, Samia, Salah [17] and Nordin, Mohamad [18]	PMSM sensorless control with Extended Kalman Filter	• It is capable determining speed under all operational circumstances	• It does not take into account the effectiveness of speed control (PI) • it places significant demands on the processor.
Bist,Sadhana [20], Kim,Son,Jubum, and Lee [21]	Sliding mode observer for sensorless PMSM control	• Simple design • Accurate speed estimation even at high speeds • Robust to changes in parameters	• Issues at standstill and low speeds • Undesirable chattering phenomenon • Issues with time delay.
Siraj Hajo and Mahlet Legesse [24] [25]	PMSM drive speed controlled by a fuzzy logic-PI controller	• Online tuning of PI speed controller via FLC • well despite parameter changes and variable load torque	• need physical sensors to measure rotor speed and position • Due to the use of sensors, the system costs have increased
Hussain, Shoeb, and Bazaz presented a study titled [26]	Utilizing neural network observer for sensorless PMSM drive control	• Not influenced by motor-parameters for speed estimation. • It also features a robust speed controller.	• Lacks of adaptation for speed estimation • execution times complexity and challenging implementation.

Table 2.1: Summary of Literature Review on Speed Control Methods for PMSM

	Back-EMF	MRAS	EKF	SMO
Processor Requirements	Low	Low	High	Low
Estimation Accuracy	Satisfactory	Good	Excellent	Good
Sensitivity to Variations	Poor	Can be fixed	Decent	Decent
Resistance Estimation	No	Yes	Probably	No
Re-tuning Ability	Decent	Decent	Poor	Decent

Table 2.2: Comparison of Sensorless Control Methods for PMSMs

Chapter 3

Mathematical Modeling of PMSM

3.1 Coordinate Frames Transformation

Coordinate frame is an essential step when mathematically modeling the dynamics of Permanent Magnet Synchronous Motor (PMSM). The coordinate frame chosen for analysis greatly simplifies the equations governing the motor's behavior. For designing of the vector control method transformations are necessary for eliminate cross couplings to enhance performance of the control strategies [27]. In this sub section we are going to describe the two most known approaches for coordinate transformation, park transformation and clarke transformation. The coordinate axes of the clark and park transformations are depicted below. .

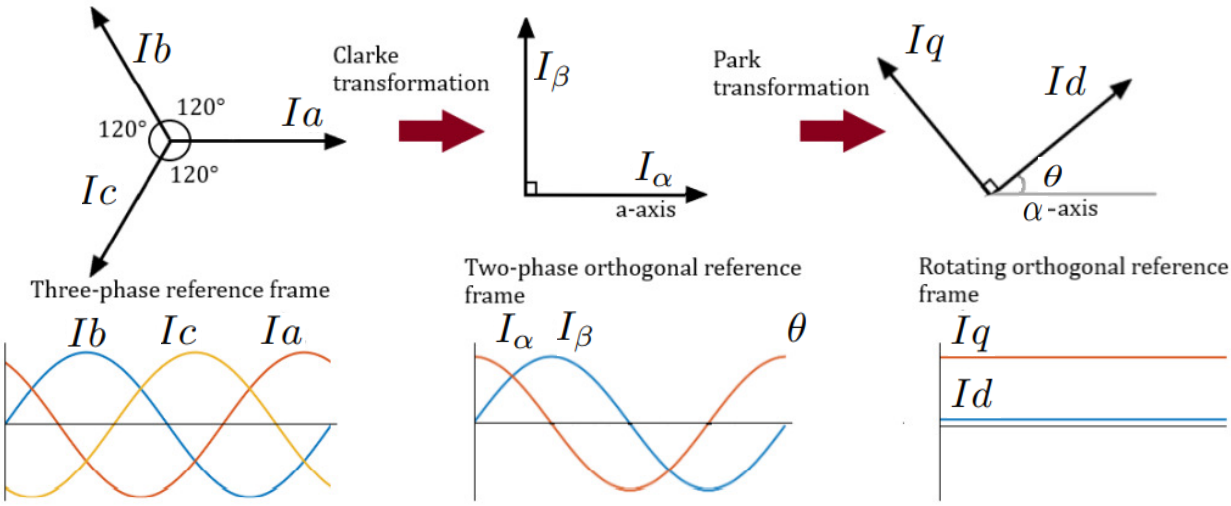

Figure 3.1: The Coordinate Axes in the Three Reference Frames [5]

3.1.1 Clarke Transformation

The clarke transformation converts the 3-phase stator currents from the stationary frame (α-β) to the two-phase frame (d-q). The transformation equations are given as follows:

$$\begin{bmatrix} Id \\ Iq \end{bmatrix} = \begin{bmatrix} \cos(\theta) & \sin(\theta) \\ -\sin(\theta) & \cos(\theta) \end{bmatrix} \begin{bmatrix} I\alpha \\ I\beta \end{bmatrix} \tag{3.1}$$

3.1.2 Park Transformation

The park transformation converts stator currents from stationary frame (d-q) to rotating frame (α-β) in alignment with rotor flux. These transformation equations are presented as:

$$\begin{bmatrix} I\alpha \\ I\beta \end{bmatrix} = \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix} \begin{bmatrix} Id \\ Iq \end{bmatrix} \tag{3.2}$$

where: $I\alpha$ and $I\beta$ are the three phase stator currents in the stationary reference frame.

3.1.2.1 Clarke Park Transformations

The dynamics of a three phase motor can be understood from various perspectives by employing multiple frames [28]. The three-phase stationary reference frame is the most commonly used framework often referred to as the abc-frame. In this frame, which is physically aligned with motor, and its three axes identified as a, b, and c, correspond to three phases situated within the stator. Current and voltage measurements are conducted within the abc-frame. However, it can be challenging to determine magnitude and angle within abc-frame, necessitating its conversion into a two-phase system. To address this issue, clarke and park transformations, along with their inverses, have been introduced. These transformations allow for the conversion of the stator three phase currents into rotating frame [29]. Through clarke conversion changes 3-phase measurements into a stationary two-axis orthogonal frame. While these measurements are set within frame, the rotor's frame keeps a continuous rotation [30].

The combined conversions of Clarke and Park can be expressed as follows

$$\begin{bmatrix} Id \\ Iq \end{bmatrix} = \frac{2}{3} \begin{bmatrix} \cos(\theta) & \cos\left(\theta - \frac{2\pi}{3}\right) & \cos\left(\theta + \frac{2\pi}{3}\right) \\ \sin(\theta) & \sin\left(\theta - \frac{2\pi}{3}\right) & \sin\left(\theta + \frac{2\pi}{3}\right) \end{bmatrix} \begin{bmatrix} Ia \\ Ib \\ Ic \end{bmatrix} \tag{3.3}$$

The inverse transformation is similarly expressed as follows

$$\begin{bmatrix} Ia \\ Ib \\ Ic \end{bmatrix} = \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \cos\left(\theta - \frac{2\pi}{3}\right) & -\sin\left(\theta - \frac{2\pi}{3}\right) \\ \cos\left(\theta + \frac{2\pi}{3}\right) & -\sin\left(\theta + \frac{2\pi}{3}\right) \end{bmatrix} \begin{bmatrix} Id \\ Iq \end{bmatrix} \tag{3.4}$$

The details of these reference frame transformations are shown in figure 3.2.

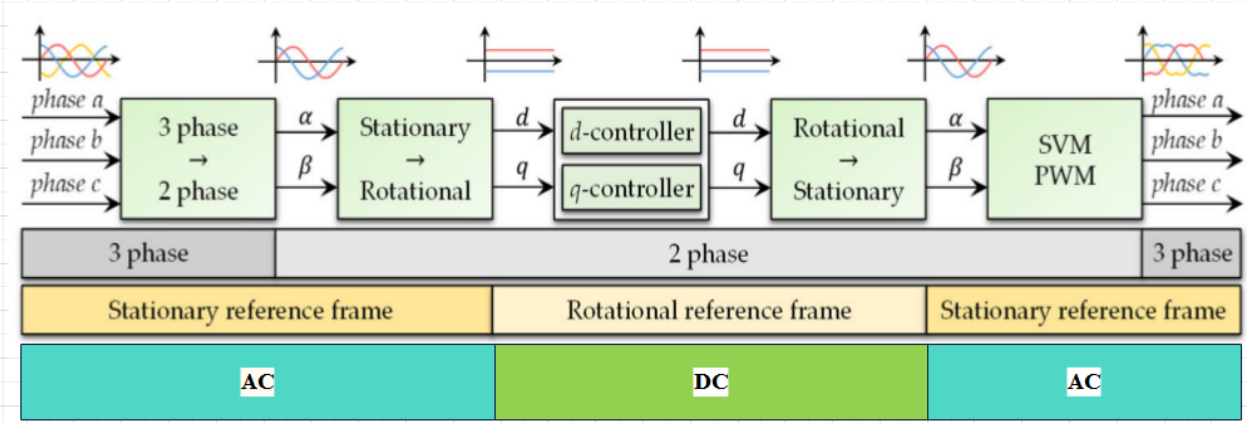

Figure 3.2: Transformations for Field-Oriented Control

3.1.3 Field Oriented Control

Field Oriented Control (FOC) is a control strategy widely used in the control of motor to achieve high performance speed and torque control. The FOC technique can be implemented to enhance the control system's performance and accuracy. The FOC technique involves transforming three phase stator currents of motor into 2-coordinate frame, typically referred to as d-q frame. In this frame, the control variables are the direct axis current (Id) and the quadrature axis current (Iq), which correspond to the torque and magnetizing current components, respectively. This control technique allows independent control of these two components, enabling precise torque and flux control [4].

The field oriented control algorithm consists of the following main steps:

Step 1. Clarke Transformation: The stator currents are converted to a two-coordinate reference (d-q) frame. This transformation converts three-phase currents into their corresponding d-q components.

Step 2. Park Transformation: The transformed currents from the clarke transformation are further transformed into a rotating frame that aligns with rotor's position magnetic field. The park transformation involves angle transformation based on the rotor position to obtain the Id and Iq currents.

Step 3. Current Control: The FOC employs PI controllers to regulate the Id and Iq currents. The PI controllers adjust the motor's voltage or current references to track the desired torque and flux components.

Step 4. Inverse Park Transformation: After the current control, the control signals from the PI controllers are converted back to stationary frame.

Step 5. Inverse Clarke Transformation: is applied to obtain the three phase voltage references from the transformed control signals in the stationary reference frame.

Step 6. SVPWM: The voltage references obtained from the inverse clarke transformation are used to generate switching signals for PMSM power electronic converter. The SVPWM technique is commonly employed to generate switching patterns to control voltage applied to motor phases.

3.2 Motor Model in abc Frame

The modeling of SMPMSM is similar to classical synchronous motor the difference is flux from the magnets is constant on PMSM [31]. To simplify the the dynamic model of surface permanent magnet synchronous motor the following assumptions are included [32].

1. Saturation and machine core losses are negligible.
2. Hysteresis and eddy current losses are neglected.
3. There are no field current dynamics and induced EMF is sinusoidal.
4. Consider three phase supply voltage and constant rotor flux.

Based on the above assumptions and basic concepts, both mechanical and electrical equations introducing the dynamic property of the motor. The model of motor in the natural three phase reference frame from the stator's perspective represents, PMSM as a simple three phase balanced system in the stationary frame, as depicted in figure 3.3.

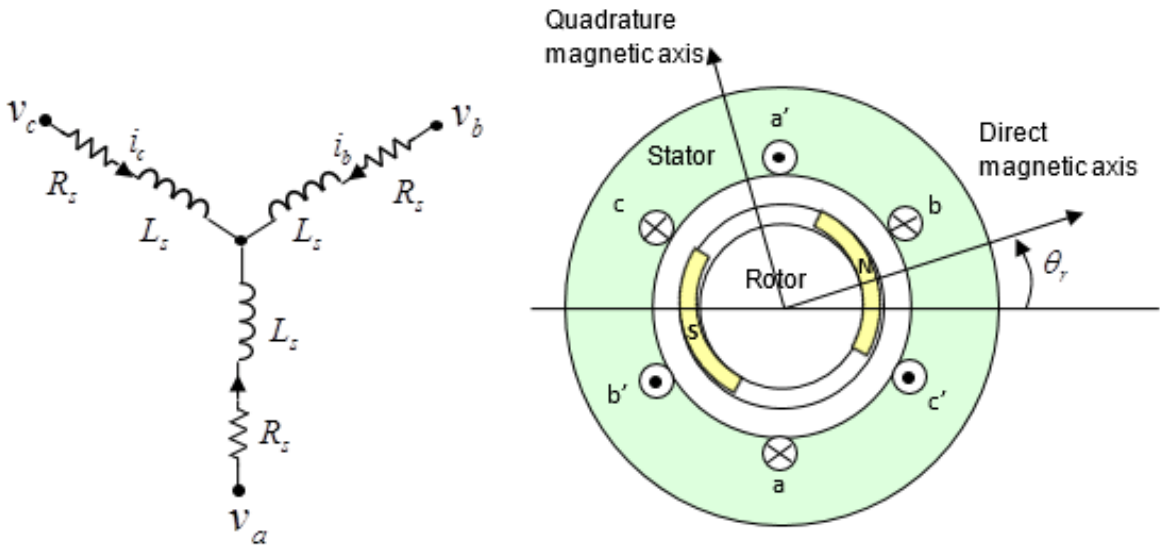

Figure 3.3: Three Phase Equivalent Model of Permanent Magnet Synchronous Motor [33]

The voltage equation for three phase stationary frame representation of motor is given by

$$\begin{bmatrix} V_a \\ V_b \\ V_c \end{bmatrix} = R_s \begin{bmatrix} i_a \\ i_b \\ i_c \end{bmatrix} + \frac{d}{dt} \begin{bmatrix} \Lambda_s a \\ \Lambda_s b \\ \Lambda_s c \end{bmatrix} \tag{3.5}$$

Where: $V_{abc} = 3$ phase voltages, $R_s =$ Stator resistance, $\Lambda_{sabc} =$ Stator flux, $i_{abc} =$ phase currents, and Λ_{sabc} [34]. The stator flux components Λ_{sa}, Λ_{sb}, and Λ_{sc} are given by:

$$\begin{bmatrix} \Lambda_{sa} \\ \Lambda_{sb} \\ \Lambda_{sc} \end{bmatrix} = L_s \begin{bmatrix} i_a \\ i_b \\ i_c \end{bmatrix} + \begin{bmatrix} \Lambda_{ra} \\ \Lambda_{rb} \\ \Lambda_{rc} \end{bmatrix} \tag{3.6}$$

The primary element of the matrix denotes the flux created by the current flowing through the stator coils, while the second element refers to the flux induced in each stator phase by the magnets within the rotor. Where: $L_s =$ Inductance matrix, $\Lambda_r abc =$ flux produced by the permanent magnets.

$$\begin{bmatrix} \Lambda_r a \\ \Lambda_r b \\ \Lambda_r c \end{bmatrix} = \Lambda_r \begin{bmatrix} cos(\theta_e) \\ cos(\theta_e - \frac{2\pi}{3}) \\ cos(\theta_e + \frac{2\pi}{3}) \end{bmatrix} \tag{3.7}$$

The three phase stationary frame representation of PMSM can be written as,

$$\begin{bmatrix} V_a \\ V_b \\ V_c \end{bmatrix} = R_s \begin{bmatrix} i_a \\ i_b \\ i_c \end{bmatrix} + L_s \frac{d}{dt} \begin{bmatrix} i_a \\ i_b \\ i_c \end{bmatrix} + \Lambda_r \frac{d}{d\theta_e} \begin{bmatrix} cos(\theta_e) \\ cos(\theta_e - \frac{2\pi}{3}) \\ cos(\theta_e + \frac{2\pi}{3}) \end{bmatrix} \tag{3.8}$$

3.3 Motor Model in dq Frame

The dynamic model of the surface-mounted permanent magnet synchronous motor (SMPMSM) is derived from the direct and quadrature reference frame axes. This approach offers conceptual simplicity by obtaining two windings on the stator [35]. Through coordinate frame transformation, motor model in the rotating frame presents the voltage equations as follows

$$Vd = R_s Id + Ld\frac{dId}{dt} - \omega_e LqIq \tag{3.9}$$

$$Vq = R_s Iq + Lq\frac{dIq}{dt} + \omega_e LdId + \omega_e \lambda_f \tag{3.10}$$

Rearranging equations 3.9 and 3.10 to derive the model of the surface permanent magnet synchronous motor from the differential equations of different state variables of the motor. In the motor model in the dq frame that rotates alongside the rotor, the current dynamics within the dq axis can be described as follows:

$$\frac{dId}{dt} = -\frac{R_s}{Ld}Id + \frac{Lq}{Ld}R_s\omega_e Iq + \frac{Vd}{Ld} \tag{3.11}$$

$$\frac{dIq}{dt} = -\frac{R_s}{Lq}Iq - \frac{L_d}{Lq}\omega_e Id - \frac{1}{Lq}\omega_e\lambda_f + \frac{Vq}{Lq} \tag{3.12}$$

The equations 3.11 and 3.12 can also be expressed in state space representation utilizing the state vector $x^T = [\ \text{id iq}\ ]$.

$$\begin{cases} \dot{x} = Ax + Bu \\ y = Cx \end{cases} \tag{3.13}$$

$$A = \begin{bmatrix} -\frac{R_s}{Ld} & \omega_e \\ -\omega_e & -\frac{R_s}{Ld} \end{bmatrix} ; B = \begin{bmatrix} -\frac{V_d}{Ld} \\ \frac{1}{Ld}[V_d - \lambda_f\omega_e] \end{bmatrix} ; C = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

The produced motor torque can be formulated in the dq frame as follows

$$Te = \frac{3}{2}P(\Lambda_d Iq - \Lambda_q Id) \tag{3.14}$$

Where: $\Lambda_d = LdId + \lambda_f$ and $\Lambda q = LqIq$, next the equation transforms into 3.15.

$$Te = \frac{3}{2}P(LdId + \Lambda_f)Iq - LqIqId) \tag{3.15}$$

As the motor is an SMPMSM, it was presumed that all phase resistances and inductances are identical: $R_d = R_q = R_s$ and $L_d = L_q = L_s$. Consequently, the mechanical equations can be reformulated as follows:

$$T_e = \frac{3}{2}P\Lambda_f Iq \tag{3.16}$$

Equation (3.16) is crucial for field oriented control as it allows controlling the torque (Iq) and flux (Id) separately within a motor. Notably, in a saliency free the permanent magnet synchronous motor, torque production relies solely on the stator q-axis current. The PMSM's key properties include stator pole pairs, which we consider in this thesis concerning mechanical and electrical characteristics like speed, angle and electrical parameters. These parameters are vital for estimating and ensuring control algorithm stability. Additionally equation (3.17) highlights the consistent relationship between electrical and mechanical position and speed.

$$\begin{cases} \omega_e = P\omega_m \\ \theta_e = P\theta_m \end{cases} \tag{3.17}$$

Where: ω_e, ω_m is electrical mechanical speed of the motor respectively, θ_m is rotor position, θ_e is electrical angle of rotor and P is pole pairs. The general equation describing the mechanical of the motor is stated as follows.

$$J\frac{d\omega_m}{dt} = T_e - T_L - B\omega_m \tag{3.18}$$

substituting equation 3.15 into equation (3.18), mechanical equations of the motor can be written as

$$J\frac{d\omega_m}{dt} = \frac{3}{2}P(L_dId + \Lambda_f)Iq - L_qIqId - T_L - B\omega_m \tag{3.19}$$

The rotor mechanical and electrical speed are related as. $\frac{d\theta_e}{dt} = \omega_e$ and $\frac{d\theta_m}{dt} = \omega_m$

3.4 Dynamic Model Verification of PMSM

3.4.1 Analysis of Open Loop Model for PMSM

The Open loop dynamic model can be obtained from equation (3.9) up to equation (3.17). As illustrated in figure 3.4 voltage and external torque are inputs and the motor speed, stator current and electromagnetic torque as outputs.

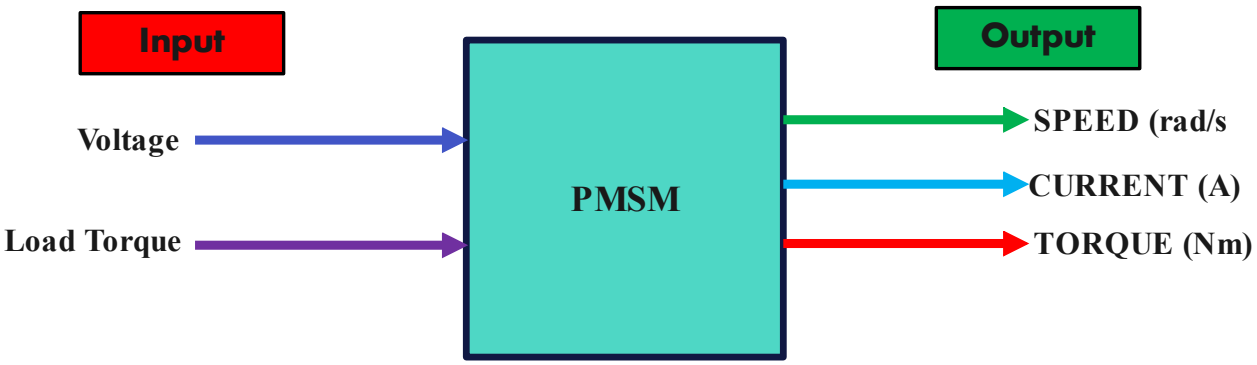

Figure 3.4: Open Loop Block Diagram of Permanent Magnet Synchronous Motor

The open loop since using vector control Vd, Vq are coupled to each other and Id=0. The open-loop transfer function can be derived as $G_o(s) = \frac{Y(s)}{U(s)}$, which provides from Vq to ω_m.

$$G_o(s) = \frac{\omega_m(s)}{Vq(s)} = \frac{\frac{3}{2}P\lambda_f}{(Js+B)(R_s+Lqs)+\frac{3}{2}P^2\lambda_f^2} \tag{3.20}$$

Equation (3.20) is the combination of electrical and mechanical transfer functions.

$$\begin{cases} G(s)elec = \frac{1}{Lqs+R_s} \\ G(s)mech = \frac{\frac{3}{2}P\lambda_f}{Js+B} \end{cases} \tag{3.21}$$

The mechanical speed (ω_m) is equal to ($\frac{\omega_e}{P}$), where ω_e is electrical speed, and P is pole pairs. The time constants [36] are $\tau_e = \frac{Lq}{R_s}$ and $\tau_m = \frac{J}{B}$, with τ_e being faster than τ_m. Id and Iq are derived from equation (3.9) and equation (3.10), electromagnetic torque is obtained from the mechanical model derived from equation (3.10), 3.14 up to 3.19 respectively.

3.5 Simulation Results for Model Verification of PMSM

3.5.1 Simulation Results of PMSM With Load

3.5.1.1 Simulation Result of Speed Response

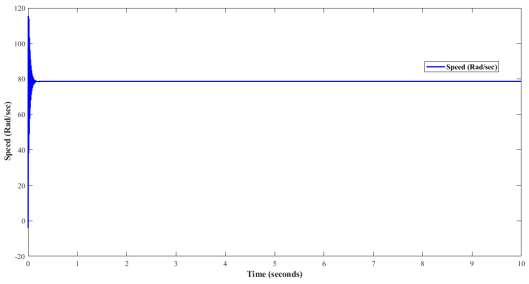

Figure 3.5: Speed Response at Load (5 Nm)

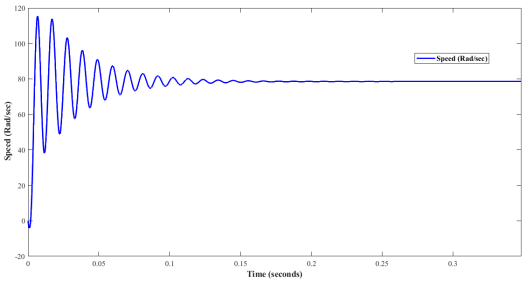

Figure 3.6: Zoomed Speed Response

The simulation results were conducted with a load torque of 5 Nm, as shown in figures 3.5 and 3.6. Initially, the speed response exhibits high overshoot and settles at 0.1 seconds

3.5.1.2 Three Phase Stator Current Response

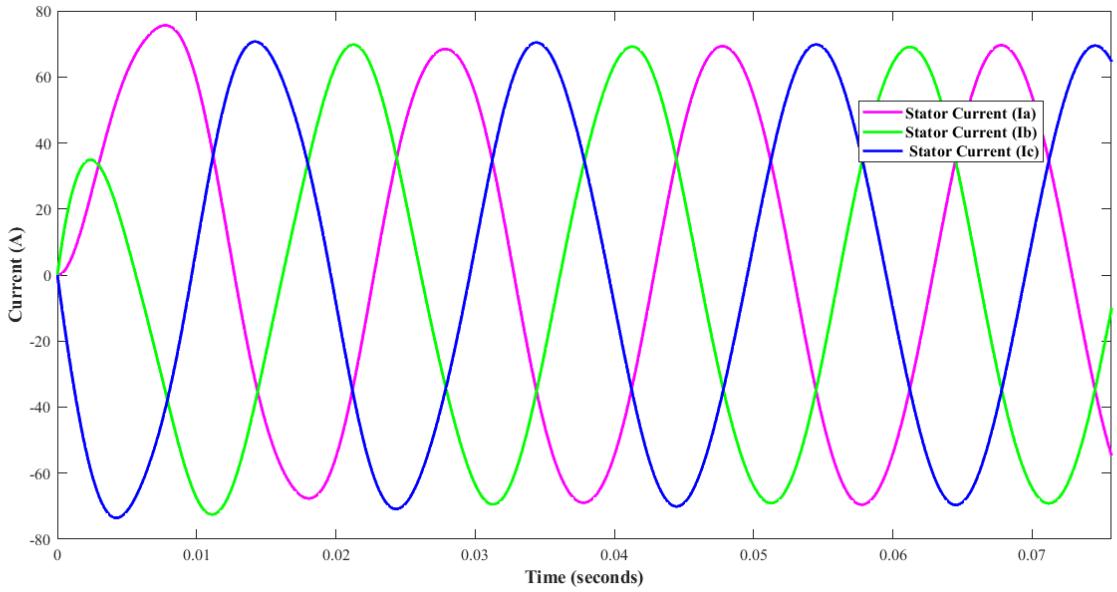

Figure 3.7: Current Response at Load (5 Nm)

At load conditions, the current magnitude increases with the mechanical load on the motor. The current produce the required torque to overcome the load.

