APPLIED FUZZY LOGIC CONTROLS FOR IMPROVING DYNAMIC RESPONSE OF INDUCTION MACHINES

by

Altaf Ahmad Syed

Submitted in partial Fulfillment of the Requirements

for the Degree of

Master of Science in Engineering

in the

Electrical and Computer Engineering

Program

YOUNGSTOWN STATE UNIVERSITY

Youngtown, Ohio

August, 2008

APPLIED FUZZY LOGIC CONTROLS FOR IMPROVING

DYNAMIC RESPONSE OF INDUCTION MACHINES

Altaf Ahmad Syed

I hereby release this thesis to the public. I understand that this thesis will be made available from the OhioLINK ETD Center and the Maag Library Circulation Desk for public access. I also authorize the University or other individuals to make copies of this thesis as needed for scholarly research.

Signature:

_____

Altaf Ahmad Syed, Student Date

Approvals:

_____

Dr. Jalal Jalali, Thesis Adviser Date

_____

Dr. Philip Munro, Committee Member Date

_____

Dr. Faramarz Mossayebi, Committee Member Date

_____

Peter J. Kasvinsky, Dean of Graduate Studies & Research Date

ABSTRACT

This thesis presents a novel approach in control systems for improving the dynamic response of the induction machine. This approach leads to a better and improved control of the torque and current response of the induction machine when compared to the classical proportional-integral (PI) type controller with de-coupling terms. Mismatches in the actual parameters and the estimated parameters of the induction machine occur for several reasons such as: incorrect parameter estimation, changes in stator and rotor inductance due to saturation, stator and rotor resistance varying with temperature, etc. Under the classical approach, the de-coupling errors resulting from the parameter mismatches can become very large at higher machine rotational speeds. Under such conditions, the classical approach results in poor dynamic control of the torque and current response of the induction machine. Therefore, an advanced fuzzy logic controller is presented as a better alternative to the classical controller. The fuzzy logic-based *d-q* controller, based on its non-linear approach, provides robust control of the torque and current response of the induction machine even in the presence of mismatched parameters. Furthermore, the performance of the fuzzy logic controller is not dependent on the machine rotational speed. Using MATLAB-SIMULINK tools, the performance of the fuzzy controller is evaluated with mismatched machine parameters at various machine rotational speeds. The results show that the use of the fuzzy logic controller offers a superior control of the torque and current response of the induction machine, independent of the motor rotational speed when compared with the use of the classical controller.

ACKNOWLEDGEMENTS

I would like to acknowledge the great support Dr. Jalal Jalali provided me during my studies at Youngstown State University. I would also like to thank Dr. Philip Munro, Dr. Robert Foulkes and Dr. Duane Rost for the knowledge they passed onto me during my undergraduate studies at Youngstown State University. Finally, I would like to thank all my friends and family for providing the support I needed during my graduate studies.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

1. INTRODUCTION

A large percentage of current electrical machines found in industry are squirrel cage AC induction machines. These machines are widely used in industrial applications because they are less expensive and more rugged and reliable than DC motors [1]. The widespread use of these machines will continue due to their versatility, dependability and low cost. In the past, squirrel cage induction machines were limited to constant speed applications, and were operated from a fixed sinusoidal supply. The development of high power switching devices in the last decade has accelerated the growth in the market for variable speed drive systems incorporating AC induction machines and variable speed drives [1]. The robust construction of the squirrel cage induction machine together with high excitation frequency capabilities of the VSD (variable speed drive) allow for the operation of induction machines at very high rotational speeds [1].

In high performance applications that require precision speed and/or torque control, induction machines are frequently used [1]. Servo motor drives and spindle motor drives are examples of applications requiring precision speed control whereas electric vehicle drive systems require robust torque control.

An induction machine connected to a mechanical load and operated from a variable frequency drive, along with associated feedback sensors, constitutes an electric machine drive system [1]. In order to provide variable speed capability and control for an induction machine, a variable speed drive must include a variable frequency source (inverter) and a control system [1]. Inverters are DC to AC converters, whose power is supplied by a rectifier fed from the AC power line. Rectifiers draw distorted, non-

sinusoidal currents; hence passive or active filters are used on the input side to reduce the low frequency harmonic content in the supply current [1]. The control system of variable speed drives usually consists of microcontrollers, microprocessors and digital signal processors (DSPs) [1]. For precision control of induction machines, high performance control systems are usually employed. Such control systems use sensors of voltage, current, speed or position for feedback and usually require some knowledge of the induction machine parameters at steady-state.

Although induction machines have several advantageous characteristics such as simplicity, ruggedness and low cost, they also exhibit non-linear and time-varying dynamic behaviors [2]. When compared to a DC machine, the control system required for controlling the induction machine behavior is complicated, since the field and torque-producing components of the stator current are linked [1]. In order to achieve precise control of induction machines, it is necessary to control these two components of the stator current independently.

In the last decade, with the development of faster microcontrollers and DSPs, the field-oriented (vector) control technique has created a renaissance in modern high-performance control of pulse width modulated (PWM) inverter fed induction machines [1]. Today, field-oriented (vector) control is the most popular control method used in high performance industrial applications using the induction machine. Hasse in 1969, and Blaschke in 1972 first proposed the concept of field-orientation [1]. The objective of the field-orientation is to make the induction machine behave like a separately excited DC machine where torque and flux can be independently controlled [1][3]. Vector control schemes have allowed the induction machine to achieve torque control

performance similar to that of a separately excited DC machine and have led to the replacement of the DC machine by the induction machine in many high performance applications [3]. Furthermore, it has also resulted in faster transient response of the induction machine due to de-coupled control of torque and rotor flux.

There are two types of vector control methods; direct and indirect field-orientation. The direct field-orientation (DFO) method uses direct measurement of the air-gap flux vector by means of special search coils, or Hall Effect sensors embedded in the air gap. Even though this method results in accurate control, the application of this method is very limited, as induction machines equipped with flux measuring sensors are required [1]. In spite of its accuracy, this method degrades the motor's main advantages of mechanical simplicity and ease of maintenance [1]. In the indirect field-orientation (IFO), the rotor flux is estimated from the stator current vector, rotor speed and the machine estimated parameters. The drawback of this method is that it is sensitive to the variations in the induction machine parameters [4].

The objective of this research is to design a robust controller employing indirect field-oriented control to improve the dynamic control of the induction. The classical proportional-integral (PI-type) controller with de-coupling compensation method is most commonly used to implement the indirect field-oriented control due to its simple control algorithm and easy implementation [5][6]. This approach is based on a fixed model of the induction machine and requires accurate values of machine parameters. When there is a mismatch between the estimated induction machine parameters and the actual parameters, the performance of classical controller can deteriorate significantly. In applications requiring robust torque control, the sensitivity of the classical controller to

the machine parameter variations is even higher. Impact of parameter variations on various vector control schemes has been studied in great detail in the past [7][8] [9] [10]. The consequence of parameter mismatch between the values used in the controller and those in the machine is that the actual rotor flux position does not coincide with the position assumed by the controller. This leads to a loss of de-coupled flux and torque control of the induction machine. Performance of the controller therefore deteriorates from the desired. The problem of loss of de-coupled flux and torque control is even more profound at high-speed operation of the induction machine [4][5].

Various methods have been suggested to address this concern [4][5][7][11]. One such method is to provide the controller with accurate induction machine parameter values at all times. This means that the controller has to account for the several variables such as: changes in motor temperature, changes in the stator and rotor inductance due to saturation and high frequency, rotor skin effect, etc. Using classical methods, it is very difficult and computationally intensive to design a high performance induction machine drive system that takes all parameter variations into account. Other methods include use of internal model controller or additional PI controllers to compensate for the parametric variation errors. The method requiring an accurate model of induction machine is not practical while the use of internal model controller is not suited for variable speed operation [4].

The research presented in this study uses fuzzy logic-based non-linear approach for the design of a robust induction machine control system. The fuzzy logic control is a non-linear control technique and it allows for the control of non-linear system such as

induction machines. There are three main characteristics of a fuzzy logic controller [2][12]:

1) The fuzzy controller is a linguistic controller and does not require a precise and accurate mathematical model of the controlled object.
2) The fuzzy logic controller is an ideal flexible non-linear type controller as it can overcome the influence of non-linear variations.
3) The fuzzy logic controller provides robust control as it is insensitive to parametric variations of the controlled process.

The application of fuzzy logic in the control of induction machines has been well-documented [1][8][9][13][14][15]. It has been shown that the performance of the induction machine system as well as robustness to parameter variations can be improved by the use of fuzzy logic controller design [2][14][16]. However, these fuzzy logic-based designs exist for controlling induction machines to a desired speed, and no effective control system yet exists for the torque control of induction machine. In this thesis, an advanced fuzzy logic controller for an indirect field-oriented induction machine is presented. This fuzzy logic controller improves the torque and current response of induction machine and is robust to parameter variations. This advanced fuzzy logic controller can be used in high performance applications such as automotive drive systems, dynamometer test stands, high-speed spindles, etc.