3.5.1.3 Electromagnetic Torque Response

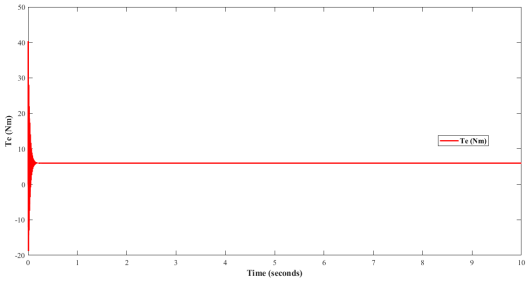

Figure 3.8: Torque under Load (5 Nm)

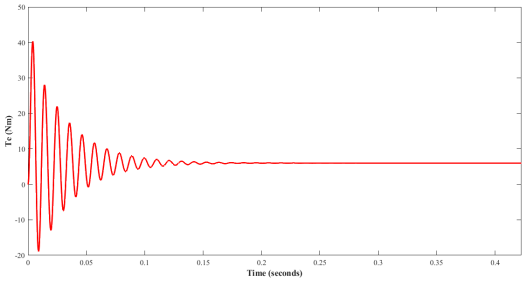

Figure 3.9: Zoomed Te Response

In loaded scenarios figure 3.9, stator current rises to meet the necessary electromagnetic torque to sustain motor speed and carry out mechanical operations, leading to a proportional upsurge in output torque.

3.5.2 Simulation Results of PMSM Without Load

3.5.2.1 Speed Response

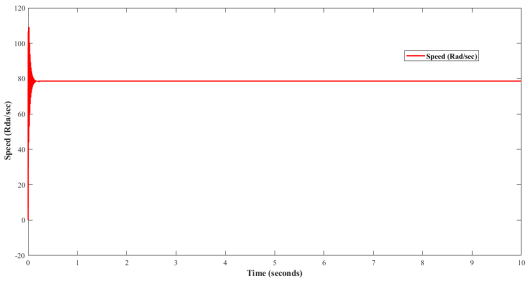

Figure 3.10: Speed Response without Load

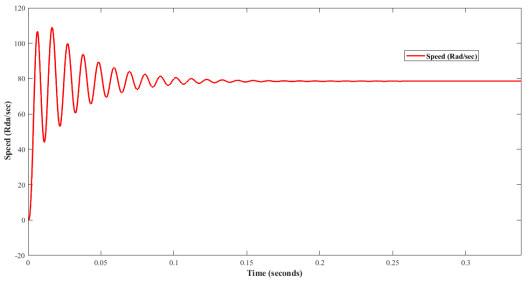

Figure 3.11: Zoomed Speed Response

.

The simulation involves without any load torque (0 Nm), as illustrated in simulation results, In both loaded and unloaded scenarios, motor's steady operation this demonstrates effectiveness of models.

3.5.2.2 Three Phase Stator Current Response

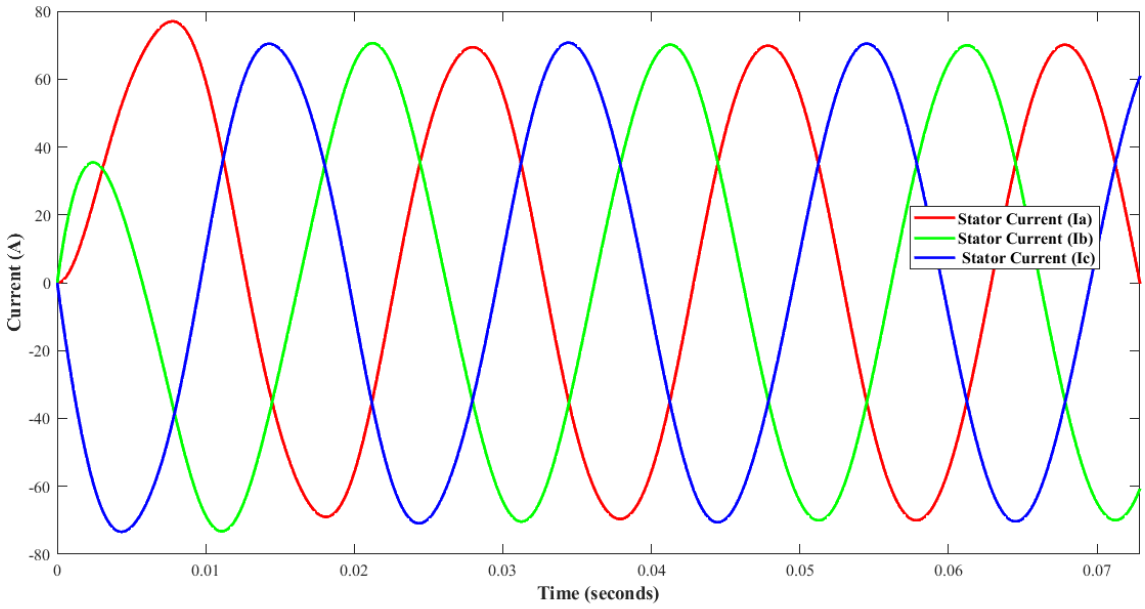

Figure 3.12: Stator Current under No-Load Conditions

As depicted in figure 3.12, in the no-load condition, motor draws the same amount of current as in the loaded scenario, while maintaining a constant speed in both cases.

3.5.2.3 Electromagnetic Torque Response

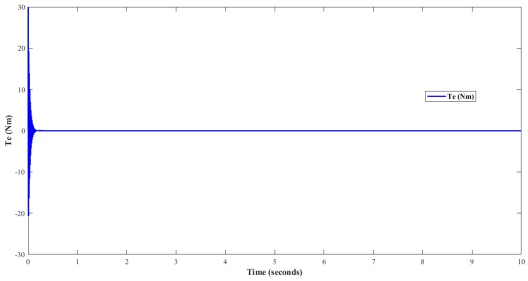

Figure 3.13: Torque (Te) Without Load

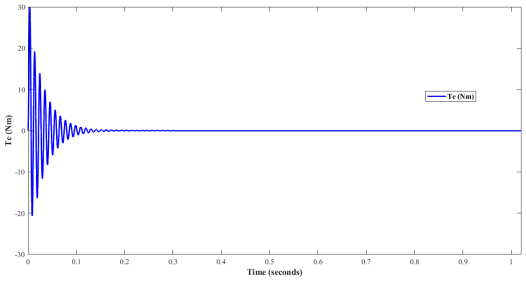

Figure 3.14: Zoomed Te Response

The simulation in figure 3.14 shows torque generated by the motor is zero. Since there is no mechanical load resisting the motor rotation, there is no need for the motor to produce significant torque.

The PMSM's open loop analysis involves applying input based on the parameters and observing the model verification simulation results. These simulation results demonstrate the accurate representation of the motor's behavior under different load conditions.

3.5.3 Space Vector Pulse Width Modulation (SVPWM)

Space Vector Pulse Width Modulation is an advanced control technique used to drive PMSMs. It's a method that allows for regulation of both the motor's speed and torque. SVPWM improves upon traditional voltage control methods and is known for its efficiency and reduced harmonic distortion in the motor's output. It is a sophisticated control technique used in inverter-based motor drives. It offers superior performance compared to conventional sinusoidal pulse width modulation but involves several intricate steps in its implementation. The steps to implement SVPWM can be outlined as follows:

1. Compute the angle θ and the reference voltage vector $\vec{V}_{\text{ref}}$ utilizing the input voltage components.
2. Calculate the modulation index and assess if it falls within the over-modulation range.
3. Identify the sector wherein $\vec{V}_{\text{ref}}$ resides and ascertain the neighboring space vectors $\vec{V}_1$ and $\vec{V}_2$ based on the sector angle θ.
4. Compute the time intervals T_1, T_2, and T_0 using the sampling time T_s and θ.
5. Determine the modulation durations for the various switching states.

To calculate the reference voltage vector $\vec{V}_{\text{ref}}$, use the following equation:

$$|V_{\text{ref}}| = \sqrt{V_\alpha^2 + V_\beta^2}$$

Where:

$|V_{\text{ref}}|$ is the magnitude of the reference space vector voltage.

V_α is the α component of the reference voltage vector.

V_β is the β component of voltage vector.

The sector in which the reference voltage vector is found can be determined by the angle θ, where θ is in the range $[0°, 60°]$. The modulation index (MI) in this linear region can be expressed as:

$$\text{MI} = \frac{\vec{v}_{\text{ref}}}{v_{\max}}$$

$\vec{v}_{\text{ref}}$ is reference vector. $v_{\max}$ is peak square phase voltage.

The maximum MI is achieved when the reference vector $\vec{v}_{\text{ref}}$ equals the radius of the inscribed circle

$$\vec{v}_{\text{ref}}(\max) = \frac{2}{3} v_{\text{dc}} \cos\left(\frac{\pi}{6}\right)$$

Sector Determination To accurately identify the timing and order of switches, it is crucial to identify the sector reference output is located. There are 6 sectors, each defined by a specific angle range as follows

1. sector 1: $0° < \theta < 60°$
2. sector 2: $60° < \theta < 120°$
3. sector 3: $120° < \theta < 180°$
4. sector 4: $180° < \theta < 240°$
5. sector 5: $240° < \theta < 300°$
6. sector 6: $300° < \theta < 360°$

By determining the sector of voltage vector, you can calculate timing and switching sequence.
Time Duration: Switching time durations are computed for each triangular sector between two state vectors, durations are as follows

$$T_1 = \frac{T_z}{2\sqrt{3}} \left(\frac{V_{\text{ref}}}{V} - \frac{1}{2} \right)$$

$$T_2 = \frac{T_z}{2\sqrt{3}} \left(\frac{V_{\text{ref}}}{V} + \frac{1}{2} \right)$$

$$T_0 = T_z - T_1 - T_2$$

Here, T_z represents the time interval of a single sector.
Switching Time Calculation The method employed approximates the intended stator reference voltages is determined using only eight available states. The switching times for each (S1 to S6) can be determined according to the sector are detailed in Table **??**, distinguishing between upper and lower switches.

Table 3.1: Switching Time Calculation at Each Sector

Sector	Upper switches (S1, S3, S5)	Lower switches (S2, S4, S6)
1	$S1 = T1 + T2 + T0/2$	$S4 = T0/2$
	$S3 = T2 + T0/2$	$S6 = T1 + T0/2$
	$S5 = T0/2$	$S2 = T1 + T2 + T0/2$
2	$S1 = T1 + T0/2$	$S4 = T2 + T0/2$
	$S3 = T1 + T2 + T0/2$	$S1 = T0/2$
	$S5 = T0/2$	$S2 = T1 + T2 + T0/2$
3	$S1 = T0/2$	$S4 = T1 + T2 + T0/2$
	$S3 = T1 + T2 + T0/2$	$S1 = T0/2$
	$S5 = T2 + T0/2$	$S1 = T2 + T0/2$
4	$S1 = T0/2$	$S4 = T1 + T2 + T0/2$
	$S3 = T1 + T0/2$	$S6 = T2 + T0/2$
	$S5 = T1 + T2 + T0/2$	$S2 = T0/2$
5	$S1 = T2 + T0/2$	$S4 = T1 + T0/2$
	$S3 = T0/2$	$S6 = T1 + T2 + T0/2$
	$S5 = T1 + T0/2$	$S2 = T2 + T0/2$
6	$S1 = T1 + T2 + T0/2$	$S4 = T0/2$
	$S3 = T0/2$	$S6 = T1 + T2 + T0/2$
	$S5 = T1 + T0/2$	$S2 = T2 + T0/2$

Chapter 4

Controller Design

Introduction

In this chapter, detailed analysis and design of PI controller for the regulation of both current and speed within the PMSM system is presented. Additionally, analysis and design of MRAS estimator, which incorporates an adjustable model of the PMSM, is provided, with a primary focus on its utility in sensorless speed estimation. The fuzzy logic controller, specifically to fine tune PI controller for the purpose of accurately estimating motor speed and maintaining effective regulation.

4.1 Proportional Integral Controller Design

The Proportional-Integral (PI) controller is used to regulate the speed of the PMSM in the context of the MRAS estimator for sensorless speed control. The controller design involves tuning PI controller gains to achieve the desired system performance. The objective is to minimize the speed error between the reference speed and the estimated speed of the motor. The control signal $u(t)$ for the proportional integral controller is described as follows.

$$u(t) = K_p e(t) + K_i \int_0^t e(t)\, dt \tag{4.1}$$

where: $u(t)$ is the control signal, K_p is the proportional and K_i is the integral gain of the controller, $e(t)$ is the speed error.

Tuning the PI controller gains is crucial to secure stable and efficient motor control. Various tuning methods can be employed, such as trial and error, Ziegler-Nichols method, or advanced optimization approach. In this thesis, fuzzy logic-based approach is used to determine the optimal controller gains. The fuzzy logic controller takes into account factors like motor operating conditions, load torque and inertia to dynamically adjust the gains based on predefined fuzzy rules.

FOC is a widely used technique in motor control because it allows for independent control of the motor's torque and flux, making it possible to control motor's speed and current. In FOC for PMSM, three Proportional Integral (PI) controllers are typically employed:
D-Axis Current Controller: This controller regulates the d-axis current component.
Q-Axis Current Controller: This controller regulates the q-axis current component.
Speed Controller: The third PI controller in the FOC system is responsible for regulating the motor's speed. It adjusts the reference current components provided to d-q axis controllers to achieve desired motor speed.

By implementing these three PI controllers with the MRAS speed estimator and other essential components, the overall block diagram of the complete sensorless speed control scheme for PMSM is illustrated in the figure 4.1.

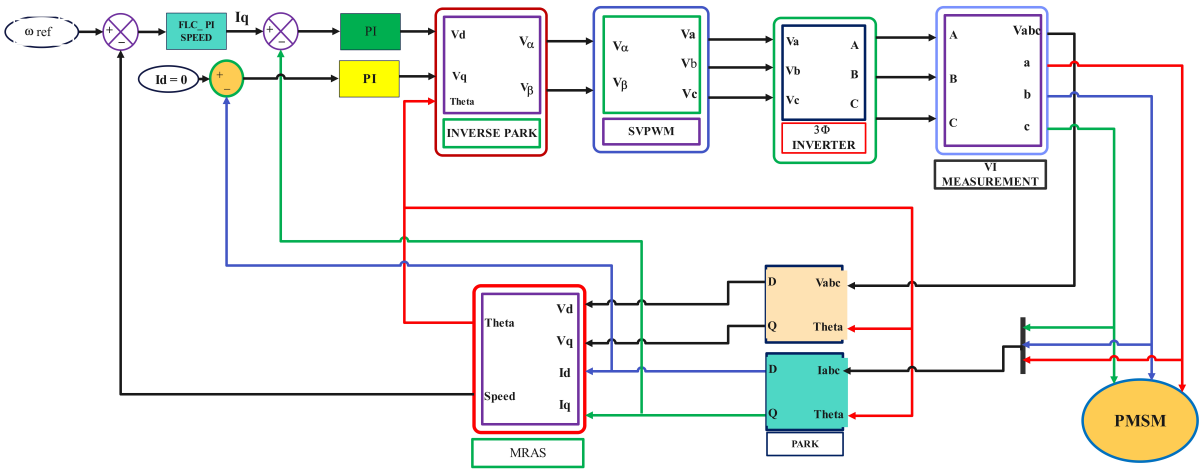

Figure 4.1: Sensorless MRAS based Speed Estimator for PMSM

Here is an overview of PMSM drive control system

PMSM: is the electromechanical component of the system that converts electrical energy into mechanical energy. It consists of a rotor with permanent magnets and a stator with windings.

Power Electronic Converter: is responsible for converting the DC voltage (from a DC source or an inverter) to the required AC voltage for driving the PMSM. It generates the three-phase voltage signals based on the control commands.

MRAS Speed Estimator: The MRAS is a sensorless speed estimation algorithm used estimating the motor's rotor speed and position. It uses a reference model and an adaptive mechanism to adjust its parameters and track the actual speed accurately.

Fuzzy Logic-PI Controller: is regulating the speed and current of the PMSM. It takes speed error and current error (difference between reference currents and actual currents) as inputs and generates the control signals for the power electronic converter.

Speed Reference Generator: The speed reference generator provides desired speed to the control system based on user's input or the system's speed control strategy.

Current Reference Generator: The current reference generator provides the desired current references for the PI current controllers.

Coordinate Transformations: The coordinate transformations (clarke and park transformations) enable the conversion of 3-phase currents (abc) into a more manageable d-q frame.

Current Control Loop: consists of two PI controllers (one for Id and another for Iq). These controllers adjust the voltage references to regulate the motor currents and, consequently, the torque and magnetizing current.

Speed Control Loop: The speed control loop produces control signal for regulating motor's speed by comparing speed reference to the estimated speed obtained from the MRAS.

Space Vector Pulse Width Modulation : The voltage references obtained from the inverse clarke transformation are used to generate the switching signals for the motor's power electronic converter. SVPWM is commonly employed to generate switching patterns to control voltage applied to motor phases.

4.1.1 Design of Current Controller

For an efficient design of PI controllers, comprehending closed-loop transfer function is fundamental. The open loop transfer function of the d-q axis current, as derived from equation (3.9), can be obtained as follows

$$\frac{Id}{V_d + \omega_e LdIq} = \frac{1}{Lds + R} \tag{4.2}$$

The block diagram in figure 4.2 demonstrates the transfer function in a closed-loop system , which incorporates PI controller within the d-axis.

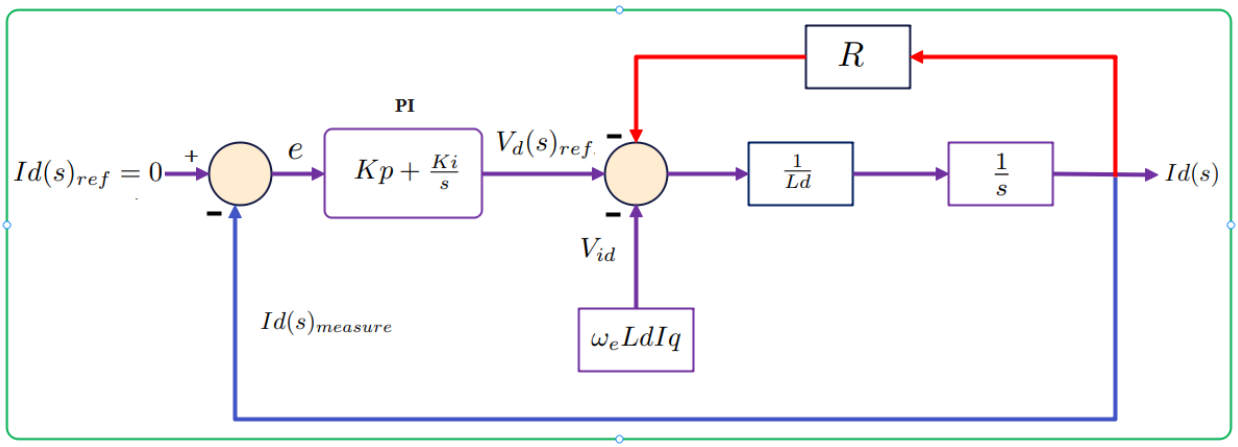

Figure 4.2: Block Diagram of the PI Controller for Flux Generation Component

The presence of the $\omega_e LdIq$ and $\omega_e(LdId + \lambda_f)$ elements introduce complexity to the closed-loop transfer function. For the sake of simplifying the calculation of PI controller parameters, the contributions from the $\omega_e LdIq$ and $\omega_e(LdId + \lambda_f)$ portions of the equation are excluded using a feed-forward decoupling approach. As a result, the complete transfer function governing the relationship between the Id flux component and its reference Id ref from figure 4.2 can be stated as:

$$\frac{Id(s)}{Idref(s)} = \frac{Kps + Ki}{Lds^2 + (Kp + R)s + Ki} \tag{4.3}$$

similarly, the open loop transfer function of the q-axis current can be derived from equation (3.10) in a manner similar d-axis current.

$$\frac{I_q}{V_q - \omega_e(LdId + \lambda_f)} = \frac{1}{Lds + R} \tag{4.4}$$

The closed-loop transfer function for the q-axis including the PI controller is illustrated.

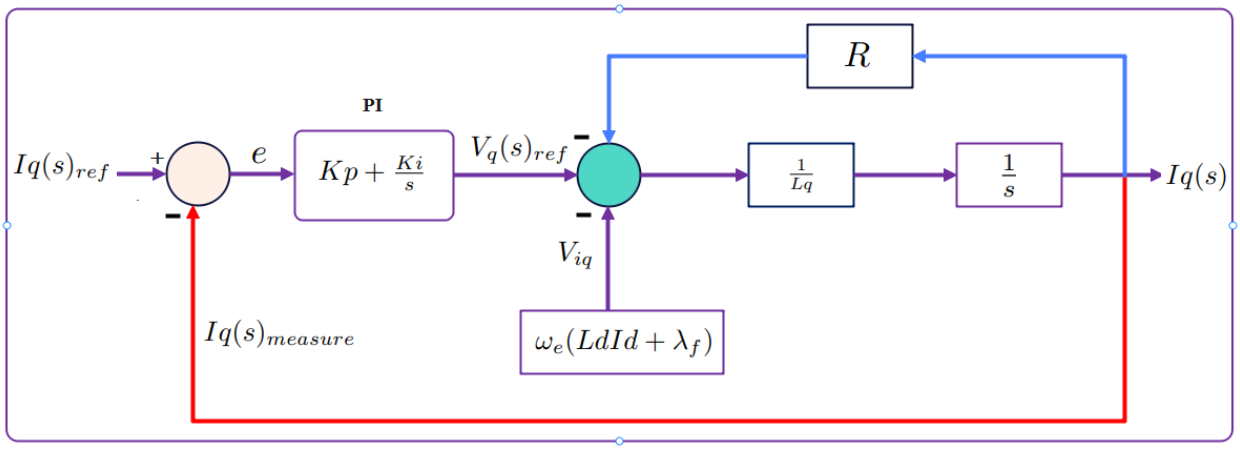

Figure 4.3: Block Diagram of the PI Controller for Torque Generation Component

The torque component in the closed-loop transfer function using a PI controller is the ratio between Iq measurement and Iq reference. Its transfer function can be derived from the above figure.

$$\frac{Iq(s)}{Iqref(s)} = \frac{Kps + Ki}{Lds^2 + (Kp + R)s + Ki} \tag{4.5}$$

Due to the fact that Ld = Lq = Ls for SMPMSM, the closed-loop transfer functions derived from equation (4.3) and equation (4.5) are identical. Hence, designing the q-axis current controller is sufficient to determine the appropriate controller parameters for the d-axis current controller. By comparing the denominator of second order closed-loop systems as presented in equation (4.6) into one of the above equations equation (4.5), the parameters of current controller the gains are determined based on motor parameters and are expressed in equation (4.7) and equation (4.8)

$$s^2 + 2\xi\omega_n s + \omega_n^2 \tag{4.6}$$

$$Kp = 2\xi\omega_n Ld - R \tag{4.7}$$

$$Ki = \omega_n^2 Ld \tag{4.8}$$

Where: ξ is the damping ratio, ω_n is the natural frequency.

4.1.2 Controller Design Specification

In the design process, certain parameters are taken into consideration. Particularly, the damping coefficient (ξ) is intentionally set to 0.707 to achieve a critically damped response. Simultaneously, the selection of the natural frequency (ω_n) determines the desired settling time of closed loop, aligning it with the intended bandwidth of the system. A greater ω_n corresponds to a shorter desired settling time for the closed loop. Opting for this parameter in reference to the open loop system's bandwidth, calculated as $\omega_n = \frac{Ld}{R}$, presents advantages. To aid in this selection, normalized parameter denoted as $0 < \gamma < 1$ is introduced. It is utilized to calculate the desired closed-loop bandwidth as follows [37]:

$$\omega_n = \frac{1}{1-\gamma} \cdot \frac{Ld}{R} \tag{4.9}$$

In many cases, a typical value of γ around 0.8 is chosen to achieve satisfactory performance. The natural frequency of the current loop is set at $\omega_{ni} = 560\,\text{rad/sec}$. Consequently, inner loop needs to operate significantly faster than outer loop, typically by a factor of 5 to 10, to ensure optimal performance. Thus, the natural frequency for the speed controller of the outer loop is determined as $\omega_{ns} = \frac{\omega_{ni}}{5} = 110\,\text{rad/sec}$. The current controller's gains can be calculated utilizing Equations 4.7 and 4.8. The motor parameters used in these calculations are $R = 2.875\,\Omega$ and $L = 0.0085\,\text{H}$ from Appendix B. This results in $Kp = 3.945$ and $Ki = 2665.6$.

4.1.3 Design of Speed Controller

The speed controller tracks commanded speed, using motor electro-mechanical properties as described by the following equations

$$Iq(t) = \frac{3}{2}\frac{p\lambda_f}{2J}\frac{d\omega_e(t)}{dt} + B\omega_e(t) + PTL(t) \tag{4.10}$$

From equation (4.10) differential equation can be derived as

$$\frac{d\omega_e(t)}{dt} = \frac{3}{2}\frac{P\lambda_f}{2J}Iq(t) - \frac{B}{J}\omega_e(t) - \frac{P}{J}TL(t) \tag{4.11}$$

Where: λ_f is Magnetic flux, P is pole pairs, J is inertia (kg/m^2), B is Viscous friction coefficient, ω_e is electrical speed of the rotor.

Applying the laplace transformation, we can establish the relationship between angular velocity and the q-axis current

$$\left(s + \frac{B}{J}\right)\omega_e(s) = \frac{3}{2}\frac{p\lambda_f}{2J}Iq(s) \tag{4.12}$$

$$\frac{\omega_e(s)}{Iq(s)} = \frac{\frac{3}{2}\frac{P\lambda_f}{2J}}{s + \frac{B}{J}} \tag{4.13}$$

From the current Proportional-Integral (PI) controller, the transfer function from the q-axis reference current to the actual q-axis current is given by:

$$\frac{Iq(s)}{Iq(s)^*} = \frac{2\xi\omega_n - \frac{R}{L}}{s^2 + 2\xi\omega_n s + \omega_n^2} \tag{4.14}$$

By substituting equation 4.14 into 4.13, we can establish the relationship between the referenced q-axis current $Iq(s)^*$ and the speed $\omega_e(s)$.

$$\frac{\omega_e(s)}{Iq(s)^*} = \frac{\frac{3}{2}\frac{P\lambda_f}{2J}\left(2\xi\omega_n - \frac{R}{L}\right)}{\left(s + \frac{B}{J}\right)\left(s^2 + 2\xi\omega_n s + \omega_n^2\right)} \tag{4.15}$$

When the chosen natural frequency ω_n significantly exceeds the mechanical factor B/J, the inner current loop dynamics can be neglected. This simplifies the model to a first-order approximation

$$\frac{\omega_e(s)}{Iq(s)^*} \approx \frac{\frac{3}{2}\frac{P\lambda_f}{2J}}{s + \frac{B}{J}} \tag{4.16}$$

The parameters for the PI current controller can be computed in the following manner

$$Kp = \frac{2\xi\omega_n - \frac{B}{J}}{3\frac{p^2\lambda_f}{2J}} \tag{4.17}$$

$$Ki = \frac{\xi\omega_n^2}{3\frac{P^2\lambda_f}{2J}} \tag{4.18}$$

It is recommended to set the outer loop bandwidth between 5% to 10% of the inner-loop bandwidth. consequently, with $\omega_n = 110$ rad/sec and $\xi = 0.707$, the controller gains can be calculated as $Kp = 0.05$ and $Ki = 2$

4.2 Model Reference Adaptive System

MRAS is a popular technique Utilized to estimate the speed of the rotor in the PMSM. It utilizes a reference and adjustable models to estimate the rotor speed enabling sensorless control [38]. The MRAS uses two independent machine models with different structures to estimate the same state variable (rotor speed of motor). The purpose of using two models is to provide a reference for the estimation process and to enable adaptation to varying the operating conditions.

The reference model replicates the dynamic behavior of the PMSM and is designed based on its mathematical model. It represents the expected relationship between the stator currents, rotor speed, and other motor parameters. The reference model's output provides the desired rotor speed estimate.

The adjustable model on the other hand, is designed to track the actual behavior of the motor. It estimates the rotor speed by comparing measured stator currents with reference model's predicted currents. By adjusting the parameters of adjustable model, it aims to minimize the difference between the actual and predicted stator currents.

Adaption mechanism The estimation process in MRAS involves comparing the outputs of the reference and adjustable models and adjusting the parameters to reduce the difference between the two outputs. This adjustment is typically done using adaptive algorithms that update the adjustable model's parameters in real-time based on the estimation error. [39], [40]. The reference model determines the state variable independently, without relying on the rotor speed and position. On the other hand, the adjustable model calculates the same state variable by incorporating rotor and position value provided by the adaptive mechanism. The advantage of using MRAS for speed estimation is its adaptability to variations in motor parameters and operating conditions. The adjustable model continuously adjusts its parameters to track changes in the motor's characteristics, enhancing speed estimation. In the MRAS estimator, the stator current and voltage measurements are used to determine rotor flux. By comparing the estimated flux with the measured rotor flux, an error signal is generated. This error signal is then passed through a proportional-integral controller to generate a control signal, which in turn adjusts the parameters of the adjustable model. Through this parameter adjustment, the estimated speed is fine-tuned to closely align with the actual rotor speed.

4.2.1 Model Reference Adaptive System Estimation Model

The fundamental concept of model reference adaptive system use two models to calculate the identical state variable and utilizing the difference between their outputs to estimate the desired variable. When estimating the rotor speed of the PMSM, one common critique of model reference adaptive system is its sensitivity to variations in motor parameters [41], [42]. The accuracy of these parameters is crucial as the models primarily rely on mathematical equations representing the motor. This issue can be mitigated to some extent by using the motor itself as a reference model. In this thesis, the PMSM itself is used as the reference model, the MRAS speed estimation technique simplifies the design and implementation process. It eliminates the need for additional models or estimators, making the system more robust and efficient. The model reference adaptive system scheme is make clear show in figure 4.4 .

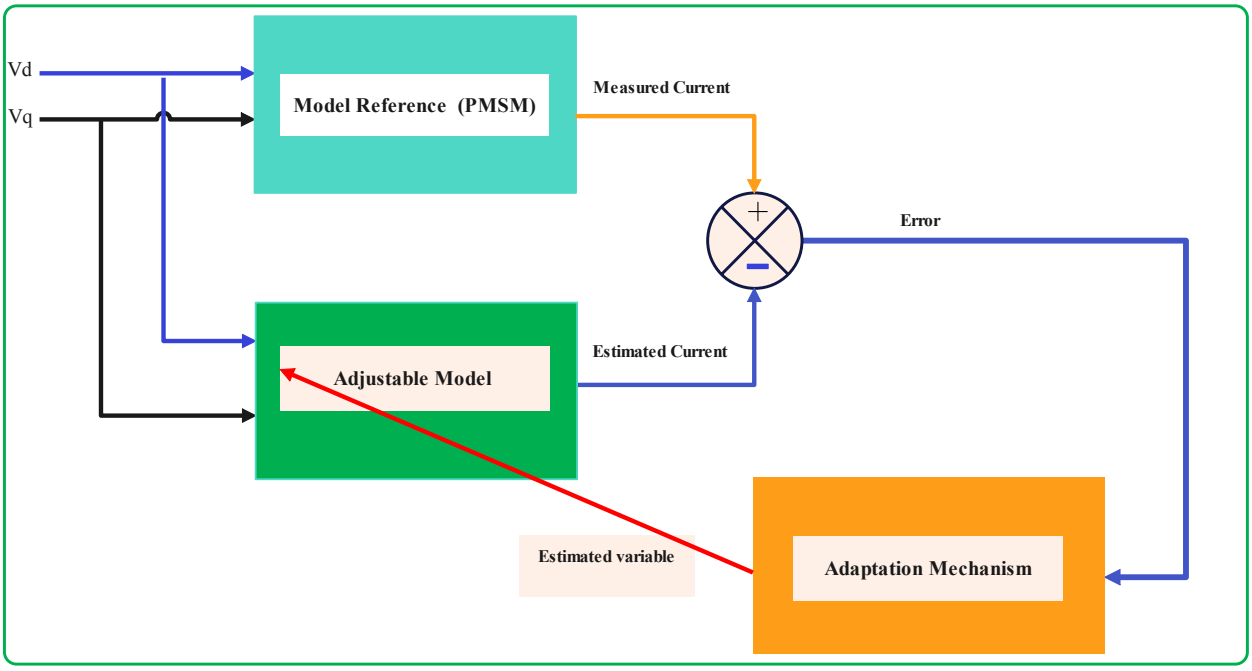

Figure 4.4: Model Reference Adaptive System

The model reference adaptive system speed estimation process involves the following steps:

1. Generate the reference model based on the mathematical model of the PMSM, considering the motor's electrical characteristics and mechanical dynamics.

2. Take measurements of the stator currents and utilize them as inputs to both the reference and adjustable models.

3. Calculate the predicted currents using model reference and adjustable model.

4. Compare the measured currents with the predicted currents.

5. Refine adjustable model parameters to minimize the gap between measured and predicted currents.

6. Utilize the predicted rotor speed derived from the adjustable model as the speed estimation for sensorless control.

4.2.2 Adjustable Model for PMSM

The adjustable model for the PMSM incorporates the mathematical equations that Detail the motor's performance within the d-q reference frame. The model takes into account the stator voltages, currents, and rotor flux linkages to estimate the rotor speed. The adjustable model represents the motor electrical equations in the dq-frame from equation (3.9) the adjustable model consists of the following equations

$$\begin{cases} \dot{\hat{x}} = \hat{A}\hat{x} + \hat{B} \\ y = C\hat{x} \end{cases} \tag{4.19}$$

$$\hat{A} = \begin{bmatrix} -\frac{R_s}{Ld} & \hat{\omega_e} \\ -\hat{\omega_e} & -\frac{R_s}{Ld} \end{bmatrix} ; \hat{B}u = \begin{bmatrix} \frac{V_d}{Ld} \\ \frac{1}{Ld}[V_d - \lambda_f \hat{\omega_e}] \end{bmatrix} ; C = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

In these equations V_d, V_q represent voltages, Ld and Lq represent inductance, R_s represents resistance and λ_f the rotor flux linkage in the dq-frame. The state vector $\hat{x} = [\hat{id} \quad \hat{iq}]^T$ consists of estimated speed $\hat{\omega_e}$ and currents $[\hat{id}, \hat{iq}]$. The PI controller serves as an adaptive mechanism in the MRAS speed estimation of motors. It adjusts the parameters of the adjustable model based on the estimation error, combining proportional and integral control actions.

The PI controller enhances speed estimation accuracy, reduces steady-state error, and ensures robust operation of the PMSM system under varying operating conditions. The measured state vector $x = [id \quad iq]^T$ and the error $\zeta = \hat{x} - x$ are examined with the Popov hyper-stability theorem [43]. The detailed description of the Popov hyper-stability theorem can be found in Appendix A.

The transfer function of the adaptive PI mechanism can be obtained by deriving as follows [44].

$$\hat{\omega_e} = (K_p^M + \frac{K_i^M)}{S})[(\hat{iq} - iq)(id + \frac{\lambda_f}{Ld}) - (\hat{id} - id)iq] + \hat{\omega_e}(0) \tag{4.20}$$

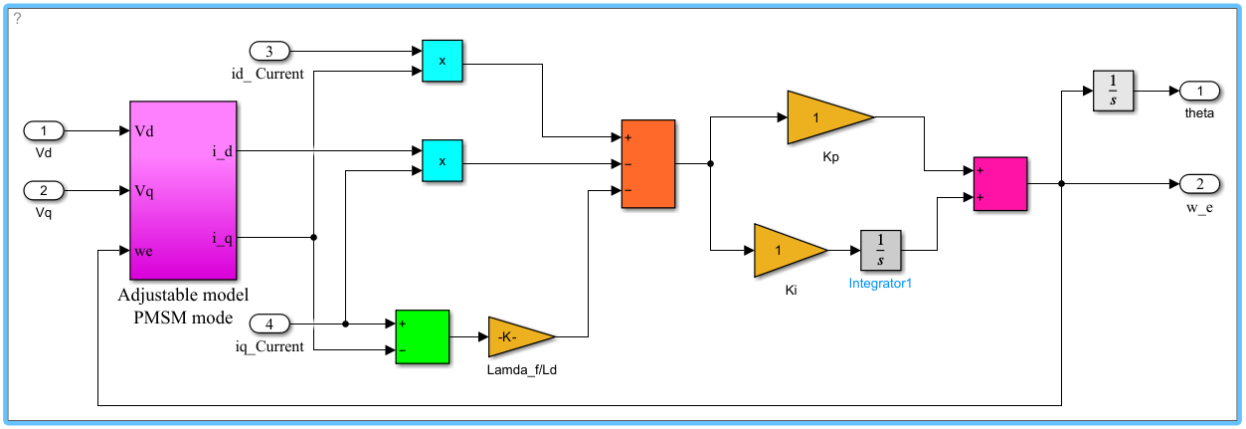

Figure 4.5: Model Reference Adaptive System Simulink Block Diagram

4.3 Motivations for Choosing Fuzzy Logic Controller

There are several motivations for choosing a fuzzy logic controller in various applications:

Handling Uncertainty: FLCs are designed to handle imprecise and uncertain information. In real-world systems, there are often uncertainties and ambiguities that cannot be easily modeled or represented by precise mathematical equations. FLCs provide a framework to incorporate this uncertainty and make decisions based on fuzzy rules and linguistic variables.

Nonlinearity: Many systems exhibit nonlinear behavior, which can be challenging to model accurately using conventional control techniques. FLCs are well suited for dealing with nonlinear systems since they do not require explicit mathematical models. They can capture and represent complex nonlinear relationships through fuzzy rules and membership functions.

Expert Knowledge: FLCs allow the integration of expert knowledge into the control system. Experts can provide linguistic rules and heuristics based on their experience, which can be difficult to formalize using traditional control methods. FLCs enable the translation of this expert knowledge into a control strategy that can be applied in real-time.

Robustness: FLCs tend to exhibit good robustness against disturbances and modeling uncertainties. The fuzzy rule based structure allows FLCs to handle uncertainties and variations in the system dynamics. This robustness makes FLCs suitable for systems where precise modeling is challenging or where the system parameters may change over time.

4.4 Fuzzy Logic Controller (FLC)

A Fuzzy Logic Controller (FLC) is a control system that utilizes fuzzy logic principles to make decisions and control actions. It is a type of intelligent control system that can handle uncertainties and imprecisions in system modeling and control, making it particularly useful in complex and nonlinear systems [10].

The FLCs are designed based on the concepts of fuzzy set and fuzzy logic, which allow for the representation of imprecise or vague information. Fuzzy logic enables the use of linguistic variables and fuzzy sets to describe system behavior and control actions in a more intuitive and human-understandable manner. The diagram illustrating structure of fuzzy logic controller is presented in figure 4.6.

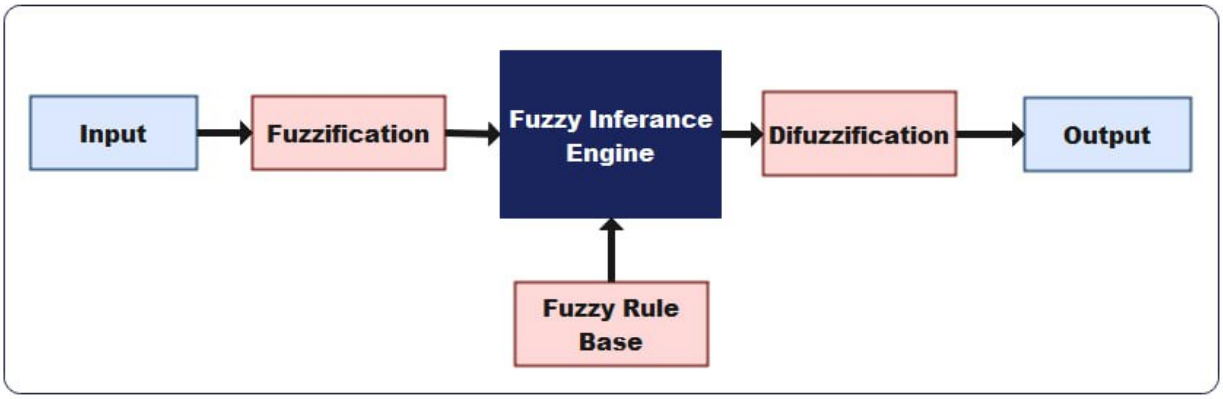

Figure 4.6: Fuzzy Logic Controller Structure

The general outline of the design process for the fuzzy logic controller is described as.

Inputs and Outputs: The key inputs to FLC could include error, error change and generate the output control signals, which will be used to adjust the PI controller's parameters. The universe of discourse for the error variable establishes the extent or scope within which the error can exhibit variation [45]. If the desired motor speed range is from 0 to 100 RPM, the universe of discourse for the error variable (e) could be Minimum value -100 RPM indicating the motor is too slow due to large negative error and Maximum value 100 RPM indicating the motor is too fast due to large positive error.

Membership Functions: Membership functions are used to define the linguistic variables and their associated degrees of membership in fuzzy sets. Membership functions determine the degree of truth for each input and output variable, representing the fuzzy logic controller's linguistic terms. The selection and design of membership functions should capture the nonlinear relationships between the inputs and outputs and effectively capture the system dynamics. Membership functions for the input variables, such as motor speed error & change of error, can be Described using triangular, trapezoidal, or gaussian functions. In this thesis use triangular, the shape and parameters of the membership functions are chosen based on the system characteristics and control requirements. Similarly, membership functions for the output variables, representing the adjustments to the PI controller gains are defined to appropriately capture the range of adjustments needed.

Fuzzy Rules: Fuzzy rules determine how the inputs are mapped to the outputs. These rules are typically written in an if-then format [46]. The fuzzy logic rules should be carefully crafted based on domain knowledge, experience and system characteristics. The rule base should cover a wide range of operating conditions and provide adequate the control action.

Fuzzy Inference System: The fuzzy logic rules and membership functions to determine the appropriate adjustments to the PI controller gains. The fuzzy inference system performs a process called fuzzy logic reasoning, which involves fuzzification, rule evaluation, aggregation and defuzzification. Fuzzification involves mapping the crisp input values (motor speed error, rate of change of error) into their corresponding fuzzy sets using the defined membership functions. Rule evaluation applies the fuzzy logic rules to determine the appropriate adjustments to the gains based on the fuzzy sets' degrees of truth. Aggregation combines the output adjustments from multiple rules to obtain a comprehensive adjustment for each gain. Finally, defuzzification converts the fuzzy output adjustments into crisp values that can be directly applied to the PI controller. Various defuzzification methods can be employed, such as centroid based methods like center of gravity, height based methods or the use of weighted average techniques. When it comes to the choice of defuzzification method, there is no standardized approach for its selection. In this thesis, the choice has been made to utilize the center of gravity method for defuzzification.

4.5 Fuzzy Logic Controller Design

Fuzzy logic controller is a control system that uses fuzzy logic principles to make decisions and control a system behavior of PMSM. It is a rule based control methodology that mimics human decision making processes by incorporating linguistic variables and fuzzy rules. In the context of controlling permanent magnet synchronous motor with unpredictable parameter variations, the PI controllers may not always yield optimal performance. To address this challenge, a promising approach is to incorporate FLC to retune PI controller gains [47].

The beauty of using fuzzy logic controller lies in its adaptability and ability to dynamically adjust the controller gains based on the current system dynamics. By employing linguistic variables, fuzzy logic rules and membership functions, the fuzzy logic controller can intelligently fine-tune the PI controller gains [4]. This adaptability allows the system to optimize its response, improve stability and enhance robustness against varying parameters. By considering crucial system parameters such as speed, load torque and other relevant variables, the fuzzy logic controller can effectively determine the appropriate gains for the PI controller. The linguistic variables provide a descriptive language to characterize the system's behavior, while the fuzzy logic rules guide the adjustments of the gains.

By combining the strengths of PI controller and adaptability of FLC, the control system becomes an alluring solution for nonlinear systems (permanent magnet synchronous motor) with unpredictable changes. It not only delivers superior performance but also exhibits resilience and efficiency in the face of evolving operating conditions. In this thesis tuning PI controller that adjust kp and ki parameter gains based on the speed error (e(t)) and the rate of change of speed error (ce(t)) and the output signal u(t) corresponds to the variation in quadrature reference current (I*q)

$$e(t) = \omega_e^*(t) - \omega_e(t) \tag{4.21}$$

$$ce(t) = \frac{d}{dt}(e(t)) \tag{4.22}$$

where ω_e^* (t) is reference & $\omega_e(t)$ actual. After self-tuning, PI controller can be defined :

$$u(t) = Kp_{\text{new}} \cdot e(t) + Ki_{\text{new}} \cdot \int e(t)dt + Kp_{\text{fuzzy}} \cdot e_{\text{fuzzy}}(t) + Ki_{\text{fuzzy}} \cdot \int e_{\text{fuzzy}}(t)dt$$

where:

- $u(t)$ is control signal.
- Kp_{new} and Ki_{new} are gains obtained after self-tuning.
- Kp_{fuzzy} and Ki_{fuzzy} are the gains obtained from the fuzzy control that vary online.
- $e(t)$ is the speed error.
- $\int e(t)dt$ is integral error.
- $e_{\text{fuzzy}}(t)$ is the fuzzy error, which is calculated based on the fuzzy logic control.
- $\int e_{\text{fuzzy}}(t)dt$ is integral of fuzzy error.

The overall structure of tuning fuzzy PI controller for speed control of PMSM illustrates figure 4.7.

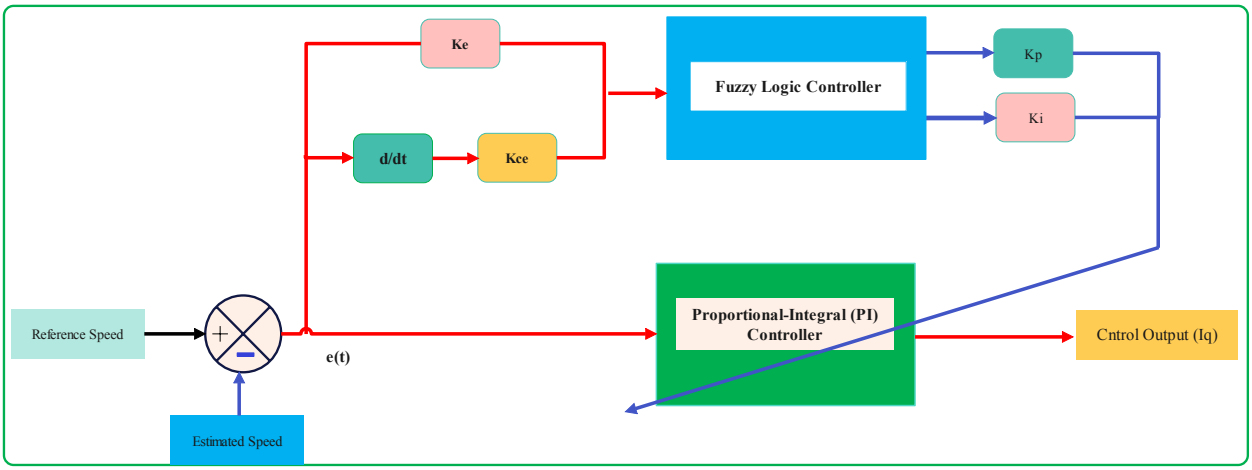

Figure 4.7: Fuzzy PI Controller Structure

The parameters ke, kce, kp, and ki in figure 4.7 are referred to as scaling factors. These scaling factors play a crucial role in achieving normalized input-output for FLC. Their purpose is to ensure that the variables used within the controller are scaled appropriately. The scaling factor should satisfy the following criteria, normalization the scaling factor should normalize the input and output variables to a common range. This ensures that all variables have a similar scale, which helps in comparing and combining their values within the fuzzy logic controller. Membership function range the scaling factor should be chosen such that it covers the entire range of the membership functions defined for each input and output variable [46]. This allows the fuzzy logic controller to effectively capture the linguistic terms and their corresponding degrees of membership. Control range the scaling factor should be selected

based on the desired control range of system. It should map the normalized inputs and outputs to appropriate control range, ensuring that the controller can achieve the desired control objectives. Accuracy the scaling factor should be chosen carefully to maintain the accuracy of the fuzzy logic controller [6]. It should not introduce significant rounding errors or loss of precision during the scaling process. By fulfilling these requirements, the scaling factor in the tuning fuzzy PI controller can effectively normalize the variables, maintain accuracy, achieve the desired control range.