In order to design an effective fuzzy logic controller and evaluate its performance, a detailed model of the induction machine and the induction control system is presented. Additionally, by the use of MATLAB-SIMULINK, extensive simulations of the complete system were performed to validate the robustness of the fuzzy controller. Using

MATLAB-SIMULINK, the performance of the fuzzy controller is evaluated with mismatched induction machine parameters under different operating conditions such as: step change in load in speed control mode, and step change in torque command. The simulation results from the fuzzy controller were then compared to that of the classical controller. The results show the superiority of the proposed advanced fuzzy controller over the classical controller in presence of the parameter mismatch and at high speeds. Results validate the robustness and effectiveness of the proposed fuzzy logic controller for controlling the dynamic torque and current response of the induction machine system.

This research is structured as follows:

- Chapter 2 provides the background information on the space vector transformations (*d-q* transformation), the dynamic equivalent circuit of the squirrel cage induction machine and the voltage source inverter model.
- In Chapter 3, the scalar and vector control techniques available for control of induction machines along with their advantages and disadvantages are presented. Furthermore, a detailed plant model of the induction machine for indirect field-oriented control (vector control) is created for use with MATLAB-SIMULINK.
- In Chapter 4, a fuzzy logic-based induction machine control system is presented for effective control of induction machines. The design of the fuzzy logic-based *d-q* controller used in this control system is also detailed in this chapter. Furthermore, the complete model of the fuzzy logic-based induction machine control system is implemented using MATLAB-SIMULINK.
- In Chapter 5, a complete simulation environment consisting of the operator inputs model, the fuzzy logic-based induction machine control system and the induction

machine plant model is presented. Using MATLAB-SIMULINK, various simulations of the complete induction machine system are performed using the fuzzy controller and the classical controller. The simulation results are then presented which compare the performance of the fuzzy controller to that of the classical controller.

- Chapter 6 provides the conclusions drawn from this research. Potential areas for further research that could result in further improvements of the fuzzy logic controller are also presented in this chapter.

2. DERIVATION OF INDUCTION MACHINE SYSTEM EQUATIONS

2.1. INTRODUCTION

DC motors have been traditionally used in the applications requiring high dynamic performance and precision speed and torque control. Examples of such applications include: high performance dynamometer systems, elevator motors, spindle drive motors and servomotors. Even though induction machines have been commonly used in industrial settings, their application was mostly limited to constant speed applications. With the recent advances in power electronics, several classes of variable speed drives have been developed that provide the AC induction machine with an edge over the DC machine. Because of simple, rugged and robust design of the induction machine, they can be operated at much higher speeds compared to the DC machine. The advances in the power electronics and digital control systems make induction machines an interesting solution for a wide range of applications. This increased interest in the induction machine has even led to its use in the automotive industry for fuel cell, hybrid-electric, and electric vehicles.

The other type of electric machines quickly gaining in popularity is AC permanent magnet machines. The most important benefits, which are expected when introducing permanent magnet machines, are lower losses and a higher torque density. Among the drawbacks are comparatively higher cost, increased temperature sensitivity, relatively complex controls and the need for sensors of flux. For high-speed capabilities, a fiber glass band is required for the retention of the surface mounted magnets [17].

Although the magnet-retaining band prevents the magnets from detaching, it degrades the heat dissipation of the magnets [17]. The induction machine on the other hand has a robust rotor construction, which makes it suitable for high-speed applications. The induction machine is also capable of providing a wider constant horsepower (HP) range compared to an AC permanent magnet machine [1][4][17]. And with new innovative cooling techniques, the new generation of induction machines also offer relatively higher torque density as well as higher efficiency. With the induction machines offering so many advantages over their rivals, their use in the industrial settings will continue to increase.

In an application where the goal is to optimize the system dynamic response and achieve high dynamic performance, a robust controller design is essential. Because of the non- linear nature of the induction machine, the controller must be able to control the non-linear behavior and provide a fast and accurate response. In this chapter, background information is provided on the concept of space vector transformations, induction machine equivalent circuit, and the voltage source inverter. The focus of this chapter is on deriving the dynamic equations for the induction machine and related components. The dynamic equations can then be used create a dynamic model of the induction machine.

2.2. INDUCTION MACHINE SYSTEM

Figure 2.1 shows the model of an AC induction machine system. The block diagram in Figure 2.1 shows the basic blocks involved in an induction machine system.

The five basic blocks are; voltage rectifier, input DC supply, inverter module, controller and an induction machine. The AC power source is typically a three-phase, 230V or 460V system. A rectifier is used to convert the three-phase AC voltage to provide the DC supply for the inverter. The input DC supply is usually placed in the form of a capacitor bank between the rectifier and the inverter. The inverter module along with the

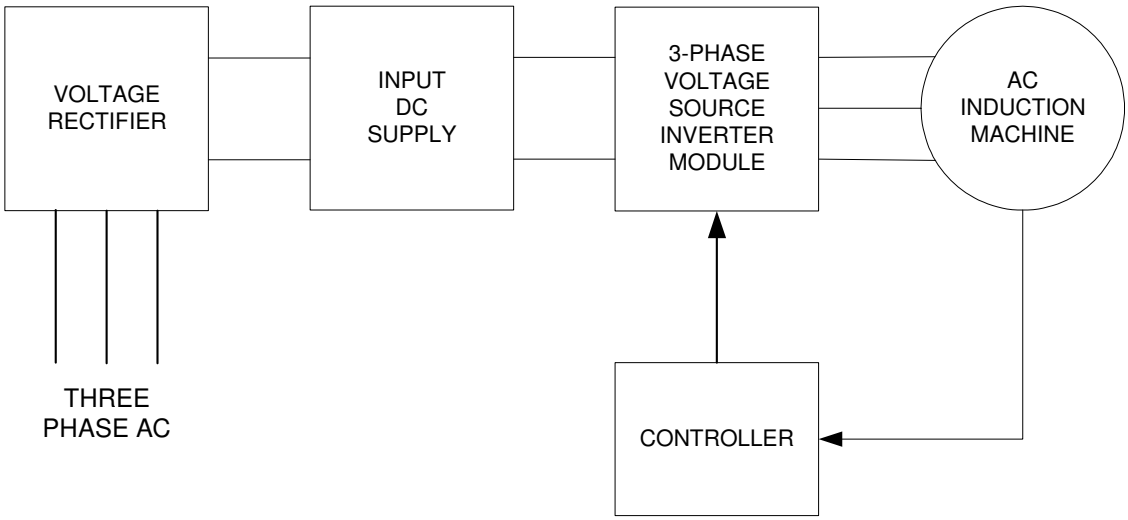


Figure 2.1. Block diagram of induction machine system

inputs from the controller converts the DC supply voltage (V_{DC}) to three-phase AC voltage. This three-phase AC voltage is sometimes optionally filtered and then drives the induction machine. As the inputs to the induction machine are three-phase AC voltages, the idea of a rotating direct and quadrature-axis frame (*d-q* frame) is introduced to transform the three-phase AC vectors into DC signals.

2.3. SPACE VECTOR TRANSFORMATIONS

Space vectors of three-phase variables, such as voltage, current, or flux, are very convenient for analysis and control of the induction machine system. Consider three-phase arbitrary vectors, $\hat{\Re}_{as}, \hat{\Re}_{bs}, \hat{\Re}_{cs}$, whose total vector sum, $\hat{\Re}_s$, can be represented as follows [1][3]:

$$\hat{\Re}_s = \hat{\Re}_{as} + \hat{\Re}_{bs} + \hat{\Re}_{cs} = \Re_{as} + \Re_{bs} e^{j\frac{2}{3}\pi} + \Re_{cs} e^{j\frac{4}{3}\pi} \tag{2.1}$$

Since $e^{j\frac{2}{3}\pi} = -\frac{1}{2} + j\frac{\sqrt{3}}{2}$, and $e^{j\frac{4}{3}\pi} = -\frac{1}{2} - j\frac{\sqrt{3}}{2}$, eq. (2.1) can be rewritten as follows:

$$\hat{\Re}_s = \Re_{as} - \frac{1}{2}\Re_{bs} - \frac{1}{2}\Re_{cs} + j\left(\frac{\sqrt{3}}{2}\Re_{bs} - \frac{\sqrt{3}}{2}\Re_{cs}\right) \tag{2.2}$$

As an example, the sum of three-phase currents is zero, which implies that one of the currents can be eliminated and therefore one degree of freedom can be reduced and space vectors can be represented in an equivalent two-phase space vectors. Using such an analogy the sum of the vectors can be represented as follows:

$$\hat{\Re}_s = \Re^s_{ds} + j\Re^s_{qs} = \Re_{as} - \frac{1}{2}\Re_{bs} - \frac{1}{2}\Re_{cs} + j\left(\frac{\sqrt{3}}{2}\Re_{bs} - \frac{\sqrt{3}}{2}\Re_{cs}\right) \tag{2.3}$$

where superscript *s* denotes a stationary direct-quadrature axis (*d*-*q*) frame. Equation (2.3) explains the *abc* to dq^s transformation. The transformation can be rewritten as follows [1]:

$$\begin{bmatrix} \Re^s_{ds} \\ \Re^s_{qs} \end{bmatrix} = \begin{bmatrix} 1 & -\frac{1}{2} & -\frac{1}{2} \\ 0 & \frac{\sqrt{3}}{2} & -\frac{\sqrt{3}}{2} \end{bmatrix} \begin{bmatrix} \Re_{as} \\ \Re_{bs} \\ \Re_{cs} \end{bmatrix} \tag{2.4}$$

and

$$\begin{bmatrix} \Re_{as} \\ \Re_{bs} \\ \Re_{cs} \end{bmatrix} = \begin{bmatrix} \frac{2}{3} & 0 \\ -\frac{1}{3} & \frac{1}{\sqrt{3}} \\ -\frac{1}{3} & -\frac{1}{\sqrt{3}} \end{bmatrix} \begin{bmatrix} \Re^{s}_{ds} \\ \Re^{s}_{qs} \end{bmatrix} \tag{2.5}$$

Transformations in equations (2.4) and (2.5) apply to all three-phase variables of any three-phase system, which add up to zero.