4.5.1 The Fuzzy Language of Input and Output Variables

The fuzzy language of input and output variables refers to the linguistic terms or labels used to describe different levels or states of the variables in a fuzzy logic system [48]. These linguistic terms are typically represented by fuzzy sets, which assign membership degrees to different values or ranges of the variables. In the thesis, the inputs (e) and (ce) are normalized to a range of -1 to 1 for the universe of discourse. Similarly, the membership functions

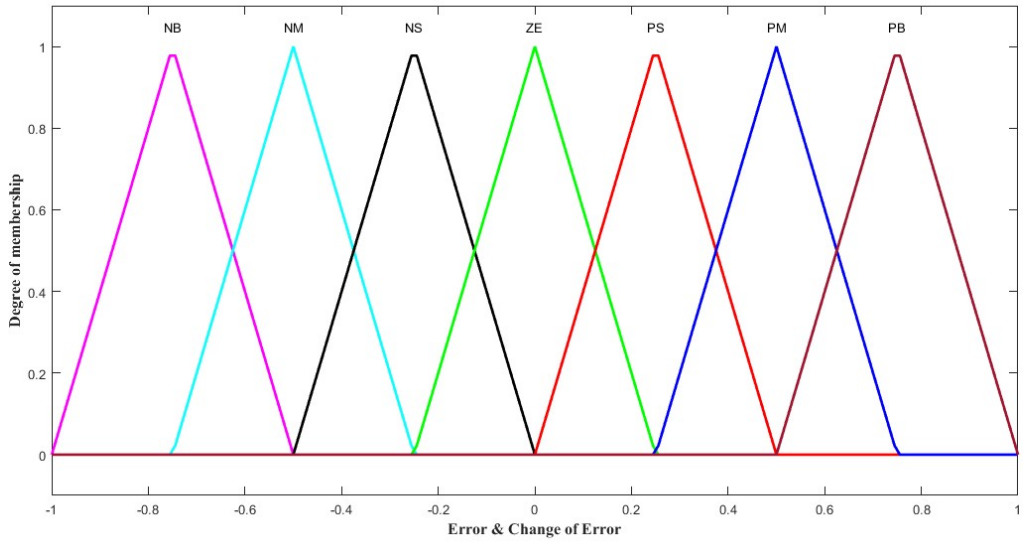

Figure 4.8: Membership Function for the Input Variables e and ce

also have a range of -1 to 1 for their universe of discourse and linguistic labels are assigned to for both input error (e) and rate of change in error (ce).

Input Variable (e) and (ce): Negative Big (NB), Negative Medium (NM), Negative Small (NS), Zero (ZE), Positive Small (PS), Positive Medium (PM), Positive Big (PB) and the rules bases represent NB, NM, NS, ZE, PS, PM, PB [10] are mentioned in the figure 4.8.

Output Variables Kp and ki: have linguistic labels ranging from 0 to 1, which are denoted as ZE, MS, S, M, B, MB, VB in rule bases figure 4.9 shows the membership functions of output fuzzy sets.

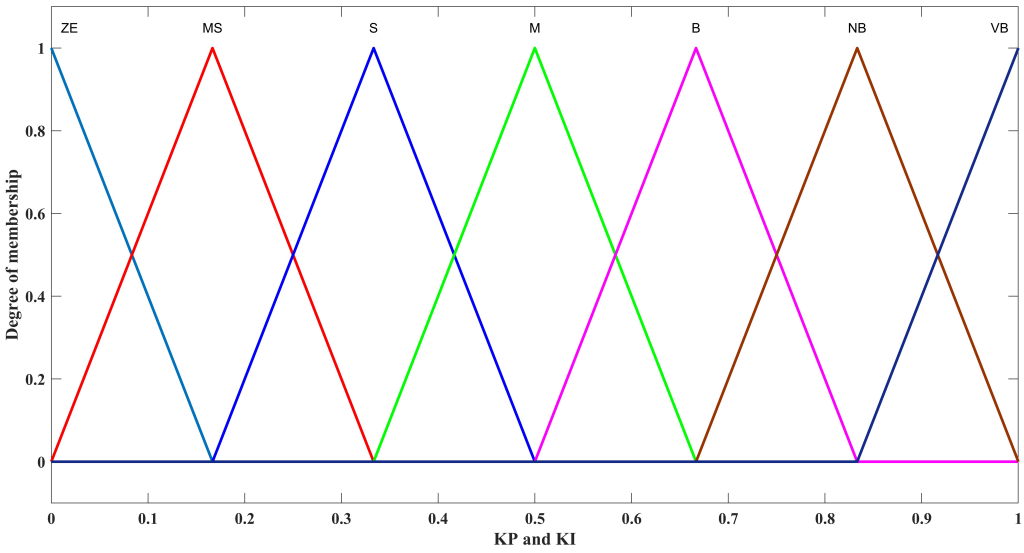

Figure 4.9: Membership Function for the Output Variables Kp and Ki

The fuzzy language plays a crucial role in mapping crisp input values to fuzzy sets and determining control actions based on inference rule. In fuzzy structure described, reasoning of outputs is obtained by aggregating inputs and applying fuzzy rules. The aggregation method is the max-min, while the defuzzification method employed is the centroid method. The fuzzy inference system has two inputs error and derivative of error and two outputs denoted as Kp1 and Ki1, the Mamdani is utilized fuzzy inference with specific modifications made to optimize the values as shown in figure 4.10. Triangular membership functions have been selected for their ability to offer both optimal control performance and simplicity. In this system, seven levels of fuzzy membership functions have been applied to all variables. Table **??** provides the 7 by 7 rule base table employed within the system.

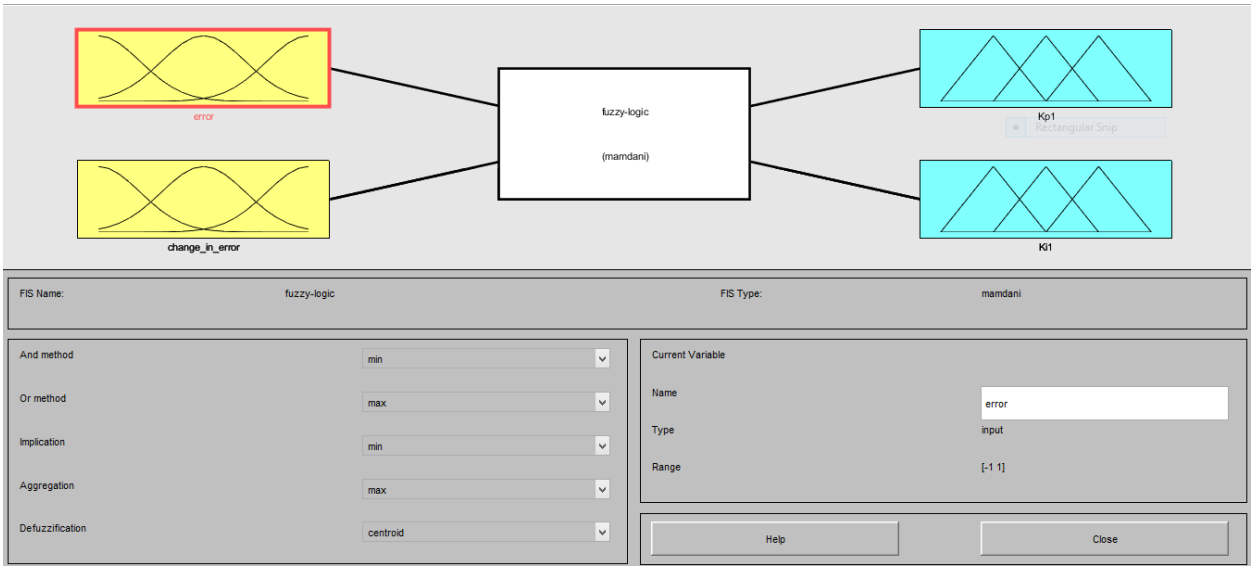

Figure 4.10: The Component Responsible for Performing Fuzzy Inference.

$e/\dot{e}$	NB	NM	NS	ZE	PS	PM	PB
NB	VB	VB	MB	MB	B	B	M
NM	PB	MB	MB	B	B	M	S
NS	MB	MB	B	B	M	S	S
ZE	MB	B	B	M	S	S	MS
PS	B	B	M	S	S	MS	MS
PM	B	M	S	S	MS	MS	ZE
PB	M	S	S	MS	MS	ZE	ZE

Table 4.1: The Set of Rules Used to Determine the values of Kp1 and Ki1

Chapter 5

Simulation Results and Discussion

Introduction

In this chapter, simulation results for PMSM speed control are presented and discussed. The evaluation includes both the proportional integral and fuzzy logic controller under conditions with and without load. The analysis covers actual and estimated speed, electromagnetic torque, three phase stator current, estimated d-q current, variable speed and forward reverse operation in various scenarios. The study involves the assessment of the performance and robustness of PI and Fuzzy Logic (FL-PI) controller, encompassing step response performance analysis, controller robustness analysis, and the impact of disturbances. The controllers' robustness against external disturbances, variable speeds, torque fluctuations and parameter variations. Additionally, the report evaluates the impact of load variations on motor performance and compares the performance of PI and (FL-PI) controllers using performance indexes of controllers. Simulation results demonstrate precise speed regulation with minimal steady state error, achieved by the MRAS based speed estimator with the (FL-PI) controller, highlighting its resilience against disturbances.

5.1 Simulation Results of Speed Control for PMSM with PI Controller

5.1.1 Dynamic Simulation Results of PMSM With Load

5.1.1.1 Actual and Estimated Speed

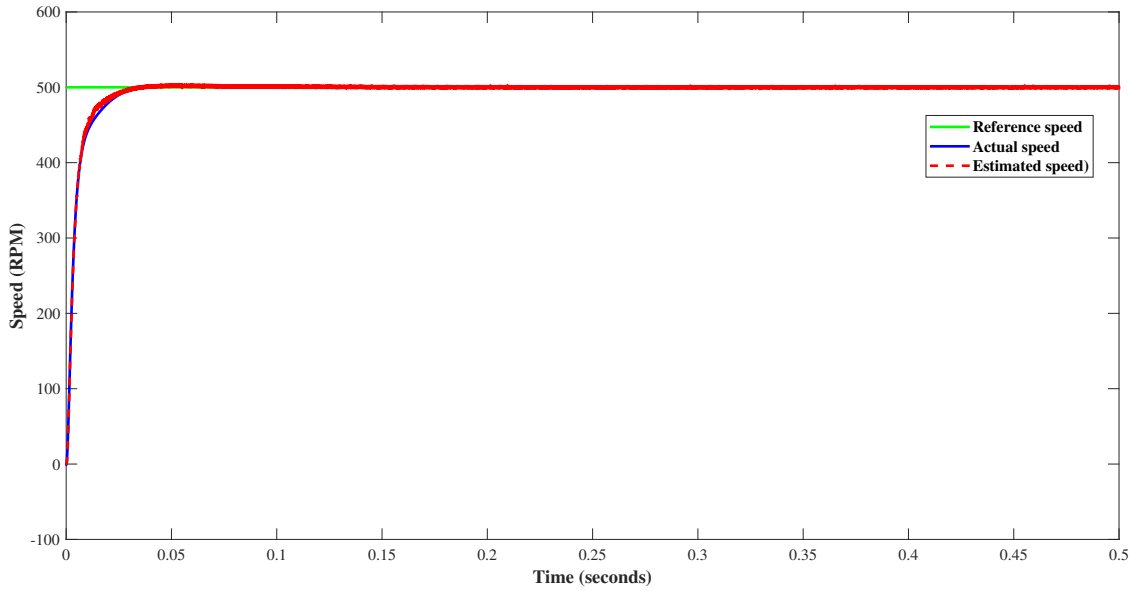

Figure 5.1: Speed Response of PMSM at 500 RPM Under Loaded Condition (1.8 Nm)

The simulation result conducted using a step speed input with a load torque of 1.8 Nm, as shown in figure 5.1, to evaluate the motor's speed response when commanded to operate at reference speeds of 500 RPM. Initially, the speed response with the PI controller exhibit overshoot of 3.6% and a steady-state error of 0.1 for the provided references.

The simulation results demonstrate that the proposed estimator successfully tracks reference step inputs at 500 RPM and 1000 RPM. In figure 5.2, the speed response is shown when the motor is commanded to operate at a reference speed of 1000 RPM with a load torque of 1.8 Nm. The PI controller exhibits an overshoot of 0.2% and zero steady-state error for the provided reference.

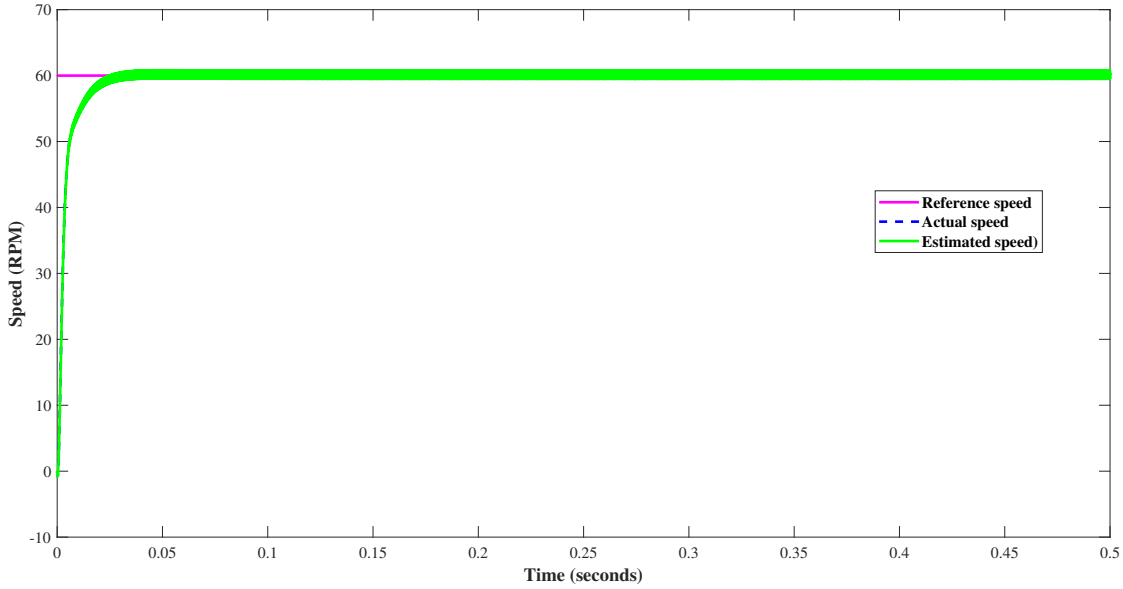

Figure 5.2: Speed Response of PMSM at 60 RPM Under Loaded Condition

5.1.1.2 Three Phase Stator Current and Estimated d-q Current

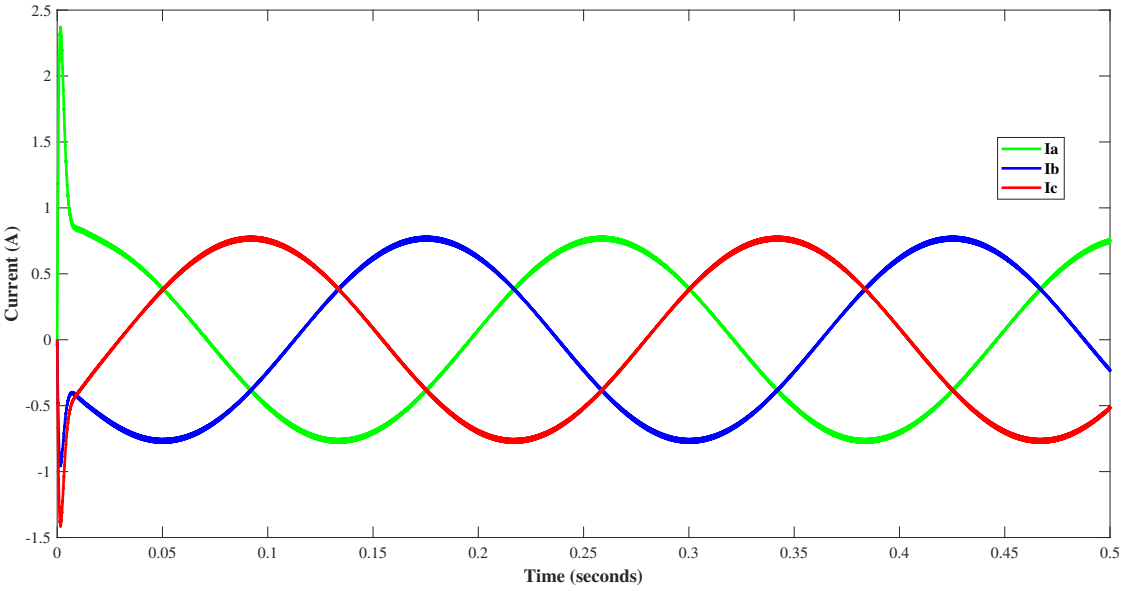

Figure 5.3: Three-Phase Stator Current in Loaded Condition (1.8 Nm)

The simulation result in figure 5.3 show the behavior of stator current in a motor under at 0.8 Nm load torque at a constant 60 RPM. Initially non-sinusoidal during startup, the stator current gradually becomes sinusoidal in the steady state, indicating stable motor operation.

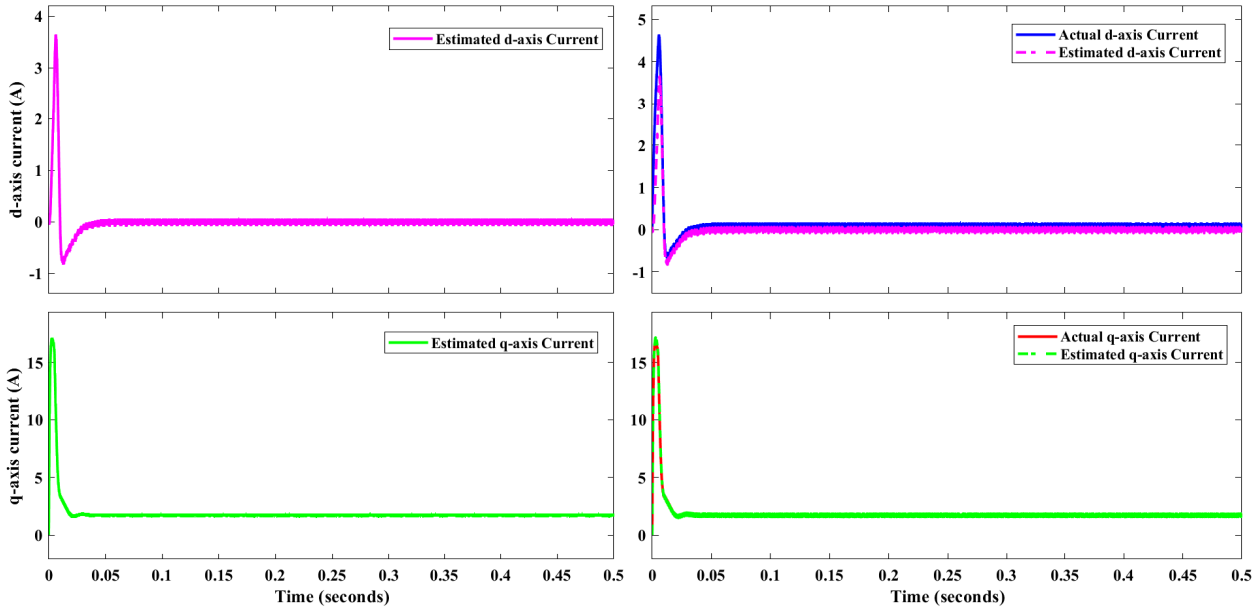

Figure 5.4: The Actual and Estimated d-q Current in Loaded Condition (1.8 Nm)

The figure 5.4 shows that Id (d-axis current) remains at zero throughout the simulation. Additionally, the Iq (q-axis current) demonstrates its changes over time as the motor responds to load conditions. This illustrates how Iq adjusts to accommodate the load torque.

5.1.1.3 Electromagnetic Torque

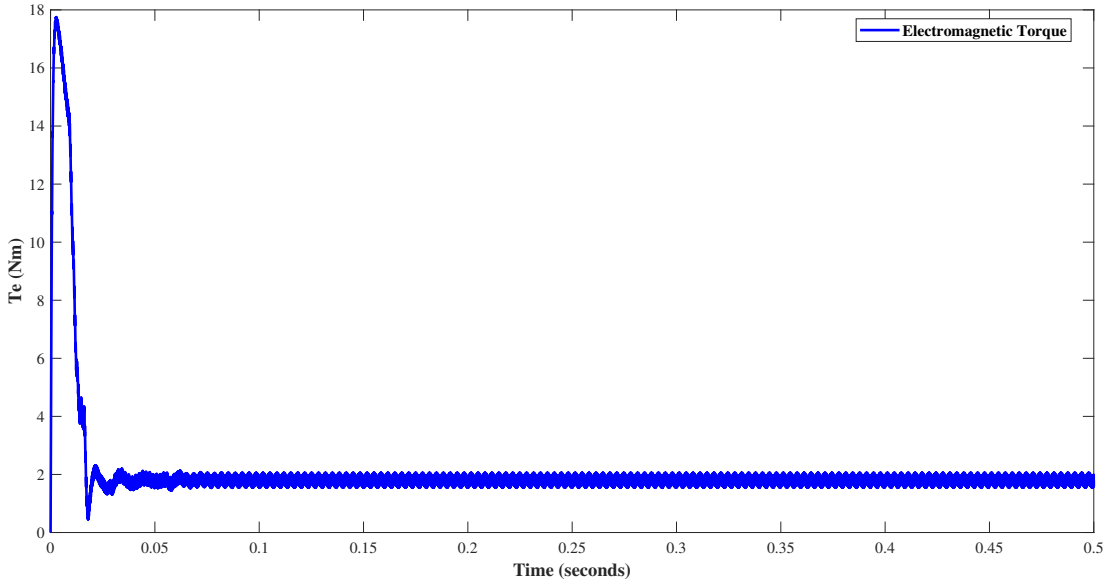

Figure 5.5: Electromagnetic Torque in Loaded Condition at 1.8 Nm

Under load, the electromagnetic torque must exceed or equal the 1.8 Nm load to sustain motor speed and perform necessary mechanical work.

5.1.2 Dynamic Simulation Results of PMSM Without Load

5.1.2.1 Actual and Estimated Speed

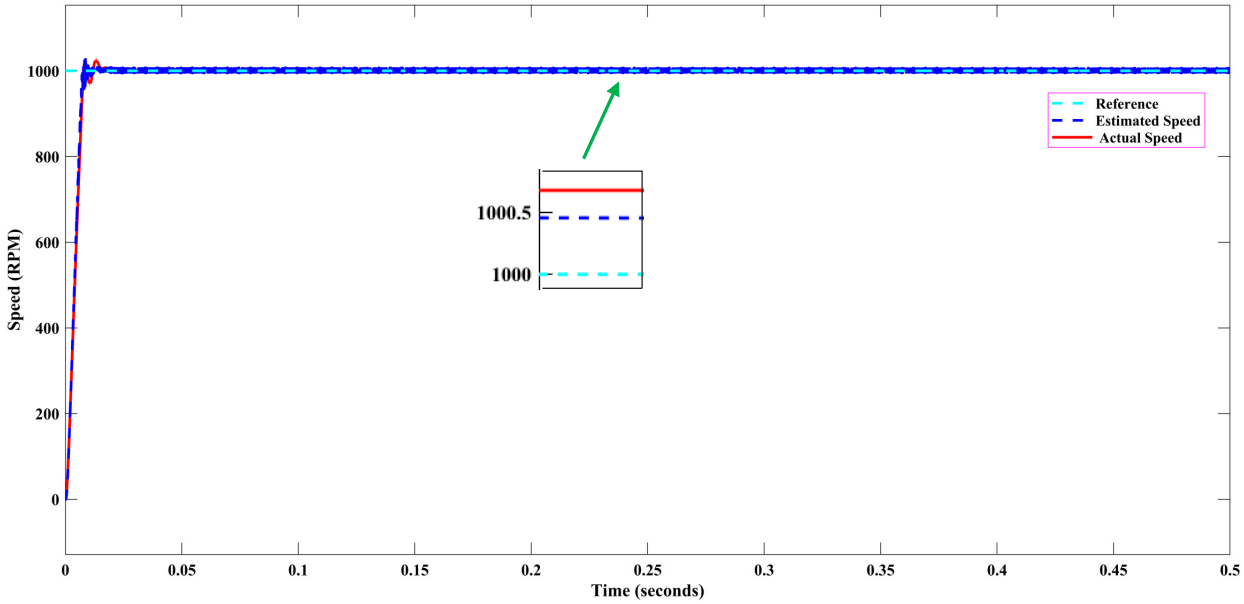

Figure 5.6: Actual and Estimated Speed in No-Load Condition

The simulation results under no load are depicted in figure 5.6. The speed response with the PI controller shows a 2.6% overshoot.

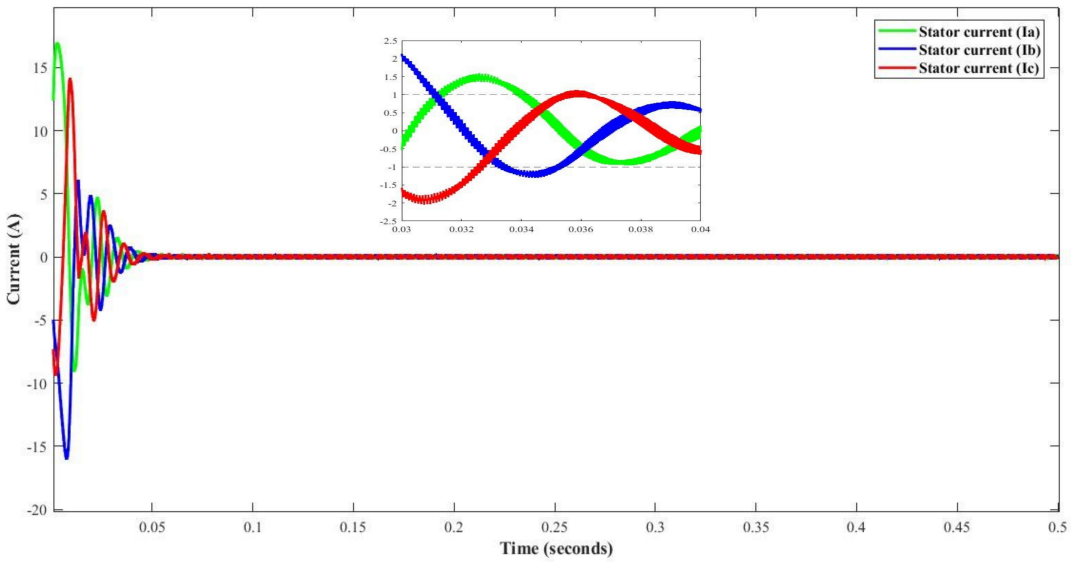

Figure 5.7: Three Phase Stator Current in Unloaded Condition

As shown in figure 5.7, in the no-load condition, the motor draw a small amount of current

to overcome its own internal losses (friction, iron losses, etc.) and is significantly lower than the current drawn under load.

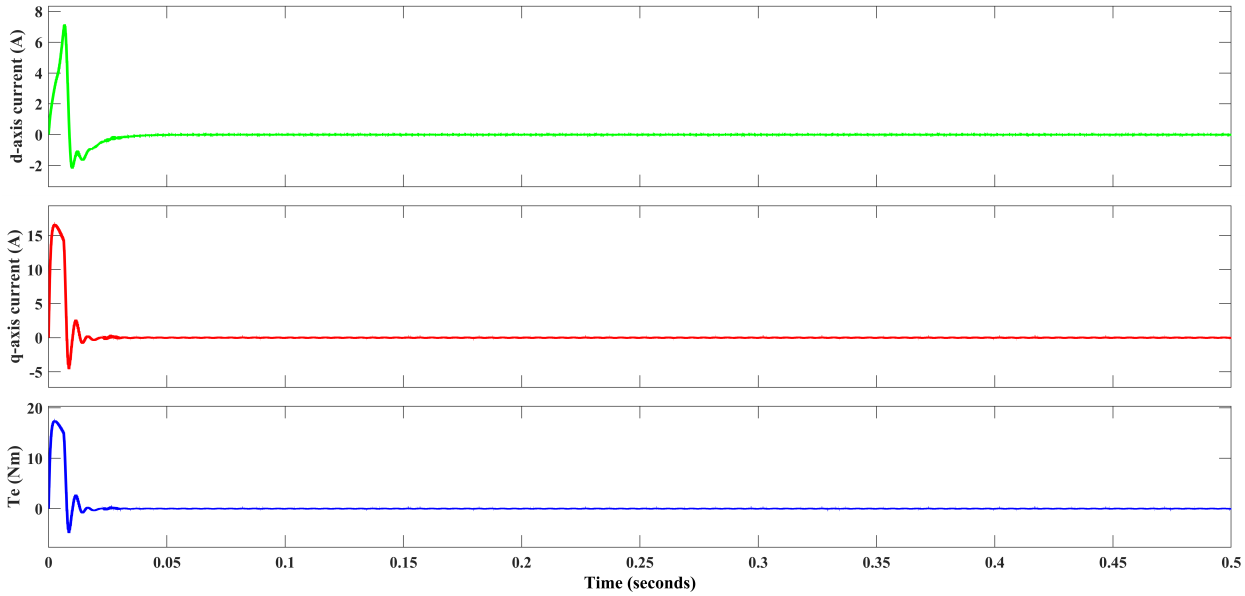

Figure 5.8: Stator Currents (Id and Iq) and Electromagnetic Torque under No-Load

The simulation results obtained under there is no mechanical load resisting the motor's rotation, there is no need for the motor to produce significant torque. Consequently, the stator currents remain near zero.

5.1.3 Variable Speed Operation of the Motor

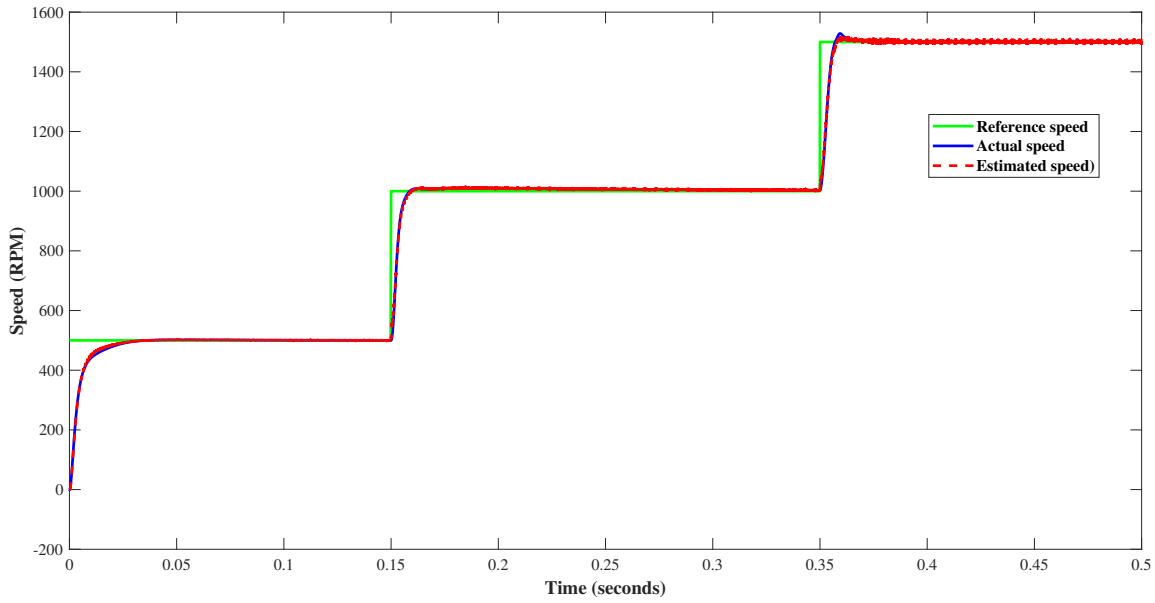

Figure 5.9: Variable Speed Response During Variable Speed Operation of the Motor

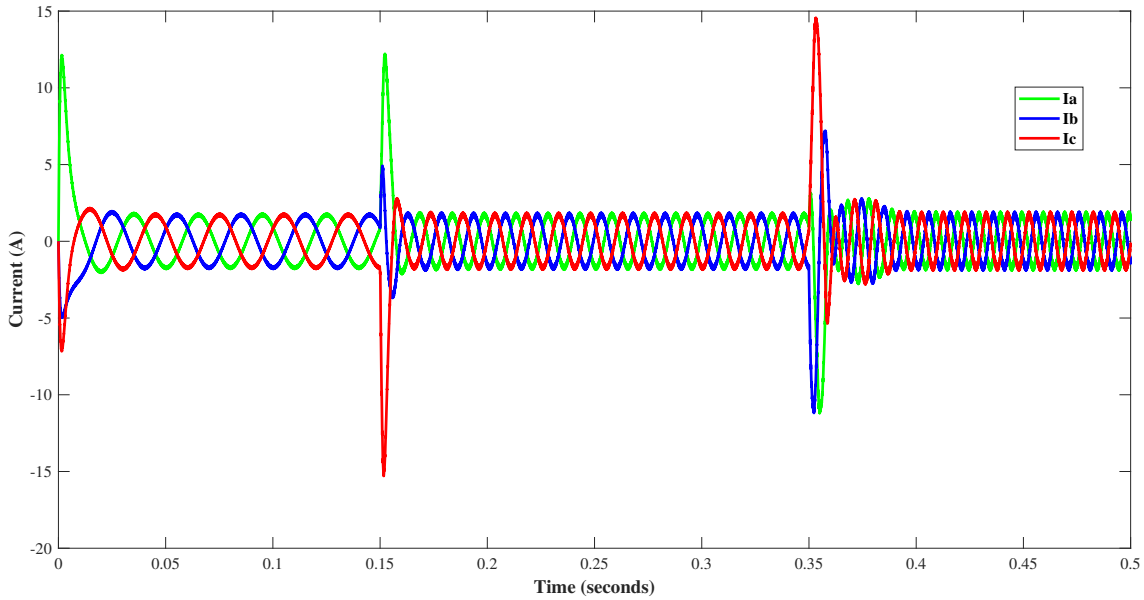

Figure 5.10: Current Response at Variable Speed

Figure 5.9 shows the proposed estimator successfully tracks various reference step inputs, with the estimated speed closely following the reference values.

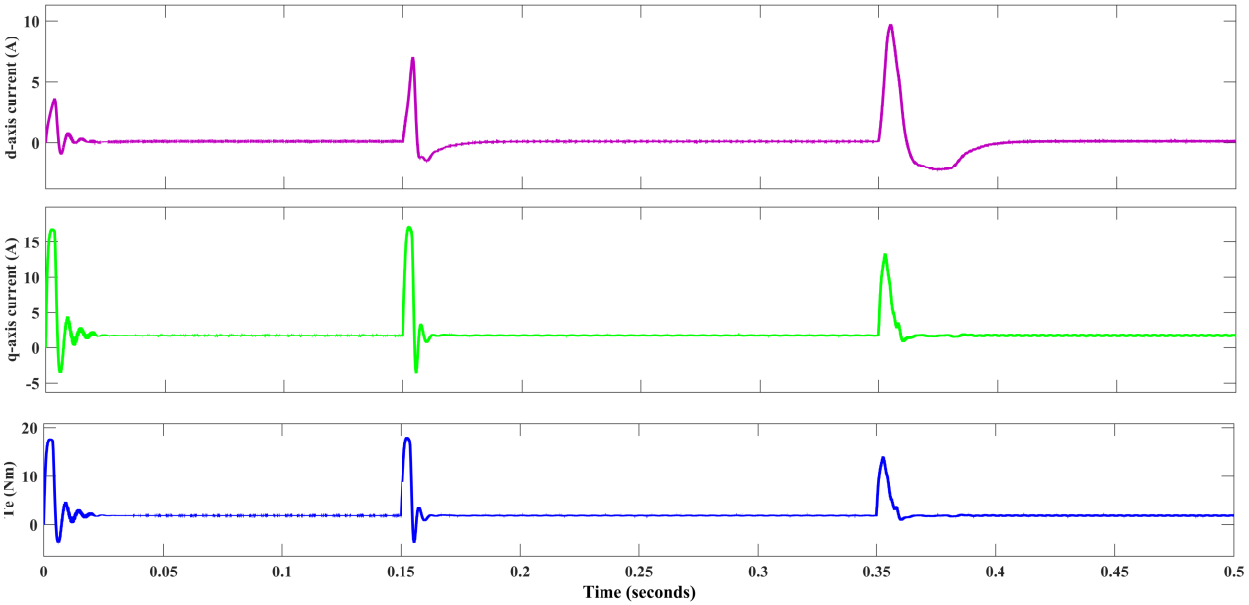

Figure 5.11: Stator Currents (Id,Iq) and Electromagnetic Torque during Variable Operation of PMSM

The simulation results, as shown in figure 5.11, indicate that the electromagnetic torque (Te) and stator current (Id and Iq) vary as the motor's speed changes. They will increase or decrease accordingly to match the load requirements at 500, 1000 and 1500 RPM and Id maintain zero as shown the simulation.

5.1.4 Forward Reverse Operation of PMSM

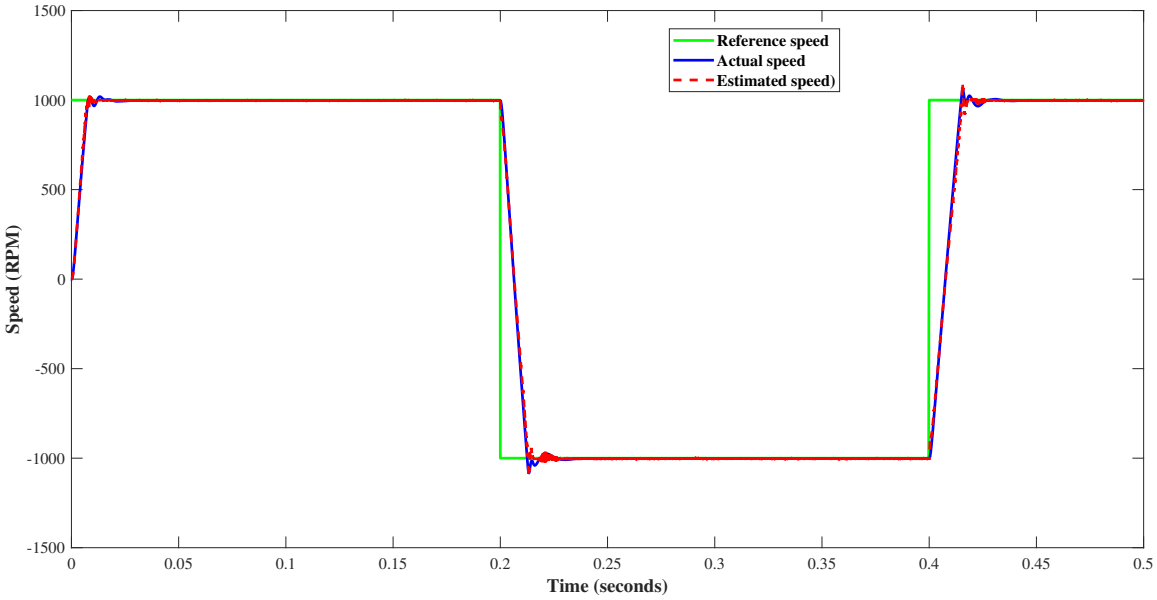

Figure 5.12: Speed Response during Forward-Reverse Operation of PMSM

Figure 5.12 illustrates simulation results while the motor operates at 1000 RPM in both forward and reverse conditions. These results demonstrate the control system's capability to sustain stability and controlled responses across changing rotational directions.

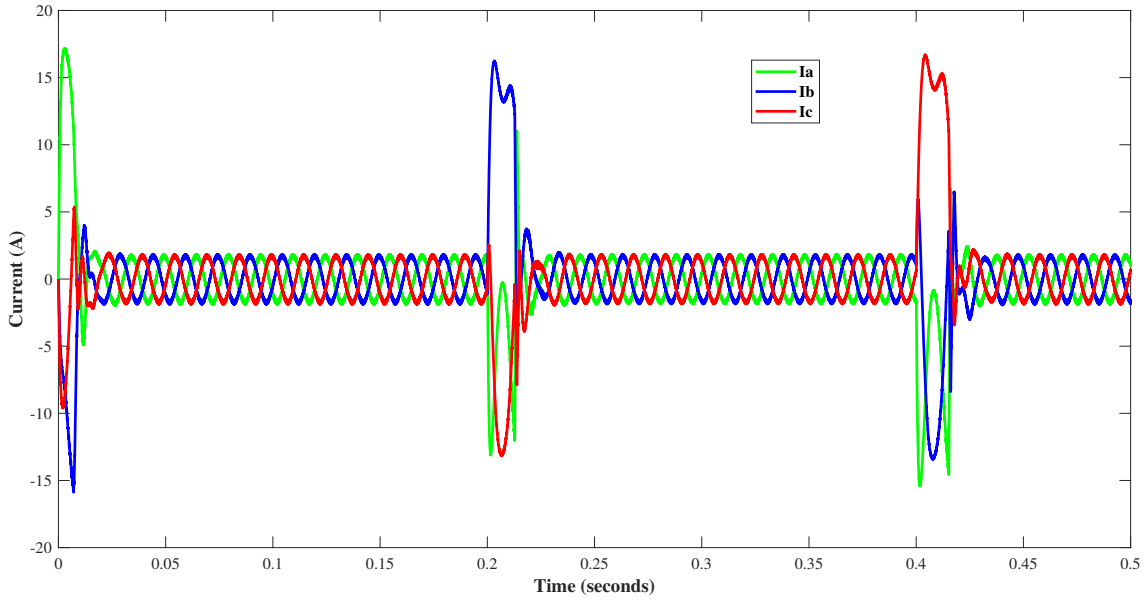

Figure 5.13: Current Response during Forward-Reverse Operation of PMSM

The simulation results show also the efficient modulation of stator currents, enabling the delivery of requisite torque and effectively surmounting challenges linked to fluctuating loads.

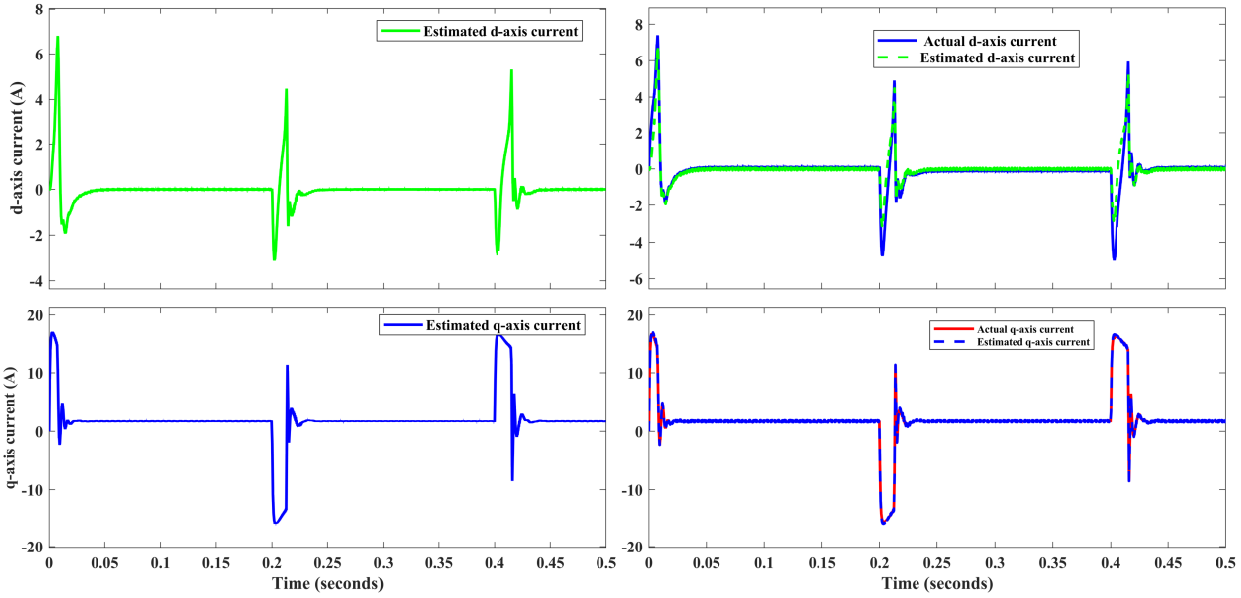

Figure 5.14: Actual and Estimated d-q Currents during Motor Forward-Reverse Operation with 1.8 Nm Load

The simulation shows close tracking between estimated and actual values of Id and Iq as speed varies, ensuring the motor adapts Iq to match load torque and maintain steady state.

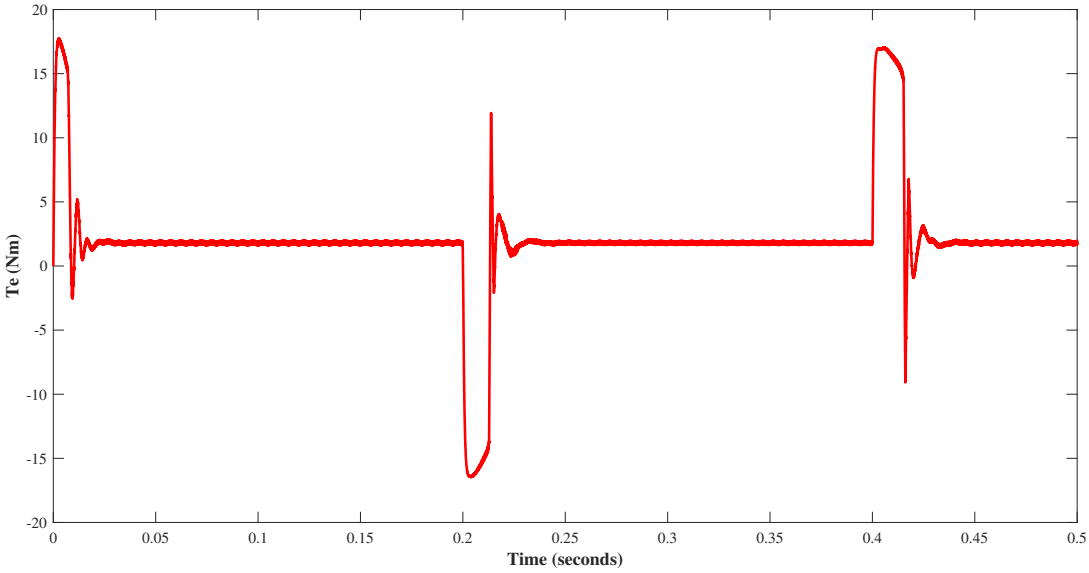

Figure 5.15: Electromagnetic Torque Response in Forward and Reverse Operation of PMSM

As shown in figure 5.15, torque (Te) generated by the motor varies as the motor responds to both forward and reverse speed commands. Torque (Te) is adjusted to match the load requirements and the desired speed, whether positive or negative. This demonstrates the motor's ability to produce torque in both directions.

5.2 Simulation Results for Speed Control of PMSM using Fuzzy Logic (FL-PI) controller

5.2.1 Dynamic Simulation Results of PMSM With Load

5.2.1.1 Actual and Estimated Speed

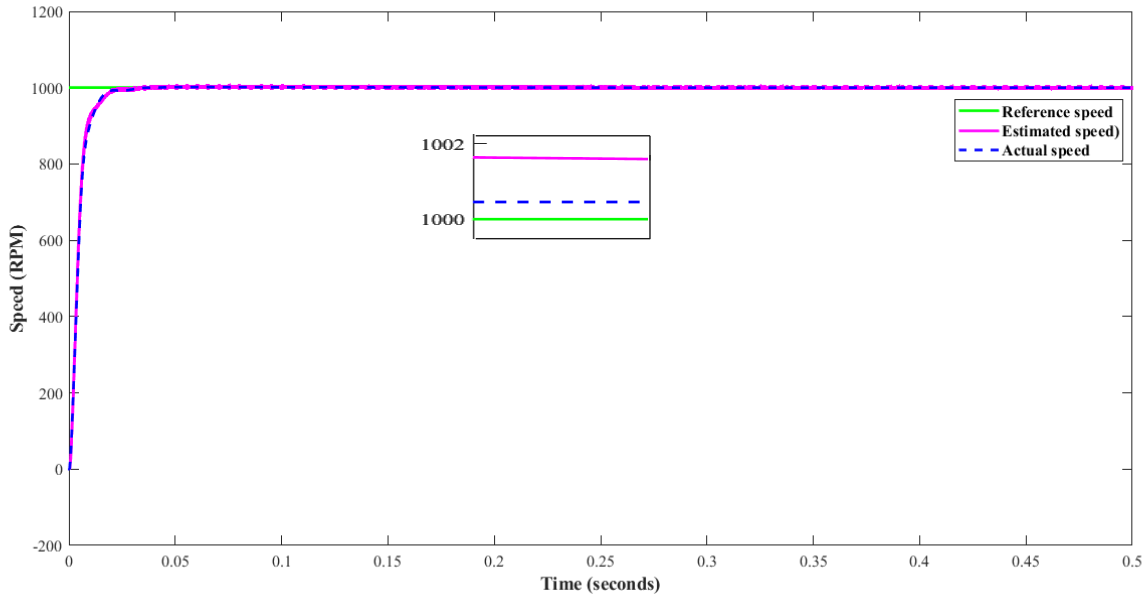

Figure 5.16: Actual and Estimated Speed in the 1.8 Nm Load Condition

The simulation results, conducted using a step-speed input with a load torque of 1.8 Nm as shown in figure 5.16, illustrate the motor's speed response when commanded to operate at 1000 RPM. Initially, the speed response with fuzzy logic (FL-PI) controller exhibits overshoot 0.2% and settling time 0.036 seconds for the provided reference.