In a balanced system, where the sum of all the three vectors is zero, the magnitude of $\hat{\Re}_s$ is $\frac{3}{2}$ higher than the magnitude (peak value), $\Re_{as}$, of phase vectors. Hence the equation (2.4) can be multiplied by $\frac{2}{3}$ and equation (2.5) can be multiplied by $\frac{3}{2}$.

The final transformation can therefore be written as follows [1][3]:

$$\begin{bmatrix} \Re^{s}_{d} \\ \Re^{s}_{q} \end{bmatrix} = \frac{2}{3} \begin{bmatrix} 1 & -\frac{1}{2} & -\frac{1}{2} \\ 0 & \frac{\sqrt{3}}{2} & -\frac{\sqrt{3}}{2} \end{bmatrix} \begin{bmatrix} \Re_{a} \\ \Re_{b} \\ \Re_{c} \end{bmatrix} = \begin{bmatrix} \frac{2}{3} & -\frac{1}{3} & -\frac{1}{3} \\ 0 & \frac{1}{\sqrt{3}} & -\frac{1}{\sqrt{3}} \end{bmatrix} \begin{bmatrix} \Re_{a} \\ \Re_{b} \\ \Re_{c} \end{bmatrix} = T \begin{bmatrix} \Re_{a} \\ \Re_{b} \\ \Re_{c} \end{bmatrix} \tag{2.6}$$

and,

$$\begin{bmatrix} \Re_{a} \\ \Re_{b} \\ \Re_{c} \end{bmatrix} = \frac{3}{2} \begin{bmatrix} \frac{2}{3} & 0 \\ -\frac{1}{3} & \frac{1}{\sqrt{3}} \\ -\frac{1}{3} & -\frac{1}{\sqrt{3}} \end{bmatrix} \begin{bmatrix} \Re^{s}_{d} \\ \Re^{s}_{q} \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ -\frac{1}{2} & \frac{\sqrt{3}}{2} \\ -\frac{1}{2} & -\frac{\sqrt{3}}{2} \end{bmatrix} \begin{bmatrix} \Re^{s}_{d} \\ \Re^{s}_{q} \end{bmatrix} = T' \begin{bmatrix} \Re^{s}_{d} \\ \Re^{s}_{q} \end{bmatrix} \tag{2.7}$$

where,

$$T = \begin{bmatrix} \frac{2}{3} & -\frac{1}{3} & -\frac{1}{3} \\ 0 & \frac{1}{\sqrt{3}} & -\frac{1}{\sqrt{3}} \end{bmatrix} \tag{2.8}$$

and

$$T' = \begin{bmatrix} 1 & 0 \\ -\frac{1}{2} & \frac{\sqrt{3}}{2} \\ -\frac{1}{2} & -\frac{\sqrt{3}}{2} \end{bmatrix} \tag{2.9}$$

2.4. THREE-PHASE VOLTAGE SOURCE INVERTER

In this section, a three-phase voltage source inverter module is modeled in detail, which converts the DC source voltage to three-phase AC voltage using switching control signals. Figure 2.2 shows a detailed model, which contains ideal switches and diodes [1][18]. Control inputs (S_a, S_b, S_c) are switching functions of time {0,1}.

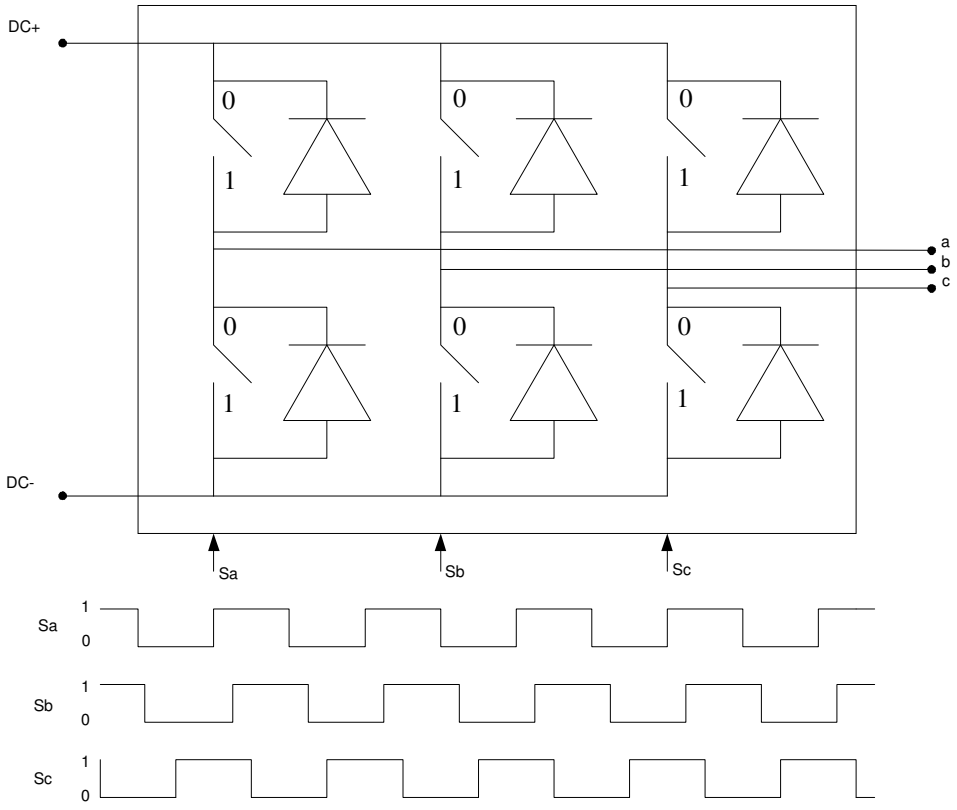


Figure 2.2. VSI Switching model

The control inputs (S_a, S_b, S_c) switch (0 for open position and 1 for closed position) at different intervals or average duty cycles (d_a, d_b, d_c), to produce an average three-phase AC signal. Therefore the average model can be written as follows:

$$\begin{bmatrix} V_a \\ V_b \\ V_c \end{bmatrix} = \begin{bmatrix} d_a \\ d_b \\ d_c \end{bmatrix} kV_{DC} \tag{2.10}$$

where V_a, V_b, and V_c are three-phase AC voltage outputs, considering the Y-connected load, V_{DC} is the DC source voltage, d_a, d_b, and d_c are phase duty cycle of inverter, and k is a constant which depends on the nature of the control signal modulation. A value of ½ is used for k, when the nature of modulation is pure sinusoidal, and a value of $\frac{1}{\sqrt{3}}$ is used when the nature of modulation is source voltage modulation. Line to line voltages and currents can be expressed as follows:

$$\begin{bmatrix} V_{ab} \\ V_{bc} \\ V_{ca} \end{bmatrix} = \begin{bmatrix} V_a - V_b \\ V_b - V_c \\ V_c - V_a \end{bmatrix} \tag{2.11}$$

$$\begin{bmatrix} d_{ab} \\ d_{bc} \\ d_{ca} \end{bmatrix} = \begin{bmatrix} d_a - d_b \\ d_b - d_c \\ d_c - d_a \end{bmatrix} \tag{2.12}$$

$$\begin{bmatrix} V_{ab} \\ V_{bc} \\ V_{ca} \end{bmatrix} = \begin{bmatrix} d_{ab} \\ d_{bc} \\ d_{ca} \end{bmatrix} kV_{DC} \tag{2.13}$$

$$I_{DC} = \begin{bmatrix} d_a & d_b & d_c \end{bmatrix} \begin{bmatrix} I_a \\ I_b \\ I_c \end{bmatrix} \tag{2.14}$$

Equations (2.10) through (2.14) represent the three-phase average model of the voltage source inverter module, and are shown in Figure 2.3 [18].

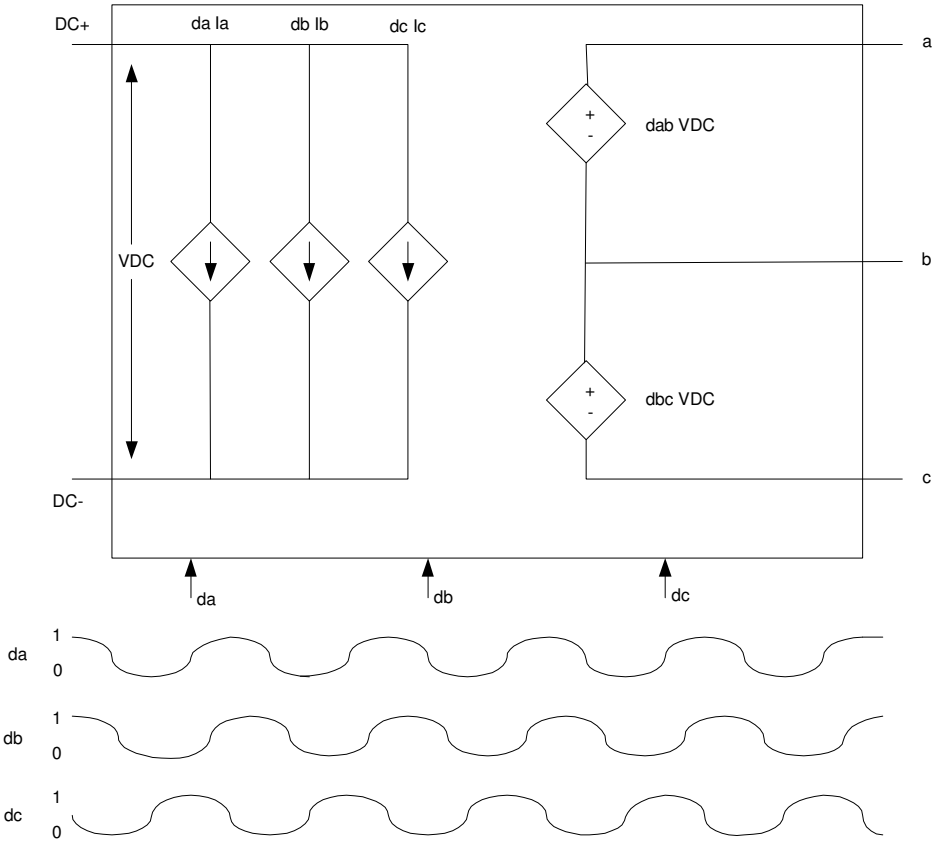


Figure 2.3. Average model of voltage source inverter

The direct and quadrature (*d-q*) components of the phase duty cycle can be defined by applying the transformation in equation (2.6). Using space vector transformations, the phase voltages can be written as follows:

$$\begin{bmatrix} V_d \\ V_q \end{bmatrix} = T \begin{bmatrix} V_a \\ V_b \\ V_c \end{bmatrix} \tag{2.15}$$

Since DC source voltage is used as input, we can define the phase duty cycle *d-q* components as follows:

$$\begin{bmatrix} d_d \\ d_q \end{bmatrix} = T \begin{bmatrix} d_a \\ d_b \\ d_c \end{bmatrix} \tag{2.16}$$

Finally by substituting equations (2.10) and (2.15) into equation (2.16), the relationship between *d-q* voltages and DC source voltage can be written as:

$$\begin{bmatrix} V_d \\ V_q \end{bmatrix} = \begin{bmatrix} d_d \\ d_q \end{bmatrix} kV_{DC} \tag{2.17}$$

The *d-q* to three-phase transformation for currents and duty cycle can be written as follows:

$$\begin{bmatrix} I_a \\ I_b \\ I_c \end{bmatrix} = T' \begin{bmatrix} I_d \\ I_q \end{bmatrix} \tag{2.18}$$

$$\begin{bmatrix} d_a \\ d_b \\ d_c \end{bmatrix} = T' \begin{bmatrix} d_d \\ d_q \end{bmatrix} \tag{2.19}$$

Substituting equations (2.16) and (2.18) into equation (2.14), and after some matrix calculations, the necessary current relationship can be written as follows:

$$I_{DC} = \frac{3}{2} \begin{bmatrix} d_d & d_q \end{bmatrix} \begin{bmatrix} I_d \\ I_q \end{bmatrix} \tag{2.20}$$

In this manner the complete three-phase average voltage source inverter can be modeled in stationary *d-q* coordinate space as shown in Figure 2.4. Finally, the relationship between the DC power and the *d-q* components of voltage and current can be written as follows:

$$P_{DC} = \frac{3}{2} \left(V_d I_d + V_q I_q \right) \tag{2.21}$$

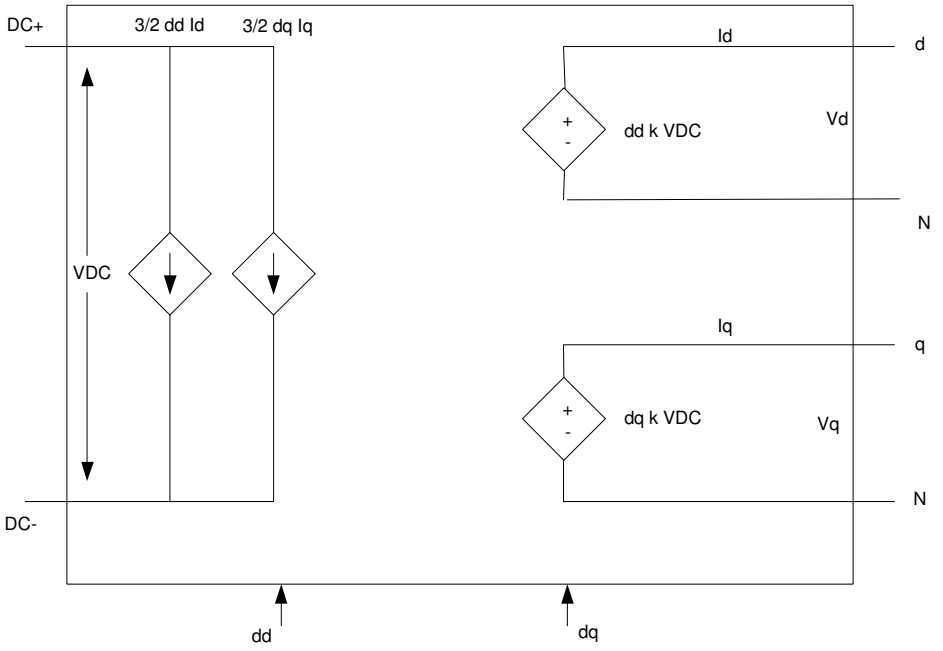


Figure 2.4. VSI average *d-q* model

This model can now be used to determine the DC power required by the inverter module to meet the load demand. In applications such as electric and hybrid electric vehicles, the battery supplies the necessary DC power to the inverter module. In such a case, the implementation of this model can show the amount of DC power consumed by the induction machine and the inverter model to meet the load demand.

2.5. SQUIRREL CAGE INDUCTION MACHINE

The most common form of induction machine used in industrial applications is the squirrel cage induction machine [1]. The other form is wound rotor induction machine. Wound rotor motors are mostly operated on a fixed 60 Hz utility power and are mostly used to limit inrush current and/or achieve high starting torque [1][19]. This

research focuses solely on the squirrel cage induction machine model. The two main components of a squirrel cage induction machine are stator and rotor. The stator has a poly-phase winding set in the slots of the laminated iron core. The rotor does not have any brushes or slip rings. In fact, aluminum or copper bars embedded in the rotor slots form the rotor windings. The rotor bars are short-circuited on each end through end rings. This type of rotor construction looks like a squirrel cage, and is hence called a squirrel cage induction machine. Both the rotor and stator cores are cylindrical. With the slot effects neglected, the air-gap is uniform between the stator and the rotor.

The set of poly-phase currents in the stator windings produces a rotating magnetic field. The windings of the stator are distributed so as to produce a close approximation to a sinusoidal space distribution of magneto-motive force (mmf). The rotor bars experience this field and thus voltage is induced in the rotor conductors. Since these rotor bars are short-circuited, the induce voltage causes current to flow. In turn, the rotor mmf produces a magnetic flux pattern, which also rotates in the air gap at the same speed as that of stator. There is a torque that tends to align the magnetic fields, which results in the rotor moving in the same direction as the magnetic fields. The rotor then accelerates to a speed at which the electromagnetic torque is balanced by the load torque.