5.2.1.2 Three Phase Stator Current and Estimated d-q Current

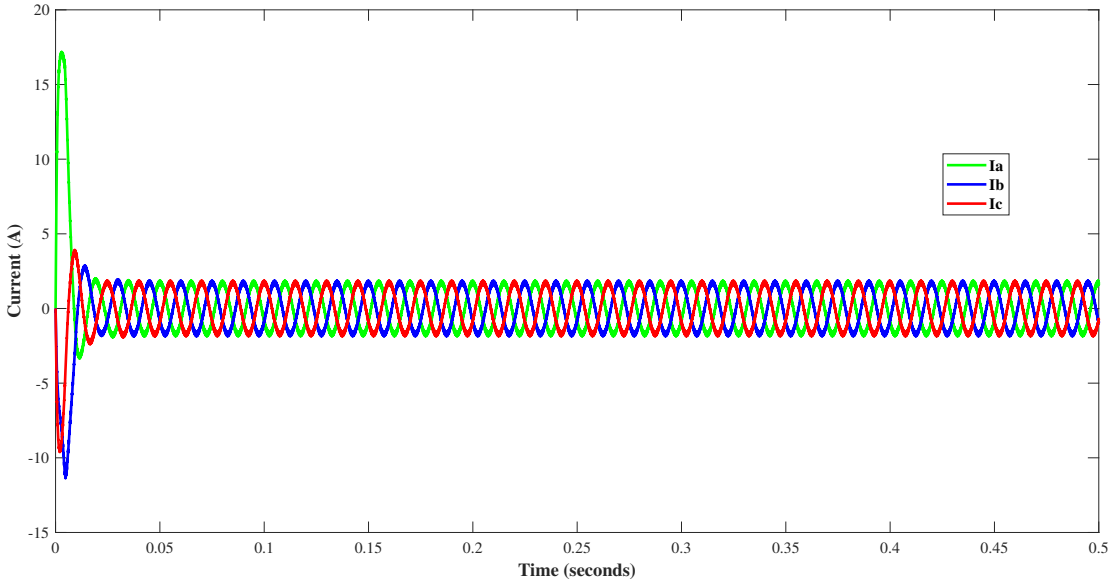

Figure 5.17: Three Phase Stator Current Over Time in a Loaded Motor (1.8 Nm Torque)

.

Initially, the current isn't sinusoidal but transitions to a sinusoidal form as the motor achieves the commanded speed from the controller. figure 5.17 displays the motor's three-phase currents at a 1000 RPM reference speed. The FL-PI controller dynamically adapts stator currents to handle the load on the shaft.

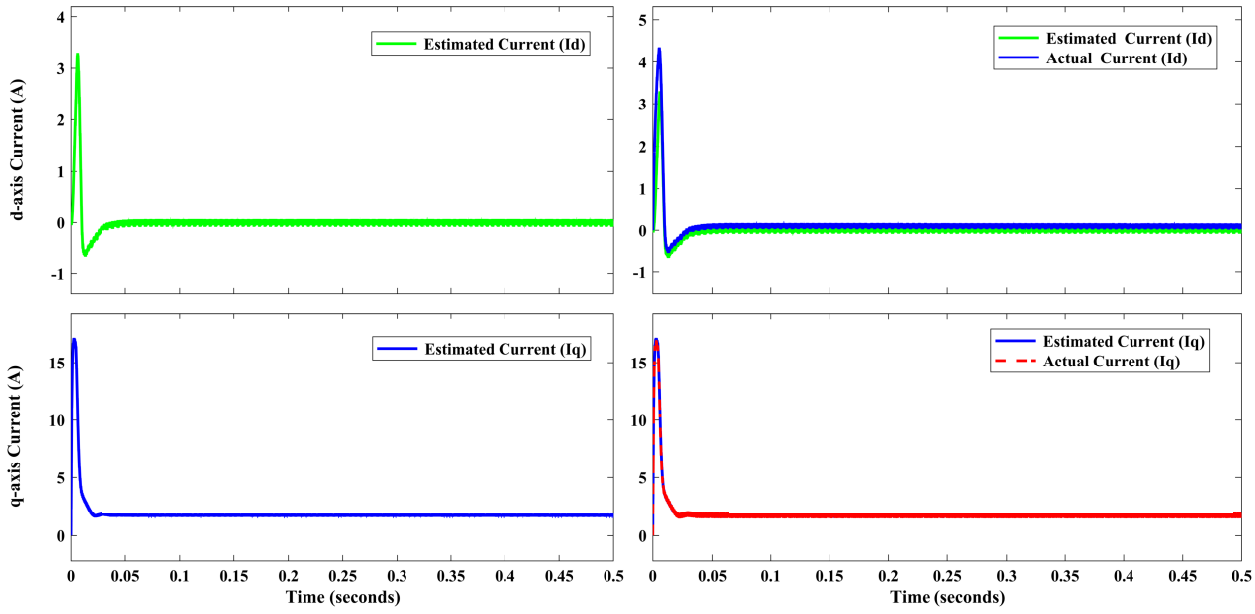

Figure 5.18: The Actual and Estimated d-q Current in Loaded Motor (1.8 Nm Torque)

5.2.1.3 Electromagnetic Torque

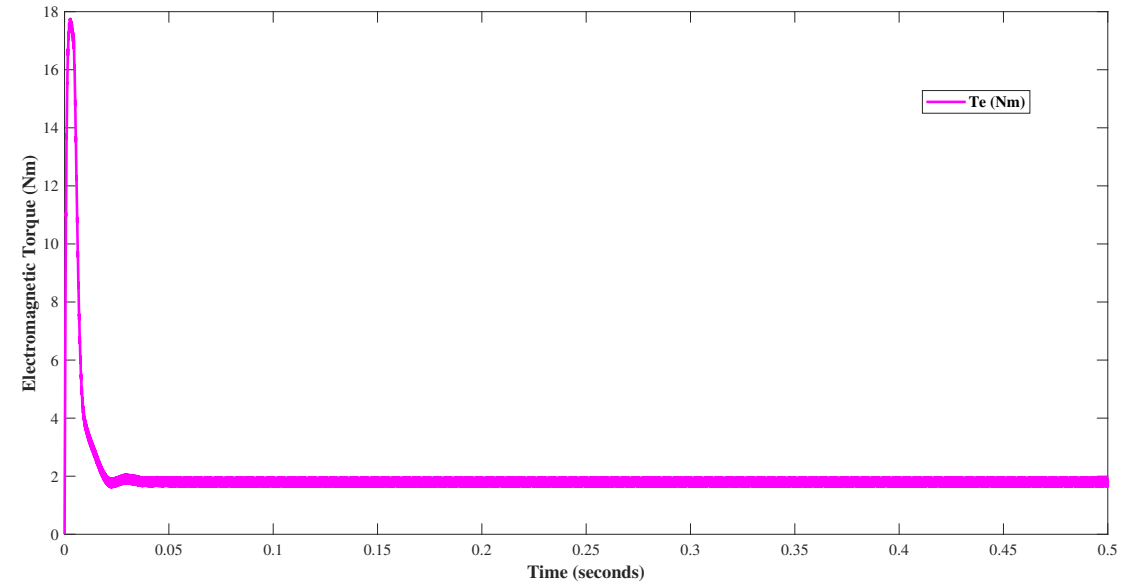

Figure 5.19: The Electromagnetic Torque (Te) in a Loaded Motor (1.8 Nm Torque)

5.2.2 Dynamic Simulation Results of PMSM Without Load

5.2.2.1 Actual and Estimated Speed

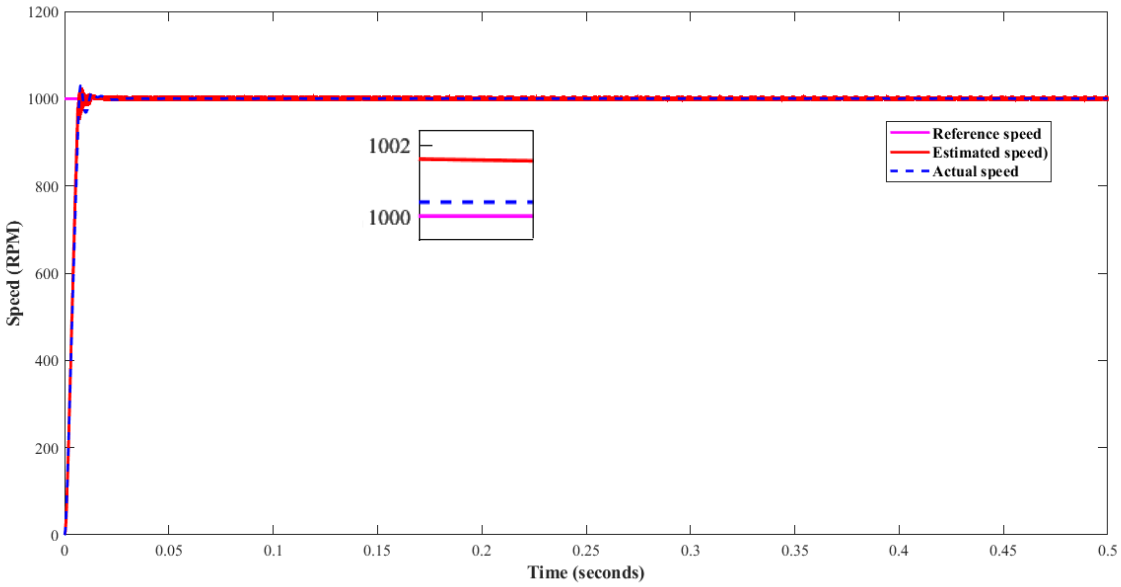

Figure 5.20: The Actual and Estimated Speed Without Load Torque

In figure 5.20, simulation shows an initial 2.6% overshoot in speed response using the fuzzy logic (FL-PI) controller, stabilizing within 0.3 seconds with the given reference.

5.2.2.2 Three Phase Stator Current and Estimated d-q Current

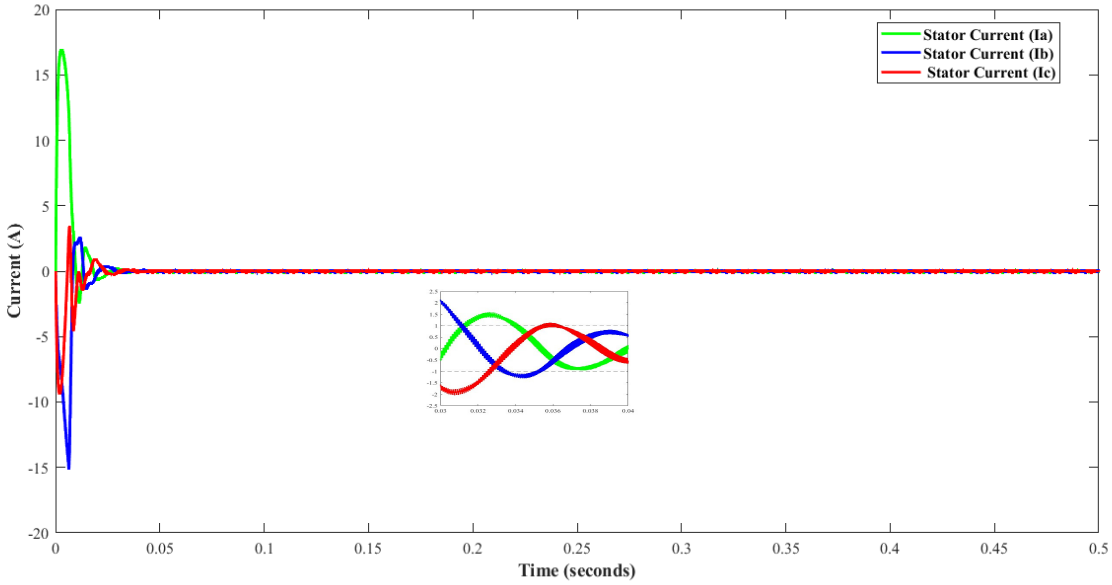

Figure 5.21: Three Phase Stator Current in Unloaded Condition

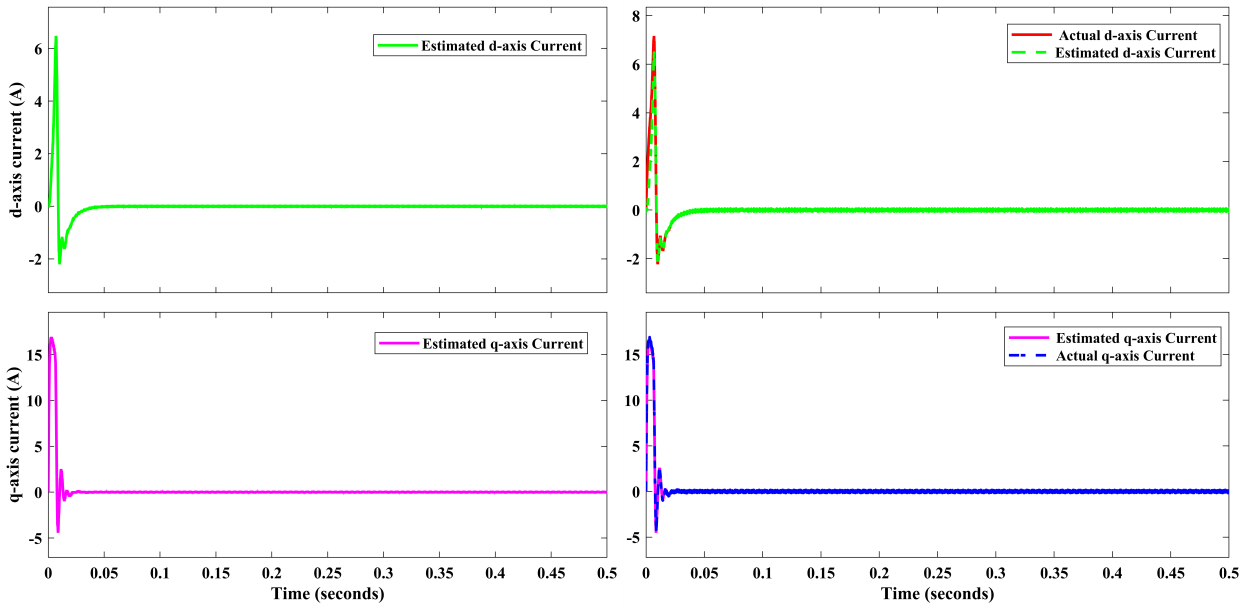

Figure 5.22: The Actual and Estimated d-q Current Without Load Torque

The simulation result depicted in figure 5.22 under no-load conditions, indicate that both the d-axis current and the q-axis current are very close to zero, and the estimated d-q current closely follows the actual d-q current.

5.2.2.3 Electromagnetic Torque

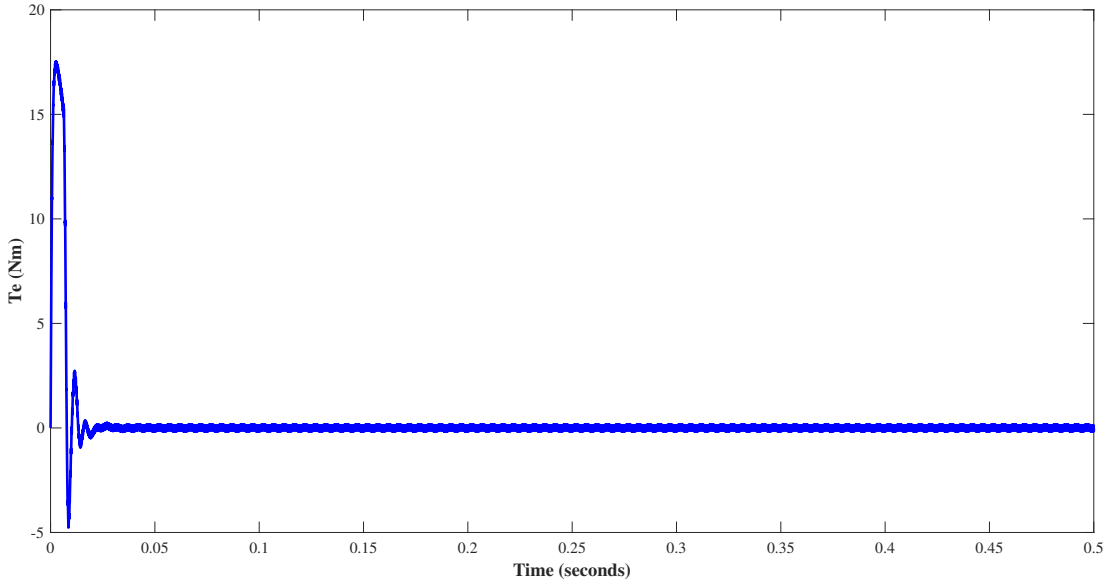

Figure 5.23: The Electromagnetic Torque (Te) Without Load Torque

The simulation result shows that torque generated by using motor is zero. Since there is no mechanical load resisting the motor rotation, there is no need for the motor to produce significant torque.

5.2.3 Variable Speed Operation of the Motor

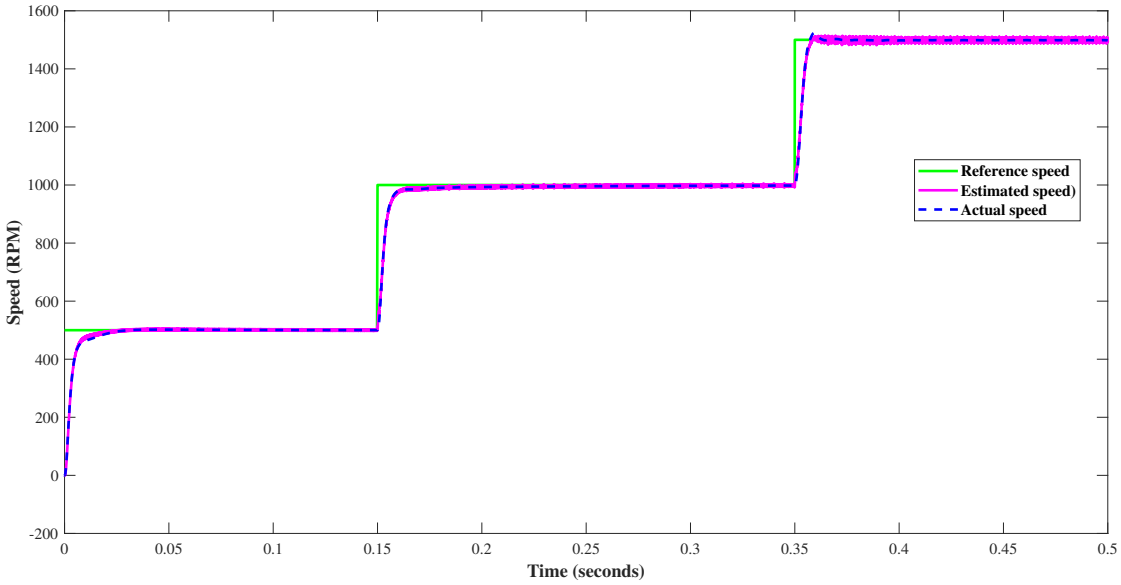

Figure 5.24: Variable Speed Response During Variable Speed Operation of the Motor in a Loaded Motor (1.8 Nm Torque)

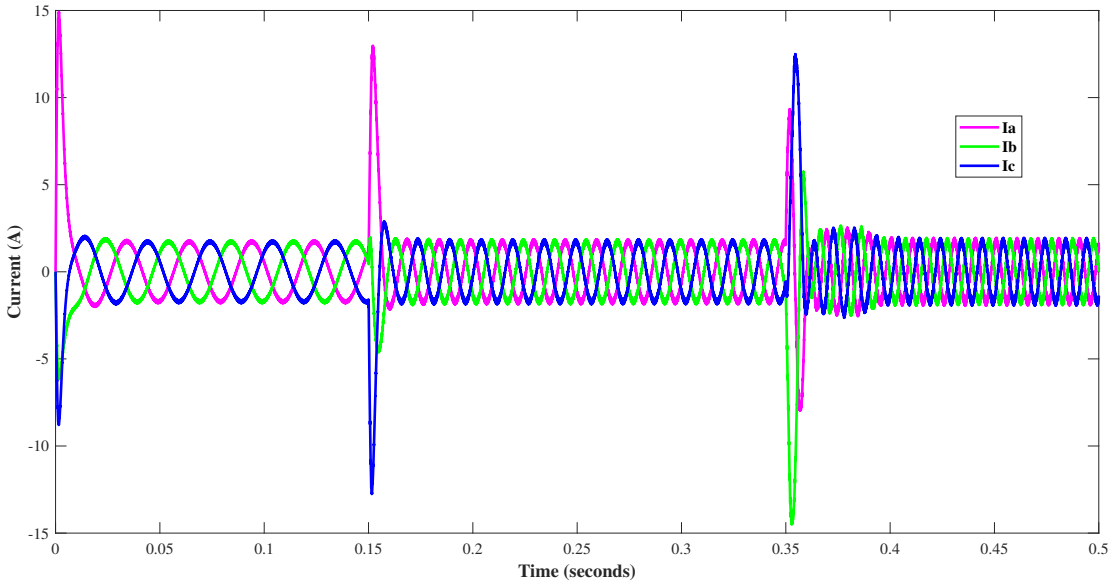

Figure 5.25: Current Response During Variable Speed Operation of the Motor in a Loaded Motor (1.8 Nm Torque)

Figures 5.24 and 5.25 depict the speed response of the fuzzy logic (FL-PI) controller for the motor under variable reference speeds (500, 1000, and 1500 RPM) while subjected to a 1.8 Nm load torque. The simulation results shown the tracking of various reference step inputs by the proposed estimator, with the estimated speed closely matching the reference.

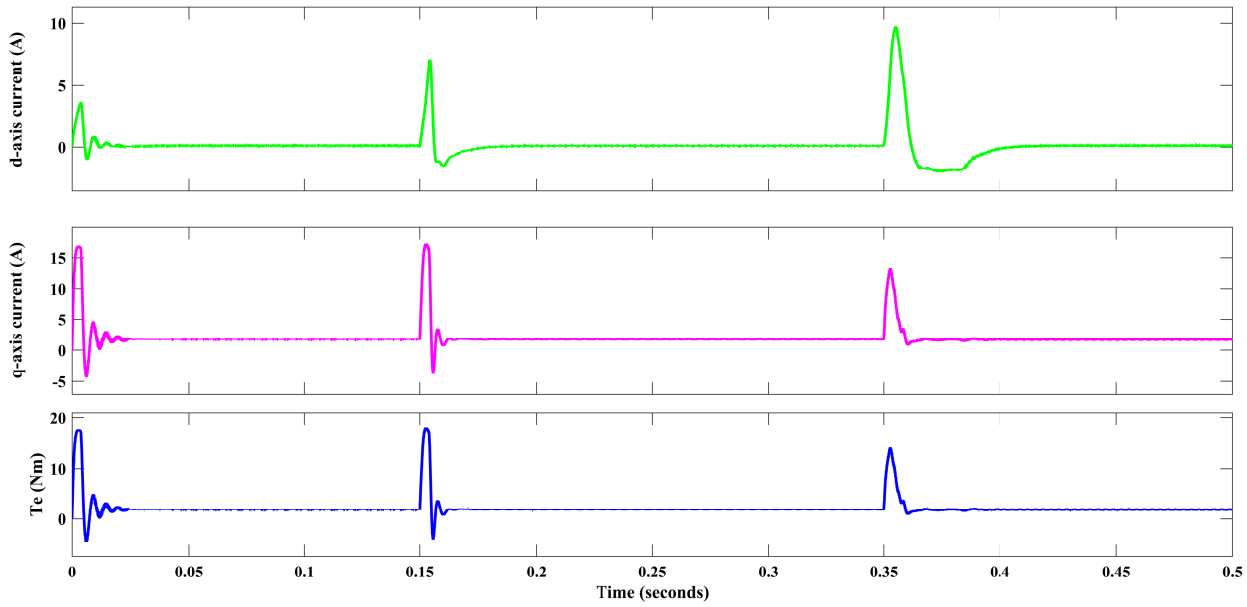

Figure 5.26: Stator Currents (Id,Iq) and Electromagnetic Torque (Te) during Variable Speed Operation of the Motor

The simulation results in figure 5.26 demonstrate variations in electromagnetic torque (Te) and stator current (Id,Iq) with changing motor speed. They adjust to meet load requirements at 500, 1000, and 1500 RPM, while Id remains at zero, as observed in the fuzzy logic (FL-PI) controller simulation.