Figure 2.5 shows the steady-state equivalent circuit (with core loss neglected) for the analysis and design of induction machine [1][18][19]. In the equivalent circuit, all rotor parameters are referred to the stator. For the equivalent circuit of Figure 2.5, R_s is the stator resistance, R_r is the rotor resistance, L_m is the magnetizing inductance, L_{1s} is the

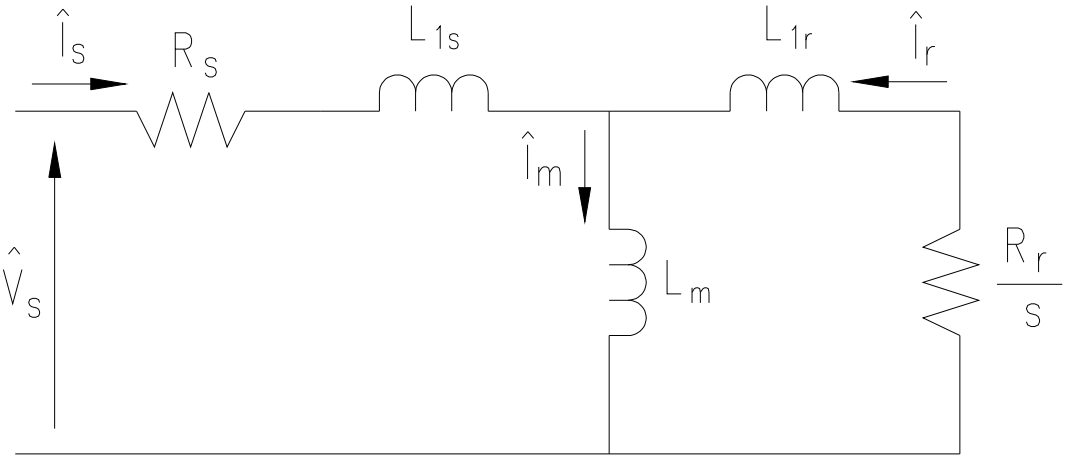


Figure 2.5. Steady-state equivalent circuit of an induction machine

stator leakage inductance and L_{1r} is the rotor leakage inductance and s is the slip. Two additional quantities, stator and rotor inductance, are now defined as:

$$L_s = L_{1s} + L_m \qquad \text{and} \qquad L_r = L_{1r} + L_m \tag{2.22}$$

Unlike synchronous machines, induction machines do not operate at synchronous speed. At rated speed, the speed of the rotor is slightly (2-7%) less than the synchronous speed. If the excitation frequency is ω and the actual rotor speed in electrical frequency unit is ω_r, then the slip is given as:

$$s = \frac{\omega - \omega_r}{\omega} = \frac{\omega_{sl}}{\omega} \tag{2.23}$$

where ω_{sl} is the rotor slip frequency. The electrical power transferred across the air gap from the stator is given as:

$$P_e = 3R_r \frac{\omega}{\omega_{sl}} I_r^2 \tag{2.24}$$

The developed torque, determined by subtracting the I^2R losses from electrical power and dividing the resultant power by the rotor speed, is given as:

$$T_e = 3P_p R_r \frac{I_r^2}{\omega_{sl}} \tag{2.25}$$

where P_p is the number of pole pairs.

Although traditional per phase equivalent circuit has been used in steady-state analysis, it is not appropriate for evaluation of dynamic performance of the induction machine [1]. Simply stated, it can not explain the dynamic performance of the induction machine. In order to understand and analyze the transient behavior of the induction machine, the dynamic equivalent circuit of the induction machine must be used.

The three-phase stator windings in an induction machine are designed to produce sinusoidal mmf in the space along the air-gap periphery. Assuming uniform air-gap and neglecting the effects of slot harmonics, the magnetic flux distribution will also be sinusoidal. For such machines the space vector transformation of section 2.4 can then be used. For a sinusoidal three-phase quantity, the corresponding space vector is a constant magnitude vector rotating at the angular speed, ω, imposed by the supply source. With space vector notation, voltage equations for the stator and rotor circuits of induction machine are given as [1]:

$$\hat{V}_s^s = R_s \hat{I}_s^s + \frac{d}{dt}\hat{\lambda}_s \tag{2.26}$$

and

$$\hat{V}_r^{'} = R_r^{'} \hat{I}_r^{'} + \frac{d}{dt}\hat{\lambda}_r = 0 \tag{2.27}$$

where $\hat{\lambda}_s$ is the stator flux, and $\hat{\lambda}_r$ is the rotor flux.

Just like in the steady-state equivalent circuit, it is very convenient to refer the rotor quantities to the stator. For that the $\frac{d}{dt}$ term must be replace by $\frac{d}{dt} - j\omega_r$ in the rotor equation. Hence,

$$0 = R_r^s \hat{I}_r^s + \left(\frac{d}{dt} - j\omega_r\right)\hat{\lambda}_r \tag{2.28}$$

where ω_r is the rotor angular speed in electrical frequency. The stator and rotor fluxes are related to the stator and rotor current as:

$$\begin{bmatrix} \hat{\lambda}_s \\ \hat{\lambda}_r \end{bmatrix} = \begin{bmatrix} L_s & L_m \\ L_m & L_r \end{bmatrix} \begin{bmatrix} \hat{I}_s \\ \hat{I}_r \end{bmatrix} \tag{2.29}$$

In algebraic form, the stator and rotor flux are given as:

$$\hat{\lambda}_s = L_s \hat{I}_s + L_m \hat{I}_r \tag{2.30}$$

and

$$\hat{\lambda}_r = L_m \hat{I}_s + L_r \hat{I}_r \tag{2.31}$$

Equations (2.26), (2.28) and (2.29) constitute the dynamic equivalent circuit of the induction machine on a stationary reference frame. Using equations (2.30) and (2.31), equations (2.26) and (2.28) can be represented as:

$$\hat{V}_s^s = R_s \hat{I}_s^s + L_s \frac{d}{dt}\hat{I}_s^s + L_m \frac{d}{dt}\hat{I}_r^s \tag{2.32}$$

and

$$0 = R_r \hat{I}_r^s + L_r \frac{d}{dt}\hat{I}_r^s + L_m \frac{d}{dt}\hat{I}_s^s - j\omega_r (L_r \hat{I}_r^s + L_m \hat{I}_s^s) \tag{2.33}$$

The dynamic equivalent circuit obtained from equations (2.32) and (2.33) is shown in Figure 2.6.

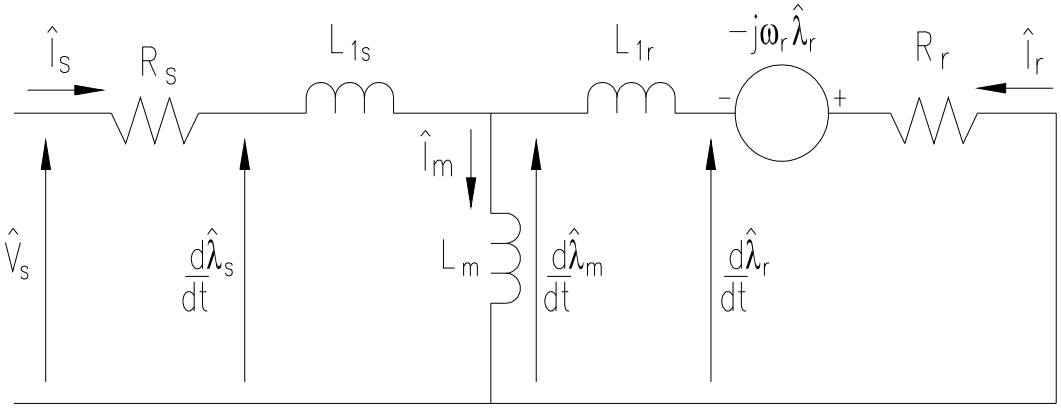


Figure 2.6. Dynamic equivalent circuit for a stator reference frame

Figure 2.7 shows an ideal induction machine. In steady-state, space vectors of motor variables revolve in the stator frame of reference with angular speed, ω, imposed by the supply source (inverter). It must be noted that this speed does not depend on the number of pole pairs, P_p. Under transient operating conditions, instantaneous speeds of the space vectors vary, and they are not necessarily the same for all vectors, but the vectors keep revolving nevertheless. Consequently, their *d* and *q* components are AC variables. Therefore, in addition to the static, three-phase to dq^s transformation (equation 2.6) and dq^s to three-phase transformation (equation 2.7), the dynamic, three-phase to *dq* can be represented in terms of the electrical angle, θ, between the rotor direct-axis and the stator phase a-axis as:

$$\begin{bmatrix} \Re_d \\ \Re_q \end{bmatrix} = \frac{2}{3}\begin{bmatrix} \cos\theta & \cos(\theta-120) & \cos(\theta+120) \\ -\sin\theta & -\sin(\theta-120) & -\sin(\theta+120) \end{bmatrix}\begin{bmatrix} \Re_a \\ \Re_b \\ \Re_c \end{bmatrix} \tag{2.34}$$

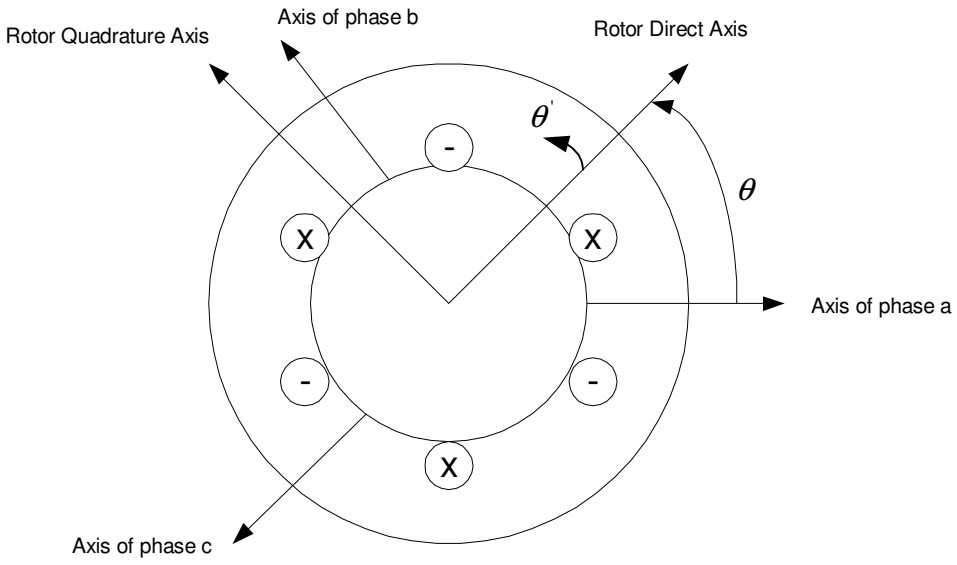


Figure 2.7. Ideal induction machine

Usually, a revolving reference frame is selected so that it moves in synchronism with a selected vector. The *d-q* components of the space vector in the revolving frame are now DC signals, constant in steady-state and varying in transient states. Considering the same space vector and a rotating *d-q* reference frame revolving at angular speed, ω_e, its stationary dq^s to rotating *dq* transformation can be written as:

$$\begin{bmatrix} \Re_d \\ \Re_q \end{bmatrix} = \begin{bmatrix} \cos(\omega_e t) & \sin(\omega_e t) \\ -\sin(\omega_e t) & \cos(\omega_e t) \end{bmatrix} \begin{bmatrix} \Re_d^s \\ \Re_q^s \end{bmatrix} \tag{2.35}$$

and the inverse, rotating *dq* to stationary dq^s transformation is given as:

$$\begin{bmatrix} \Re_d^s \\ \Re_q^s \end{bmatrix} = \begin{bmatrix} \cos(\omega_e t) & -\sin(\omega_e t) \\ \sin(\omega_e t) & \cos(\omega_e t) \end{bmatrix} \begin{bmatrix} \Re_d \\ \Re_q \end{bmatrix} \tag{2.36}$$

This transformation is known as Park Transformation [3].

The dynamic equivalent circuit of the induction machine for an arbitrary reference frame rotating at angular speed, ω_e, is now presented. Motor equations in a reference

frame rotating at ω_e can be obtained by replacing the $\frac{d}{dt}$ term by $\frac{d}{dt} + j\omega_e$ in equations 2.32 and 2.33 [1]. Thus:

$$\hat{V}_s^e = R_s \hat{I}_s^e + L_s\left(\frac{d}{dt} + j\omega_e\right)\hat{I}_s^e + L_m\left(\frac{d}{dt} + j\omega_e\right)\hat{I}_r^e \tag{2.37}$$

and

$$0 = R_r \hat{I}_r^e + L_r\left(\frac{d}{dt} + j\omega_e\right)\hat{I}_r^e + L_m\left(\frac{d}{dt} + j\omega_e\right)\hat{I}_s^e - j\omega_r(L_r\hat{I}_r^e + L_m\hat{I}_s^e) \tag{2.38}$$

where superscript *e* designates an arbitrary rotating frame.

The equations above can be simplified as:

$$\hat{V}_s^e = R_s\hat{I}_s^e + L_s\frac{d}{dt}\hat{I}_s^e + L_m\frac{d}{dt}\hat{I}_r^e + j\omega_e L_s\hat{I}_s^e + j\omega_e L_m\hat{I}_r^e$$

(2.39)

and

$$0 = R_r\hat{I}_r^e + L_r\frac{d}{dt}\hat{I}_r^e + +L_m\frac{d}{dt}\hat{I}_s^e + j(\omega_e - \omega_r)L_r\hat{I}_r^e + j(\omega_e - \omega_r)L_m\hat{I}_s^e \tag{2.40}$$

The new flux linkage terms are defined as:

$$\hat{\lambda}_s^e = L_s\hat{I}_s^e + L_m\hat{I}_r^e \tag{2.41}$$

and

$$\hat{\lambda}_r^e = L_m\hat{I}_s^e + L_r\hat{I}_r^e \tag{2.42}$$

The dynamic equivalent circuit of the induction machine for an arbitrary rotating reference frame is given in Figure 2.8.

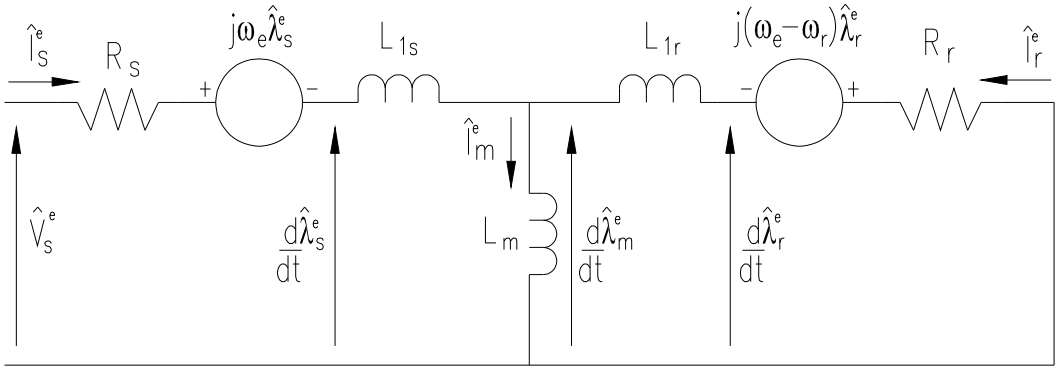


Figure 2.8. Dynamic equivalent circuit for an arbitrary rotating frame

Resolving the voltage and current space vectors into their respective *d-q* components, equations (2.39) and (2.40) can be expressed in matrix form as follows:

$$\begin{bmatrix} V_{ds}^e \\ V_{qs}^e \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} R_s + L_s\frac{d}{dt} & -\omega_e L_s & L_m\frac{d}{dt} & -\omega_e L_m \\ \omega_e L_s & R_s + L_s\frac{d}{dt} & \omega_e L_m & L_m\frac{d}{dt} \\ L_m\frac{d}{dt} & -(\omega_e-\omega_r)L_m & R_r + L_r\frac{d}{dt} & -(\omega_e-\omega_r)L_r \\ (\omega_e-\omega_r)L_m & L_m\frac{d}{dt} & (\omega_e-\omega_r)L_r & R_r + L_r\frac{d}{dt} \end{bmatrix} \begin{bmatrix} I_{ds}^e \\ I_{qs}^e \\ I_{dr}^e \\ I_{qr}^e \end{bmatrix} \tag{2.43}$$

The equations for the stationary reference frame can also be obtained by simply substituting, $\omega_e = 0$ in equation (2.43) [10]:

$$\begin{bmatrix} V_{ds}^s \\ V_{qs}^s \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} R_s + L_s\frac{d}{dt} & 0 & L_m\frac{d}{dt} & 0 \\ 0 & R_s + L_s\frac{d}{dt} & 0 & L_m\frac{d}{dt} \\ L_m\frac{d}{dt} & \omega_r L_m & R_r + L_r\frac{d}{dt} & \omega_r L_r \\ -\omega_r L_m & L_m\frac{d}{dt} & -\omega_r L_r & R_r + L_r\frac{d}{dt} \end{bmatrix} \begin{bmatrix} I_{ds}^s \\ I_{qs}^s \\ I_{dr}^s \\ I_{qr}^s \end{bmatrix} \tag{2.44}$$

For the *d-q* frame rotating at synchronous speed, ω, the dynamic equations are written in a matrix form as [13]:

$$\begin{bmatrix} V_{ds} \\ V_{qs} \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} R_s + L_s \frac{d}{dt} & -\omega L_s & L_m \frac{d}{dt} & -\omega L_m \\ \omega L_s & R_s + L_s \frac{d}{dt} & \omega L_m & L_m \frac{d}{dt} \\ L_m \frac{d}{dt} & -\omega_{sl} L_m & R_r + L_r \frac{d}{dt} & -\omega_{sl} L_r \\ \omega_{sl} L_m & L_m \frac{d}{dt} & \omega_{sl} L_r & R_r + L_r \frac{d}{dt} \end{bmatrix} \begin{bmatrix} I_{ds} \\ I_{qs} \\ I_{dr} \\ I_{qr} \end{bmatrix} \tag{2.45}$$

where, $\omega_{sl} = \omega - \omega_r$, is the rotor electrical slip frequency. Based on equation (2.45), the *d-q* dynamic equivalent circuit for a synchronously rotating *d-q* frame can be created as shown in Figure 2.9. With the flux linkages defined as,

$$\lambda_{qs} = L_s I_{qs} + L_m I_{qr} \tag{2.46}$$

$$\lambda_{ds} = L_s I_{ds} + L_m I_{dr} \tag{2.47}$$

$$\lambda_{qr} = L_m I_{qs} + L_r I_{qr} \tag{2.48}$$

$$\lambda_{dr} = L_m I_{qs} + L_r I_{dr} \tag{2.49}$$

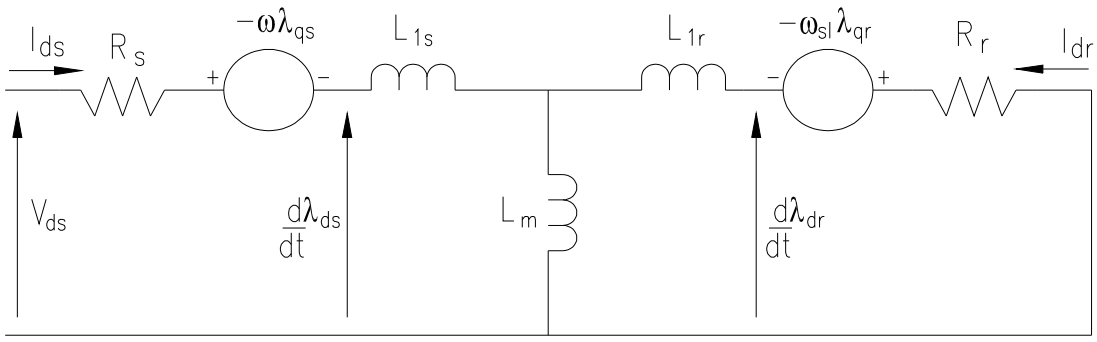
the induction machine dynamic equations are presented in the algebraic form as:

$$V_{qs} = R_s I_{qs} + \frac{d}{dt}\lambda_{qs} + \omega\lambda_{ds} \tag{2.50}$$