5.2.4 Forward Reverse Operation of PMSM

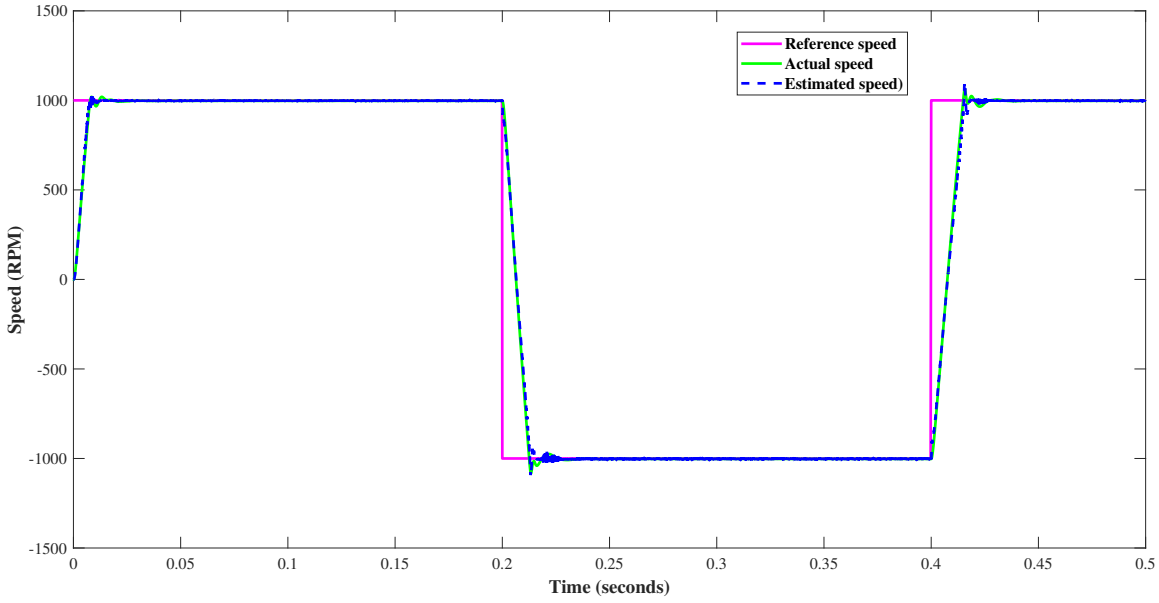

Figure 5.27: Speed Response during Forward-Reverse Operation of PMSM

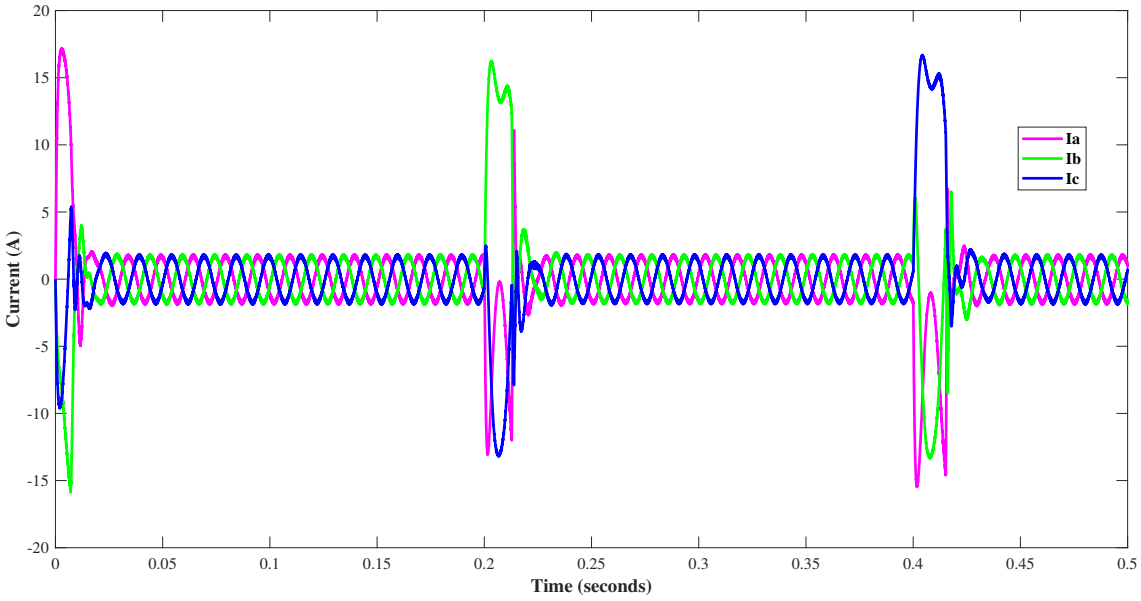

Figure 5.28: Current Response during Forward Reverse Operation of PMSM

Figure 5.27 and 5.28 shows simulation results for the (FL-PI) controller at 1000 RPM in both forward and reverse motor operation. It shows the system's stability and effective stator current modulation for delivering required torque, addressing fluctuating loads.

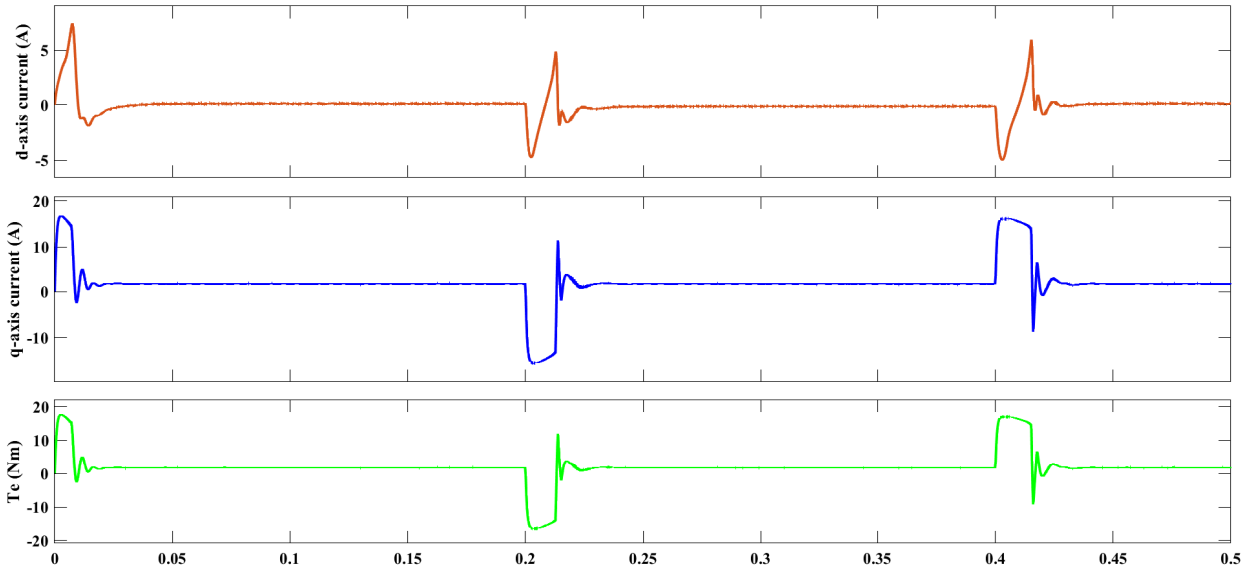

Figure 5.29: Stator Currents (Id,Iq) and Electromagnetic Torque (Te) during Forward-Reverse Operation of PMSM

As shown in figure 5.29, motor torque (Te) adapts for bidirectional operation with (FL-PI) controller, matching load and speed. Iq adjusts to accommodate load torque.

5.3 Performance and Robustness Analysis of PI and Fuzzy Logic (FL-PI) controller

5.3.1 Step Response Performance Analysis

Speed (RPM)	Overshoot (%)		Settling Time (second)		Steady State Error (%)	
	PI	FL-PI	PI	FL-PI	PI	FL-PI
500 Wl	3.6	0.4	0.2	0.187	0.1	0
500 Wol	5	2.6	0.28	0.037	0.4	0
1000 Wl	0.2	0.2	0.034	0.036	0	0
1000 Wol	2.6	2.6	0.37	0.3	0.1	0
1500 Wl	1.8	1.33	0.06	0.04	0	0
1500 Wol	3.8	3.3	0.03	0.05	0.2	0

Table 5.1: Comparison of Control Performance Metrics for Various Speeds and Load Conditions

Where: Wl stands for With Load and Wol stands for Without Load.

5.3.2 Robustness Analysis of Controller

5.3.2.1 Parameter Variation Analysis

Figure 5.30 and 5.31 display simulations involve a 25% increase and decrease in the resistance and inductance parameters while keeping all other parameters constant. These variations are examined at speeds of 500, 1000, and 1500 RPM as reference values.

The speed input varies to 500, 1000, and 1500 RPM at times 0, 0.15 and 0.35 seconds, respectively, in both the 25% increase and decrease scenarios. The simulation results indicate that, for all references, the actual speed reaches a steady state in less than 0.05 seconds. indicating the system reduced susceptibility to parameter changes.

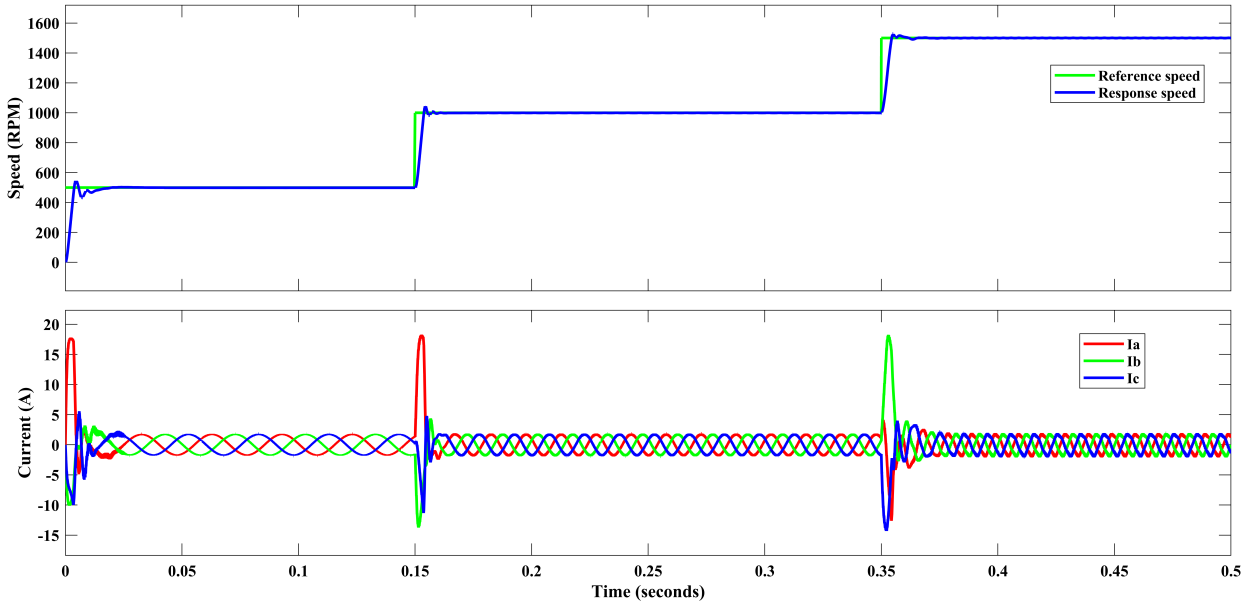

Figure 5.30: Variable Speed Operation with Rs and Ls Increased by 25% from Their Nominal Values

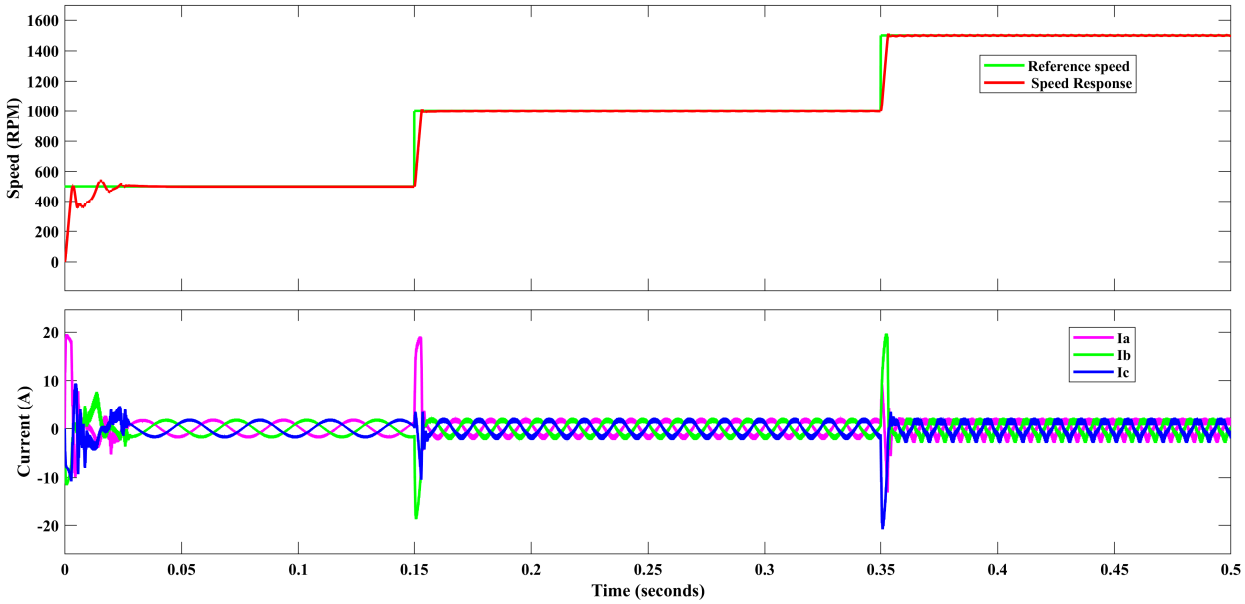

Figure 5.31: Variable Speed Operation with Rs and Ls Decreased by 25% from Their Nominal Values

No	Performance	PI Controller Overshoot (Mp) Settling time (ts)	Fuzzy Logic (FL-PI) controller Overshoot (Mp) Settling time (ts)
1	No-load torque at 1000 RPM	2.6% 0.37 sec	2.6% 0.014 sec
2	1.8 Nm load torque at 1000 RPM	0.2% 0.034 sec	0.2% 0.2 sec
3	Rs increased by 50%	3% 0.025 sec	0.2% 0.031 sec
4	Rs increased by 25%	0.3% 0.032 sec	0.2% 0.035 sec
5	Inductance (L) increased by 50%	2.5% 0.051 sec	2.1% 0.060 sec
6	Inductance (L) increased by 25%	1.9% 0.034 sec	1.3% 0.036 sec
7	Rs & L increased by 50%	2.2% 0.050 sec	1.9% 0.054 sec
8	Rs & L increased by 25%	1.5% 0.034 sec	1% 0.036 sec

Table 5.2: Performance of PI and (FL-PI) Controller Under Different Conditions and Parameter Variations

5.3.2.2 Impact of Disturbances Analysis

The motor initially operates in both PI and (FL-PI) speed control modes without any load. Subsequently 3 Nm load torque is applied and removed to assess the disturbance removal capability of both controllers. The load is suddenly applied at 0.2 seconds from the starting point and removed just as abruptly at 0.35 seconds for both the PI and (FL-PI) controllers. The corresponding responses for speed, phase current, and torque in both scenarios are shown in figure 5.32 and 5.33, respectively.

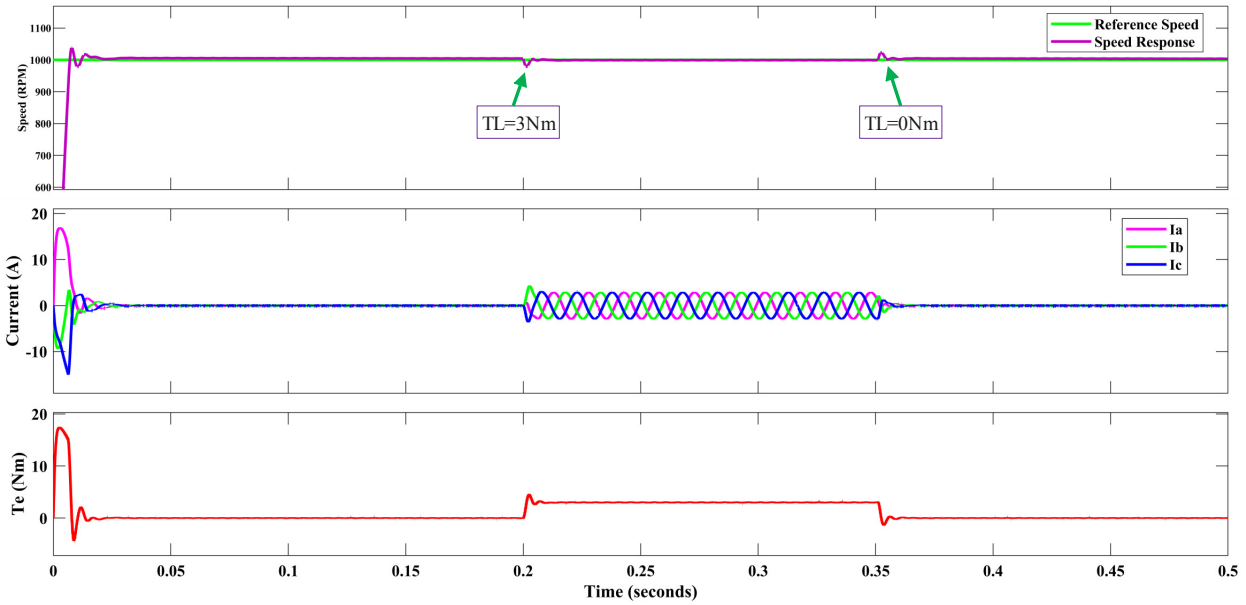

Figure 5.32: Response of Speed, Stator Current, and Torque to Load Torque Variations with PI Controller

As shown in the simulation results in figure 5.32 and 5.33 for both the PI and (FL-PI) scenarios, when an external load torque of 3 Nm is suddenly applied to the motor's shaft at 0.2 seconds, the motor's speed experiences an initial drop. This drop occurs because the torque generated by the motor is momentarily insufficient to overcome the increased load. Similarly, when the external load torque is reduced to 0 Nm at 0.35 seconds, the motor's speed tends to rise momentarily. This occurs because the motor continues to generate torque at a level appropriate for the previous load conditions but quickly returns to the speed command value after a short period of time with The PI controller exhibits a 3.9% overshoot, while the FL-PI controller shows a 3% overshoot.

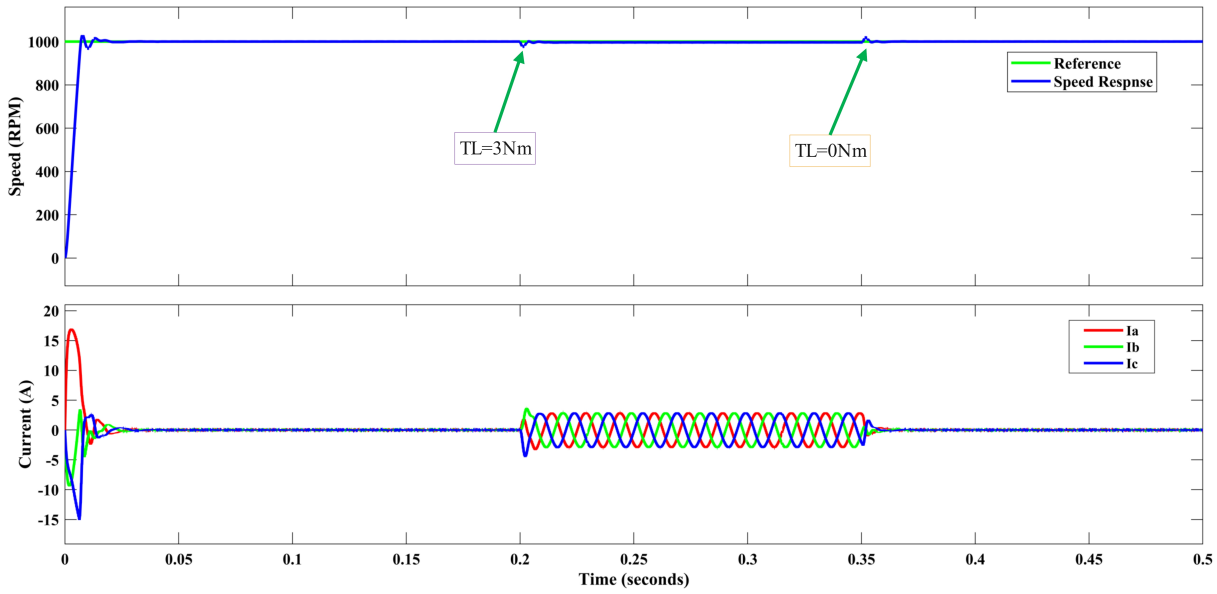

Figure 5.33: Response of Speed and Stator Current to Load Torque Variations with FL-PI Controller

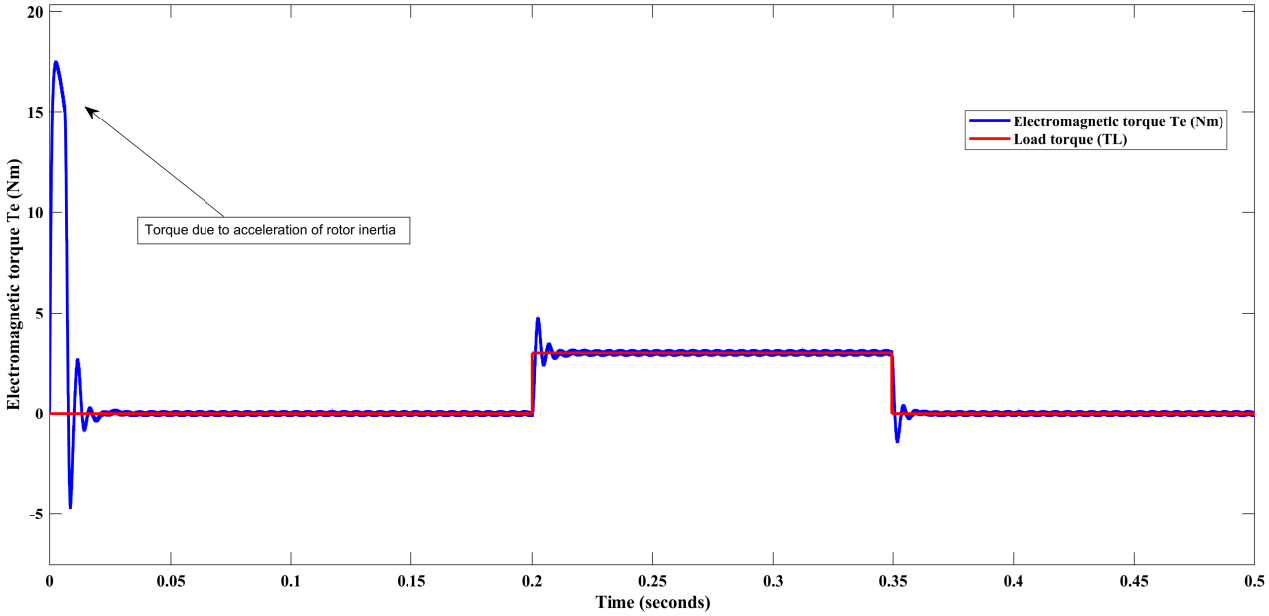

Figure 5.34: Response of Electromagnetic Torques to Load Torque Variations with FL-PI Controller

In figure 5.34, when a 3 Nm load torque is applied to the motor's shaft at 0.2 seconds, the electromagnetic torque increases to balance the load and maintain speed. The controller adjusts stator currents to raise electromagnetic torque (Te). Conversely, when the external load torque drops to 0 Nm at 0.35 seconds, Te decreases to prevent speed overshooting, helping the motor return to the desired speed.

5.4 Comparison of PI and FL-PI Controller Performance

A comparison of drive performance under PI and FL-PI speed control overlays speed responses in figure 5.35. The speed response of motor under no-load condition (TL = 0 Nm) with PI controller exhibits maximum percent overshoot of 5% and steady-state error of 0.4. FL-PI shows faster response times compared to PI when the speed is set at 500 rpm. Using FL-PI results in notable improvements, reducing the maximum overshoot from 5% to 2.6% and decreasing the steady state error from 0.4 to 0.