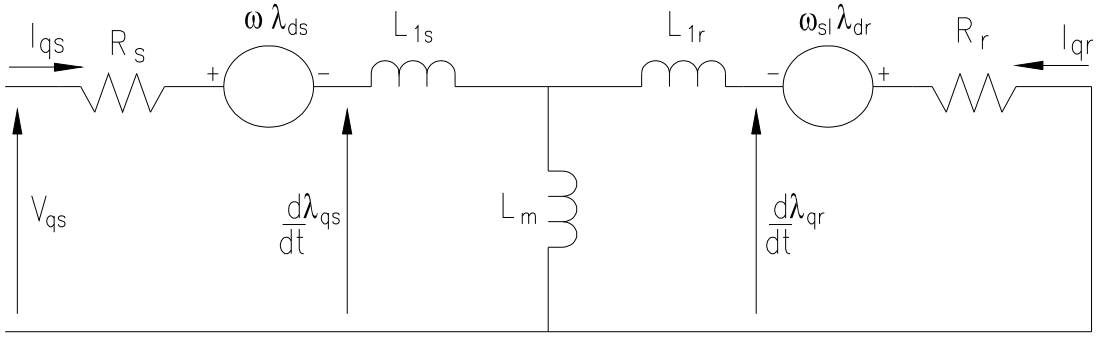
$$V_{ds} = R_s I_{ds} + \frac{d}{dt}\lambda_{ds} - \omega\lambda_{qs} \tag{2.51}$$

$$0 = R_r I_{qr} + \frac{d}{dt}\lambda_{qr} + \omega_{sl}\lambda_{dr} \tag{2.52}$$

$$0 = R_r I_{dr} + \frac{d}{dt}\lambda_{dr} + \omega_{sl}\lambda_{qr} \tag{2.53}$$



(a) d-axis Equivalent Circuit



(b) q-axis Equivalent Circuit

Figure 2.9. *d-q* dynamic equivalent circuit for a synchronously rotating frame

The relationship between the *d-q* voltage components and the actual phase voltages is given as:

$$\begin{bmatrix} V_d \\ V_q \end{bmatrix} = \frac{2}{3}\begin{bmatrix} \cos\theta & \cos(\theta-120) & \cos(\theta+120) \\ -\sin\theta & -\sin(\theta-120) & -\sin(\theta+120) \end{bmatrix}\begin{bmatrix} V_a \\ V_b \\ V_c \end{bmatrix} \tag{2.54}$$

where V_a, V_b and V_c are the actual phase voltages at any give time, t.

The general expression for electrical power can be expressed as follows:

$$P_e = \frac{3}{2}\left(V_d I_d + V_q I_q\right) \tag{2.55}$$

The developed torque can be calculated in several forms as follows [1][2][13]:

$$T_e = \frac{3}{2} p_p \, \mathrm{Im}\{\hat{i}_s \, \hat{\lambda}_s^*\} = \frac{3}{2} p_p \left(i_{qs} \lambda_{ds} - i_{ds} \lambda_{qs} \right) \tag{2.56}$$

$$T_e = \frac{3}{2} p_p \frac{L_m}{L_r} \mathrm{Im}\{\hat{i}_s \, \hat{\lambda}_r^*\} = \frac{3}{2} p_p \frac{L_m}{L_r} \left(i_{qs} \lambda_{dr} - i_{ds} \lambda_{qr} \right) \tag{2.57}$$

or

$$T_e = \frac{3}{2} p_p L_m \, \mathrm{Im}\{\hat{i}_s \, \hat{i}_r^*\} = \frac{3}{2} p_p L_m \left(i_{qs} i_{dr} - i_{ds} i_{qr} \right) \tag{2.58}$$

It can be seen that the torque equations are all non-linear, and they include a difference of product of two motor variables.

The mechanical angular speed can be calculated from the torque equation as follows:

$$T_e - T_{load} - T_{fr} = J_{eff} \frac{d\omega_m}{dt} \tag{2.59}$$

where T_{load} is the load torque, T_{fr} represents the friction, J_{eff} is the total effective inertia, and ω_m is the rotor angular speed.

The rotor electrical speed, ω_r, can be determined from the rotor angular speed as:

$$\omega_r = P_p \omega_m \tag{2.60}$$

Equations (2.46) through (2.60) can be used to effectively model a squirrel cage induction machine in a simulation environment. In the next chapter, the vector control methods (field-oriented control) for the control of the induction machine are briefly presented. The dynamic equations are then adapted for a rotor field-oriented induction machine. Finally, an induction machine plant model consisting of the induction machine dynamic *d-q* model, voltage *d-q* transformation block and current inverse *d-q* transformation block is created.

3. INDUCTION MACHINE PLANT MODELING AND SIMULATION ENVIRONMENT

3.1. INTRODUCTION

In Chapter 2, the dynamic equations governing the behavior of induction machines were developed. As can be seen from the dynamic equations of the induction machine, its behavior is non-linear and thus requires a robust control scheme for controlling its torque and current response. In this chapter, the scalar and vector control schemes available for induction machines control, along with their advantages and disadvantages are briefly presented. Using the vector control scheme for induction machine control, the dynamic equations are obtained for a rotor field-oriented induction machine. Finally, a plant model consisting of the induction machine dynamic *d-q* model, the voltage *d-q* transformation block and current *d-q* inverse transformation block is created. The induction machine plant model is then used in a simulation environment using MATLAB-SIMULINK.

3.2. SCALAR CONTROLS

Of all the control schemes, scalar controls are perhaps the easiest and the simplest to accomplish. The two types of scalar controls are briefly described below.

3.2.1. Open-Loop Scalar Speed Control (constant V/Hz)

In this method, the approach is to approximately keep the stator flux constant in the machine regardless of the excitation frequency [18][20]. To maintain the flux at a constant rated level, the stator voltage is adjusted in proportion to the supply frequency. This is the simplest approach and is commonly referred to as constant volts/Hz method. This method also does not require any feedback. For low speed operation, the stator voltage drop across the stator resistance must be taken into account for maintaining constant flux. A user is typically required to input a stator voltage boost value during the initial commissioning of the variable frequency drive. At speeds above the base-speed, the motor operates in field-weakening region as the stator voltage can not be increased any further than the machine's rated voltage. Clearly, accurate speed control is not possible, because the actual slip varies with load and is not accounted for. In some control algorithms, a fixed value of slip at some load is added to the reference speed in order to achieve target speed. However, precision speed control is still not possible because slip variation is not taken into account. The open loop scalar control drives are still very popular and widely used in low-performance applications that do not require precision control such as pumps, fans, or grinders.

3.2.2. Closed-Loop Scalar Speed Control

In this scheme, the motor speed is monitored and compared with the reference speed by means of an encoder or other speed-measuring device. The speed error signal is applied to a PI type slip controller, which generates a new reference speed [20].

However, scalar controls only attempt to control the magnitude of the variables and thus are unable to provide fast dynamic control. With the advancement in vector control schemes, scalar closed-loop methods of speed control have now become obsolete [1].

3.3. VECTOR CONTROLS

Compared to scalar controls, which only involve controlling the magnitude of the control variables, a vector or field orientated control involves adjusting the magnitude and phase alignment of the vector quantities of the motor. Field-oriented control is the most popular high performance control technique for the AC induction machine. Progress in the field of power electronics has enabled the application of induction machines for high performance drives, where traditionally only DC motors were utilized [1]. With sophisticated control methods, AC induction drives offer the same control capabilities as high performance four-quadrant DC drives. The high performance AC drives allow vector control of the induction machine running in a closed loop with speed/position sensor providing the required feedback.