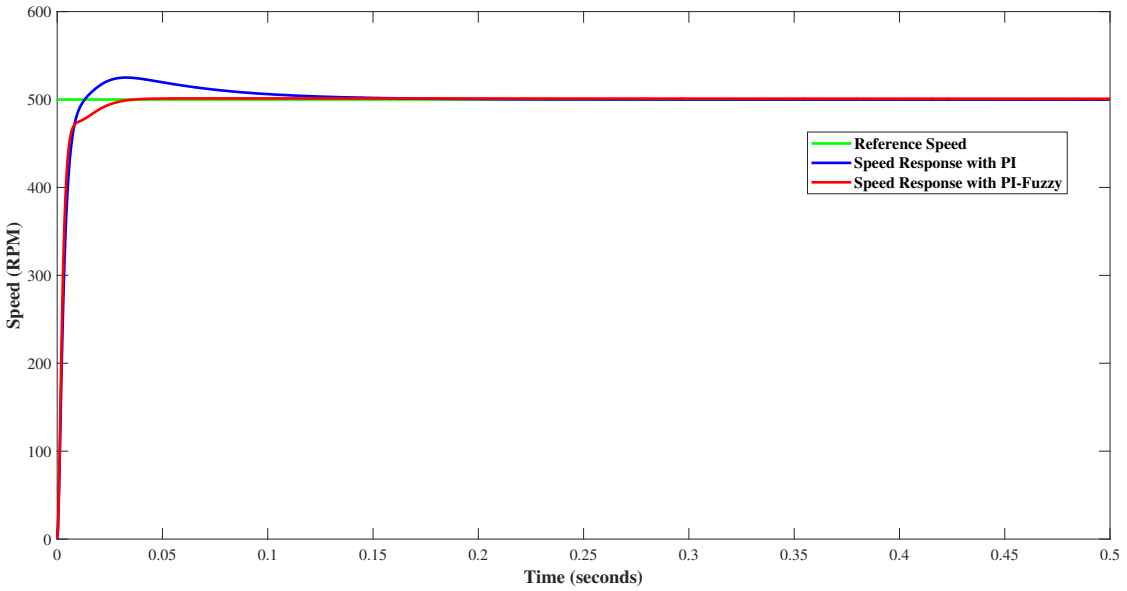

Figure 5.35: Comparison of PI with FL-PI in Speed Response for PMSM

In the previous sections, as shown in figures 5.32 and 5.33, with a 3 Nm load torque (TL=3 Nm) suddenly applied at 0.2 seconds and removed at 0.35 seconds from the starting point, at a constant speed of 1000 RPM, the PI controller exhibits a maximum percent overshoot of 4% and a steady state error of 0.8. It's evident that FL-PI controller demonstrates faster response times compared to PI when the speed is set to 1000 RPM. By using FL-PI controller, the maximum overshoot improves from 4% to 3%, and the steady state error decreases from 0.8 to 0.2.

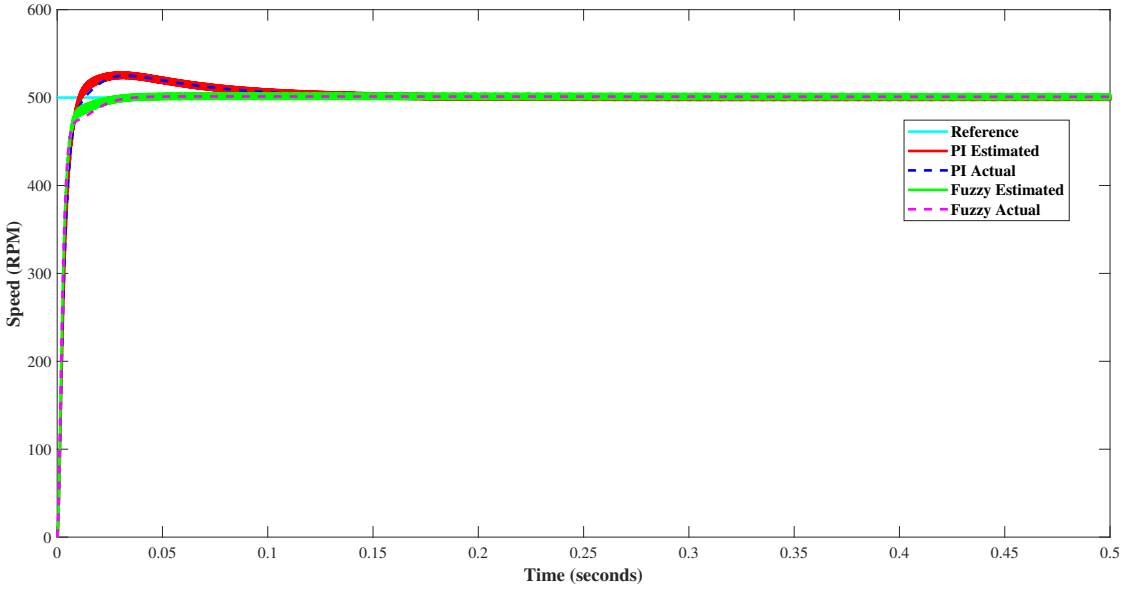

Figure 5.36: Comparison of PI with FL-PI in Speed Response for PMSM Including Estimator

Figure 5.36 demonstrates that under no-load torque conditions, the system's output speed closely follows the reference speed when subjected to a step input of 500 rpm. This figure illustrates that both PI and FL-PI controller effectively track the specified performance criteria of the reference input for both estimated and actual speed.

Chapter 6

Conclusion and Recommendation

6.1 Conclusion

In this thesis report, the design and implementation of a Model Reference Adaptive System (MRAS) to address challenges associated with mechanical position sensors for PMSM speed control have been undertaken. The design of MRAS speed estimator has been explored to overcome the issues related to mechanical speed sensors and parameter sensitivity in the context of speed control for permanent magnet synchronous motors. The model reference adaptive system has been determined to be highly suitable for the estimation of rotor speed in a PMSM. The simulation results demonstrate that the model reference adaptive system estimate rotor speed, achieving performance for motor speed control.

The performance of the controller has been evaluated in terms of its ability to analyze with loaded conditions and without load torque, its capability to track speed set points, its ability to reject disturbances, its response to drive step changes, and parameter variations of motor. The comparison between FL-PI and PI controllers has been carried out through simulations under no-load conditions and sudden load changes. The simulation results showed that the FL-PI controller outperformed the conventional PI controller in several aspects. Under no-load conditions, FL-PI controller exhibited faster response times, reduced overshoot from 5% to 2.6% and eliminated steady state error compared to the PI controller. Additionally, when subjected to sudden load torque changes, the FL-PI controller demonstrated better disturbance rejection capabilities, with overshoot of 3% compared to 3.9% for PI controller. Moreover, the FL-PI controller proved to be robust against parameter variations compared to PI controller.

6.2 Recommendations for Future Works

To further enhance the sensorless system's overall performance, propose the inclusion of an additional stator resistance estimator within the Model Reference Adaptive System (MRAS) framework. This incorporation of stator resistance estimation holds significant potential for improving the performance of sensorless control systems for Permanent Magnet Synchronous Motors (PMSMs). The key benefits of including stator resistance estimation:

- Determining the stator resistance can improve performance and offer an opportunity to estimate the motor's temperature.
- Improved Speed Estimation: Real-time stator resistance estimation enhances speed accuracy, especially under varying loads.

To enhance the performance of MRAS speed estimators in PMSMs, especially in low-speed and startup scenarios, recommend the following strategies:

- **Combined Estimation:** Increase accuracy by combining MRAS with other speed estimation methods, particularly in low-speed and transient conditions.

Hardware Implementation: Develop hardware-efficient MRAS based sensorless control systems for real-time control in applications.

Bibliography

[1] Dal Y Ohm. Dynamic model of pm synchronous motors. *Drivetech, Inc., Blacksburg, Virginia*, 16, 2000.

[2] Tomy Sebastian, G Slemon, and M Rahman. Modelling of permanent magnet synchronous motors. *IEEE Transactions on Magnetics*, 22(5):1069–1071, 1986.

[3] Pragasen Pillay and Ramu Krishnan. Modeling, simulation, and analysis of permanent-magnet motor drives. i. the permanent-magnet synchronous motor drive. *IEEE Transactions on industry applications*, 25(2):265–273, 1989.

[4] B Adhavan, A Kuppuswamy, G Jayabaskaran, and V Jagannathan. Field oriented control of permanent magnet synchronous motor (pmsm) using fuzzy logic controller. In *2011 IEEE Recent Advances in Intelligent Computational Systems*, pages 587–592. IEEE, 2011.

[5] Young Sam Kim, Sang Kyoon Kim, and Young Ahn Kwon. Mras based sensorless control of permanent magnet synchronous motor. In *SICE 2003 Annual Conference (IEEE Cat. No. 03TH8734)*, volume 2, pages 1632–1637. IEEE, 2003.

[6] Bhagyashree Shikkewal and Vaishali Nandanwar. Fuzzy logic controller for pmsm. *Int. J. Electr. Electron. Eng*, 1(3):73–78, 2012.

[7] (pmsm) construction & working principle. https://electricalbaba.com/permanent-magnet-synchronous-motor-pmsm-construction-working-principle/, 2018. Accessed on June 16, 2018.

[8] Hicret YETİŞ and Taner GÖKTAŞ. Comparative design of permanent magnet synchronous motors for low-power industrial applications. *Balkan Journal of Electrical and Computer Engineering*, 8(3):218–224, 2020.

[9] Shawon Kumar Baral. Closed loop control of pmsm motor: Field oriented control using hall sensors, 2021.

[10] Bhagyashree Shikkewal and Vaishali Nandanwar. Fuzzy logic controller for pmsm. *Int. J. Electr. Electron. Eng*, 1(3):73–78, 2012.

[11] Babak Nahid-Mobarakeh, Farid Meibody-Tabar, and Franois-Michel Sargos. Back emf estimation-based sensorless control of pmsm: Robustness with respect to measurement errors and inverter irregularities. *IEEE Transactions on Industry Applications*, 43(2):485–494, 2007.

[12] Fabio Genduso, Rosario Miceli, Cosimo Rando, and Giuseppe Ricco Galluzzo. Back emf sensorless-control algorithm for high-dynamic performance pmsm. *IEEE Transactions on Industrial Electronics*, 57(6):2092–2100, 2009.

[13] Lu An, David Franck, and Kay Hameyer. Sensorless field oriented control using back-emf and flux observer for a surface mounted permanent magnet synchronous motor. *International Journal of Applied Electromagnetics and Mechanics*, 45(1-4):845–850, 2014.

[14] VS Nagarajan, M Balaji, and V Kamaraj. Back-emf-based sensorless field-oriented control of pmsm using neural-network-based controller with a start-up strategy. In *Artificial Intelligence and Evolutionary Algorithms in Engineering Systems: Proceedings of ICAEES 2014, Volume 2*, pages 449–457. Springer, 2015.

[15] Jianning Lu, Wenbin Wang, Junshuai Zhang, and Jiaqi Su. Research on sensorless control of built-in permanent magnet synchronous motor based on extended kalman filter algorithm. In *Journal of Physics: Conference Series*, volume 2479, page 012063. IOP Publishing, 2023.

[16] Yanhao Li, Zhiyong Lan, Xiaoyang Su, and Shanqi Dai. Speed and position estimation algorithm of permanent magnet synchronous motor based on extended kalman filter. In *The proceedings of the 16th Annual Conference of China Electrotechnical Society: Volume I*, pages 515–522. Springer, 2022.

[17] Samia Allaoui, Yahia Laamari, Kheireddine Chafaa, and Salah Saad. Position and speed estimation of pmsm based on extended kalman filter tuned by biogeography-based-optimization. *Journal Européen des Systèmes Automatisés*, 54(4), 2021.

[18] Mohamad Ikhwan Nordin, Jurifa Mat Lazi, Md Hairul Nizam Talib, and Zulkifilie Ibrahim. Speed sensorless control for pmsm drives using extended kalman filter.

[19] P Borsje, TF Chan, YK Wong, and Siu Lau Ho. A comparative study of kalman filtering for sensorless control of a permanent-magnet synchronous motor drive. In *IEEE International Conference on Electric Machines and Drives, 2005.*, pages 815–822. IEEE, 2005.

[20] Abhishek Bist and Sadhana V Jadhav. Sensorless control based on sliding mode observer for pmsm drive. In *2020 IEEE International Conference on Power Electronics, Drives and Energy Systems (PEDES)*, pages 1–6. IEEE, 2020.

[21] Hongryel Kim, Jubum Son, and Jangmyung Lee. A high-speed sliding-mode observer for the sensorless speed control of a pmsm. *IEEE transactions on Industrial Electronics*, 58(9):4069–4077, 2010.

[22] Mohamed Hassan, Osama Mahgoub, and Abdelatif El Shafei. Anfis based mras speed estimator for sensorless control of pmsm. In *2013 Brazilian Power Electronics Conference*, pages 828–835. IEEE, 2013.

[23] Ambarisha Mishra, Vasundhara Mahajan, Pramod Agarwal, and SP Srivastava. Mras based estimation of speed in sensorless pmsm drive. In *2012 IEEE Fifth Power India Conference*, pages 1–5. IEEE, 2012.

[24] Hajo Siraj. Design and simulation of fuzzy logic based self-tuning pi controller for permanent magnet synchronous motors. *AAIT, School of Graduate Studies, Department of Electrical and Computer Engineering*, 2017.

[25] Mahlet Legesse. Speed control of vector controlled pmsm drive using fuzzy logic-pi controller. *AAIT, School of Graduate Studies, Department of Electrical and Computer Engineering*, 2011.

[26] Shoeb Hussain and Mohammad Abid Bazaz. Sensorless control of pmsm drive using neural network observer. In *2016 IEEE 1st International Conference on Power Electronics, Intelligent Control and Energy Systems (ICPEICES)*, pages 1–5. IEEE, 2016.

[27] Ali Abdul Razzaq Altahir. Park and clark transformations: A short review. *no. April*, pages 2–5, 2020.

[28] Rami Awar, Yahya Khaled Al Ali, and Youssef Al Jrab. Speed and postition control of a dc motor. *American University of Beirut*, 2009.

[29] Kwang Hee Nam. *AC Motor Control and Electrical Vehicle Applications*. CRC Press, Boca Raton, 2nd edition, 2018. First Published 2018, eBook Published 9 November 2018.

[30] Haitham Abu-Rub, Atif Iqbal, and Jaroslaw Guzinski. *High performance control of AC drives with Matlab/Simulink*. John Wiley & Sons, 2021.

[31] John Chiasson. *Modeling and high performance control of electric machines*. John Wiley & Sons, 2005.

[32] BK Bose. Modern power electronics and ac drives prentice hall ptr. *Upper Saddle River*, 2002.

[33] Fem parameterized permanent magnet synchronous motor. https://www.mathworks.com/help/sps/ref/femparameterizedpmsm.html, 2019. MathWorks Online Documentation, Version R2019b.

[34] Glumineau Alain and Jesus de Leon-Morales. Sensorless ac motor control: Robust advanced design techniques and applications, 2005.

[35] NarahariKarthik Varma, Vurimi Homasekhar, and KondisettySai Krishna. Pmsm mathematical model comparison with simulink model.

[36] Torkel Glad. Reglerteknik: grundläggande teori, 4.,[omarb.] uppl.. ed. *Lund: Lund: Studentlitteratur*, 2006.

[37] Liuping Wang, Shan Chai, Dae Yoo, Lu Gan, and Ki Ng. *PID and predictive control of electrical drives and power converters using MATLAB/Simulink*. John Wiley & Sons, 2015.

[38] Kumpati S Narendra and Anuradha M Annaswamy. *Stable adaptive systems*. Courier Corporation, 2012.

[39] Li Yongdong and Zhu Hao. Sensorless control of permanent magnet synchronous motor—a survey. In *2008 IEEE Vehicle Power and Propulsion Conference*, pages 1–8. IEEE, 2008.

[40] Ambarisha Mishra, Vasundhara Mahajan, Pramod Agarwal, and SP Srivastava. Mras based estimation of speed in sensorless pmsm drive. In *2012 IEEE Fifth Power India Conference*, pages 1–5. IEEE, 2012.

[41] Shicai Fan, Wuqiao Luo, Jianxiao Zou, and Gang Zheng. A hybrid speed sensorless control strategy for pmsm based on mras and fuzzy control. In *Proceedings of the 7th International Power Electronics and Motion Control Conference*, volume 4, pages 2976–2980. IEEE, 2012.

[42] AMARE TIGU ZEWUDIE. Sensorless speed control of permanent magnet synchronous motor using model reference adaptive system. 2019.

[43] I Landau. A hyperstability criterion for model reference adaptive control systems. *IEEE Transactions on Automatic Control*, 14(5):552–555, 1969.

[44] Lipeng Wang, Huaguang Zhang, and Xiuchong Liu. Robust sensorless of adrc controlled pmsm based on mras with stator resistance identification. In *Proceedings of the 30th Chinese control conference*, pages 3575–3579. IEEE, 2011.

[45] Takile Akuma Kanaha. *Fuzzy Logic Based Proportional Integral Controller of Frequency for Small Hydropower Plant*. PhD thesis, Addis Ababa Science and Technology University, 2019.

[46] SP Singh, AK Gautam, Jyoti Dubey, JP Pandey, and RP Payasi. Performance comparison of pmsm drive using pi and fuzzy logic based controllers. In *2016 IEEE Uttar Pradesh Section International Conference on Electrical, Computer and Electronics Engineering (UPCON)*, pages 563–569. IEEE, 2016.

[47] Zhao-Hua Liu, Jie Nie, Hua-Liang Wei, Lei Chen, Xiao-Hua Li, and Ming-Yang Lv. Switched pi control based mras for sensorless control of pmsm drives using fuzzy-logic-controller. *IEEE Open Journal of Power Electronics*, 3:368–381, 2022.

[48] Mutasim Nour, Omrane Bouketir, C Eng Yong, et al. Self-tuning of pi speed controller gains using fuzzy logic controller. *Modern Applied Science*, 2(6):55–65, 2008.

Appendix A

Popov Hyper-Stability Theorem

The reference model (PMSM) with measured (id, iq) currents x= $[i_d \quad iq]^T$ and adjustable reference model of motor written as

$$\begin{cases} \dot{\hat{x}} = \hat{A}\hat{x} + \hat{B}u \\ Y = C\hat{x} \end{cases} \tag{A.1}$$

$$\hat{A} = \begin{bmatrix} -\frac{R_s}{Ld} & \hat{\omega_\epsilon} \\ -\hat{\omega_\epsilon} & \frac{-R_s}{Ld} \end{bmatrix}; \quad \hat{B}u = \begin{bmatrix} \frac{V_d}{L_d} \\ \frac{1}{Ld}[V_d - \lambda_f \hat{\omega_\epsilon}] \end{bmatrix} \tag{A.2}$$

$\hat{x} = [\hat{id} \quad \hat{iq}]^T$ is estimated currents and $\hat{\omega_\epsilon}$ estimated rotor speed.

the error $\zeta = \hat{x}$-x using equations 3.13 and 4.19 gives

$\dot{\zeta} = \dot{\hat{x}} - \dot{x}$

$$\begin{cases} \dot{\zeta} = \hat{A}\zeta + (\hat{A} - Ax) + \hat{B}u - Bu \\ \dot{\zeta} = \hat{A}\zeta + \omega_1 \end{cases} \tag{A.3}$$

Let $\omega = -\omega_1$. gives

$$\omega = [(\hat{A} - A)x + \hat{B}u - Bu] \quad and \quad \omega = \begin{bmatrix} -iq \\ id + \frac{\lambda_f}{Ld} \end{bmatrix} (\hat{\omega} - \omega) \tag{A.4}$$

$$= G(\hat{\omega} - \omega) \tag{A.5}$$

When the generalized error output is introduced as $V = D\zeta$ the generalized error dynamics can be calculated as

$$\begin{cases} \dot{\zeta} = \hat{A}\zeta + I\omega_1 \\ V = D\zeta \\ \omega_1 = -\omega_1 \\ \omega = G(\hat{\omega} - \omega) \end{cases} \tag{A.6}$$

Three requirements must be satisfied in order to ensure that equation A.6 is an asymptotically hyper stable system given a system.

A. The pair$[\hat{A}, I]$ is completely controllable and the pair $[D, \hat{A}]$ is completely observable.

B. The transfer function $y(s) = D(sI - \hat{A})^{-1}$ must be strictly positive real.

C The Popov's integral inequality, as seen in equation equation (A.7).

$$\int_0^{\infty} V^T \omega \, dt \geq -\eta^2 \tag{A.7}$$

η^2 positive constant and independent of time.

The rank $[I\hat{A} \quad I] = 2$ as well as rank as $[D^T \hat{A}^T D^T] = 2$ the **A** condition is fulfilled.

The second equation **B** is satisfied, If two definite symmetric matrices **P** and **Q** exists.

$$\hat{A}^T P + P\hat{A} = -Q. \tag{A.8}$$

Choosing $Q = I$ gives $P = \begin{bmatrix} \frac{Ld}{2R_s} & 0 \\ 0 & \frac{Ld}{2R_s} \end{bmatrix}$ condition two satisfied as result P and Q are both symmetrical and also definite to verify equation (A.11) an other integral inequality is applicable as shown

$$\int_0^{\infty} \left[\frac{df(t)}{dt}\right] (Gf(t)) \, dt \geq -\frac{1}{2} G f^2(0) \tag{A.9}$$

G is a positive constant. Inserting V and ω from equations equation (A.6) gives:

$$\int_0^{\infty} [(\hat{iq} - iq)(id + \frac{\lambda_f}{Ld}) - (\hat{id} - id)iq + (\hat{\omega}_\epsilon - \omega_\epsilon)] \, dt \geq -\eta^2 \tag{A.10}$$

The integral adaptive mechanism is used to estimate the rotor speed (ω_ϵ):

$$\hat{\omega}_\epsilon = \int_0^{\infty} \phi(e) \, dt + \hat{\omega}_\epsilon(0) \tag{A.11}$$

Inserting equation equation (A.11) in to equation (A.10) given inequality is expressed as:

$$\int_0^\infty \left[(\hat{iq} - iq)(id + \frac{\lambda_f}{Ld}) - (\hat{id} - id)iq\right] + \left(\int_0^\infty \phi(e)\, dt + \hat{\omega}_\epsilon(0) - \omega_\epsilon\right) dt \geq -\eta^2 \tag{A.12}$$

The first part of the inequality is the integral of the rate of change of some function with respect to time:

$$\int_0^\infty \frac{df(t)}{dt}\, dt \tag{A.13}$$

The function $f(t)$ is connected to these equations:

$$\frac{df(t)}{dt} = (\hat{iq} - iq)(id + \frac{\lambda_f}{Ld}) - (\hat{id} - id)iq \tag{A.14}$$

The second part of the inequality is similar to the function $Gf(t)$, which is defined as:

$$Gf(t) = \int_0^\infty \phi\, dt + \hat{\omega}_\epsilon(0) - \omega_\epsilon\, dt \tag{A.15}$$

Here, G represents the integral operator. The function ϕ relates to these equations:

$$\phi = G(\hat{iq} - iq)(id + \frac{\lambda_f}{Ld}) - (\hat{id} - id)iq \tag{A.16}$$

By choosing ϕ as defined in equation equation (A.16), The third requirement is verified, which provides the following definition of the adaptive mechanism in transfer function form:

$$\hat{\omega}_\epsilon = \int_0^\infty \left[(\hat{iq} - iq)(id + \frac{\lambda_f}{Ld}) - (\hat{id} - id)iq\right] dt + \hat{\omega}_\epsilon(0) \tag{A.17}$$

$$\hat{\omega_\epsilon} = \frac{K}{s}[(\hat{iq} - iq)(id + \frac{\lambda_f}{Ld}) - (\hat{id} - id)iq] + \hat{\omega}_\epsilon(0) \tag{A.18}$$

It is possible to improve the dynamic performance of the adaptive mechanism by introducing proportional regulation, claims Consequently, the optimum adaptive mechanism is provided by as

$$\hat{\omega_\epsilon} = (Kp + \frac{Ki}{s})[(\hat{iq} - iq)(id + \frac{\lambda_f}{Ld}) - (\hat{id} - id)iq]dt + \hat{\omega}_\epsilon(0) \tag{A.19}$$

Appendix B

Permanent Magnet Synchronous Motor (PMSM) Parameter Values

To simulate the PMSM model, it was necessary to determine its parameters, either by measurement or through suitable calculations. The simulation utilized values for parameters such as stator resistance, machine inductance, moment of inertia, viscous friction coefficient, the number of pole pairs, rated Speed, rated torque, and rotor flux, as detailed in Table B.1

Motor Parameters			
Parameter	Symbol	Value	SI Unit
Torque constant	T	10	Nm
Stator Resistance	R_s	2.875	Ω
d,q-axis Inductance	L_d, L_q	0.0085	H
Moment of Inertia	J	1e-3	Kgm^2
Friction Viscous Gain	B	0	Nms
Rotor Flux Constant	λ_f	0.1750	Wb
Number of Poles	P	8	nos
Rated Speed	N	1500	rpm

Table B.1: Permanent Magnet Synchronous Motor (PMSM) Parameter Values