In the preceding chapter, the dynamic *d-q* equivalent circuit of the induction machine was presented. Using field-oriented control, the objective is to achieve de-coupled control of flux and torque so that a fast and accurate transient response can be achieved. The torque equation for the induction machine developed in the earlier chapter is given as:

$$T_e = \frac{3}{2} p_p \frac{L_m}{L_r} \mathrm{Im}\left\{\hat{i}_s \, \hat{\lambda}_r^*\right\} = \frac{3}{2} p_p \frac{L_m}{L_r} \left(i_{qs}\lambda_{dr} - i_{ds}\lambda_{qr}\right) \tag{3.1}$$

In the equation (3.1), if λ_{qr} were made equal to zero, then the torque equation would be similar to that of a DC machine. This situation is realized by aligning the direct-axis (*d*-axis) of the revolving *d-q* frame with rotor flux, λ_r. The *d-q* frame is now a revolving frame rotating at the synchronous speed, ω. This is shown in Figure 3.1. In this case, the electrical torque from equation (2.57) is given as:

$$T_e = \frac{3}{2} p_p \frac{L_m}{L_r} i_{qs} \lambda_{dr} \tag{3.2}$$

Similar results can be obtained by aligning the stator or air-gap flux with the *d*-axis of the *d-q* frame. The electrical torque for stator field alignment is given as [1]:

$$T_e = \frac{3}{2} p_p i_{qs} \lambda_{ds} \tag{3.3}$$

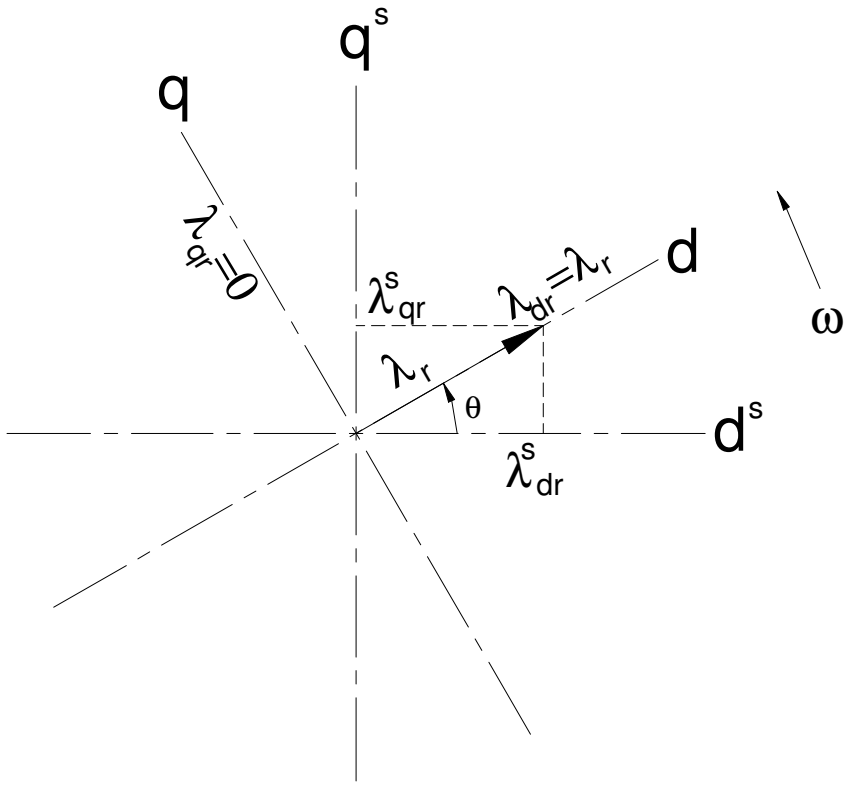


Figure 3.1. Alignment of *d-q* frame with rotor flux vector

The torque in a field-oriented control reacts instantly to changes in the I_{qs} component of stator current, and the reaction of rotor flux to changes in I_{ds} is inertial [1].

In summary, the principal of field-orientation requires that *d-q* frame must be aligned with a selected flux vector. For a given desired torque, $\overline{T}_e$, and desired flux vector, $\overline{\lambda}_f$, the desired currents, $\bar{I}_{ds}$ and $\bar{I}_{qs}$ are determined. Next, the angular position, θ_f, of the flux vector is determined and then used in the *d-q* transformation to obtain the respective three-phase phasors.

3.3.1. Direct Field-Orientation

In field-oriented control, the instantaneous position of the flux vector aligned with rotating *d-q* frame must be known at all times. The position of such a vector can be identified based on a direct measurement. In direct field-orientation, Hall Effect sensors of magnetic field are placed in the air-gap to measure the direct and quadrature component of the air-gap flux. Stator currents are measured as well. The rotor flux is then calculated as:

$$\hat{\lambda}_r = \frac{L_r}{L_m}\hat{\lambda}_m - L_{1s} I_s \tag{3.4}$$

However, these sensors are inconvenient, and require modifications to the induction machine in order to equip them. Furthermore, this spoils the ruggedness of the induction machine. For these reasons, indirect rotor field-orientation is most commonly used for vector control.

3.3.2. Indirect Field-Orientation

In the indirect field-orientation, the estimated angular position, $\overline{\theta}$, of the rotor flux vector is directly computed as:

$$\overline{\theta} = \int_0^t \overline{\omega}_{sl} dt + p_p \theta_m \tag{3.5}$$

where θ_m is the angular displacement of the rotor and $\overline{\omega}_{sl}$ is the required rotor slip frequency. Angular displacement, θ_m, is usually measured by a shaft position sensor such as a digital encoder. The required rotor slip frequency is computed as follows:

$$\overline{\omega}_{sl} = \frac{R_r}{L_r} L_m \frac{\bar{I}_{qs}}{\bar{\lambda}_r} \tag{3.6}$$

For indirect field-oriented control, the desired current $\bar{I}_{ds}$ can be found as follows:

$$\bar{I}_{ds} = \frac{1}{L_m}\left(\bar{\lambda}_r + \frac{L_r}{R_r} \frac{d}{dt} \bar{\lambda}_r \right) \tag{3.7}$$

In steady-state, $\frac{d}{dt}\bar{\lambda}_r = 0$, and $\bar{I}_{ds}$ is:

$$\bar{I}_{ds(steady\ state)} = \frac{\bar{\lambda}_r}{L_m} \tag{3.8}$$

The desired current, $\bar{I}_{qs}$, can be determined from the desired torque, $\overline{T}_e$, as:

$$\bar{I}_{qs} = \frac{2}{3P_p}\left(\frac{L_r}{L_m} \right) \frac{\overline{T}_e}{\bar{\lambda}_r} \tag{3.9}$$

The dynamic equations derived in Chapter 2 are repeated here for convenience. They are given as:

$$V_{qs} = R_s I_{qs} + \frac{d}{dt}\lambda_{qs} + \omega\lambda_{ds} \tag{3.10}$$

$$V_{ds} = R_s I_{ds} + \frac{d}{dt}\lambda_{ds} - \omega\lambda_{qs} \tag{3.11}$$

$$0 = R_r I_{qr} + \frac{d}{dt}\lambda_{qr} + \omega_{sl}\lambda_{dr} \tag{3.12}$$

$$0 = R_r I_{dr} + \frac{d}{dt}\lambda_{dr} + \omega_{sl}\lambda_{qr} \tag{3.13}$$

And the flux linkages are defined as:

$$\lambda_{qs} = L_s I_{qs} + L_m I_{qr} \tag{3.14}$$

$$\lambda_{ds} = L_s I_{ds} + L_m I_{dr} \tag{3.15}$$

$$\lambda_{qr} = L_m I_{qs} + L_r I_{qr} \tag{3.16}$$

$$\lambda_{dr} = L_m I_{ds} + L_r I_{dr} \tag{3.17}$$

For indirect field-orientation, $\lambda_{qr} = 0$, and $\lambda_{dr} = \lambda_r$, then equation (3.13) results in:

$$0 = R_r I_{dr} + \frac{d}{dt}\lambda_{dr} \tag{3.18}$$

Substituting I_{dr} from equation (3.17) into equation (3.18) results in:

$$0 = \frac{R_r}{L_r}\lambda_{dr} + \frac{d}{dt}\lambda_{dr} - \frac{R_r}{L_r}L_m I_{ds} \tag{3.19}$$

Using equations (3.16) and with $\lambda_{qr} = 0$:

$$I_{qr} = -\frac{L_m}{L_r} I_{qs} \tag{3.20}$$

Equation (3.12) can now be written as:

$$0 = -\frac{L_m}{L_r} R_r I_{qs} + \omega_{sl}\lambda_{dr} \tag{3.21}$$

Similarly, equations (3.10) and (3.11) can be rewritten for V_{ds} and V_{qs} as:

$$V_{qs} = R_s I_{qs} + \sigma L_s \frac{d}{dt} I_{qs} + \omega \sigma L_s I_{ds} + \frac{L_m}{L_r} \omega \lambda_{dr} \tag{3.22}$$

$$V_{ds} = R_s I_{ds} + \sigma L_s \frac{d}{dt} I_{ds} - \omega \sigma L_s I_{qs} + \frac{L_m}{L_r} \frac{d}{dt} \lambda_{dr} \tag{3.23}$$

where,

$$\sigma = 1 - \frac{L_m^2}{L_s L_r} \tag{3.24}$$

$$\lambda_{qs} = \sigma L_s I_{qs} \tag{3.25}$$

Equations (3.19), (3.21), (3.22) and (3.23) represent the dynamic voltage equations of the rotor field-oriented induction machine. It can be seen from the equations (3.2), and (3.19) that the stator component currents, I_{ds} and I_{qs}, of the rotating reference frame must be controlled to obtain de-coupled control of torque and rotor flux of the induction machine.

3.4. INDUCTION MACHINE PLANT MODEL

Figure 3.2 shows the simplified model of an indirect filed oriented induction machine system. This figure shows the basic blocks involved in the induction machine plant model. They are; "voltage *d-q* transformation block", "induction machine dynamic *d-q* model block", and "current inverse *d-q* transformation block".

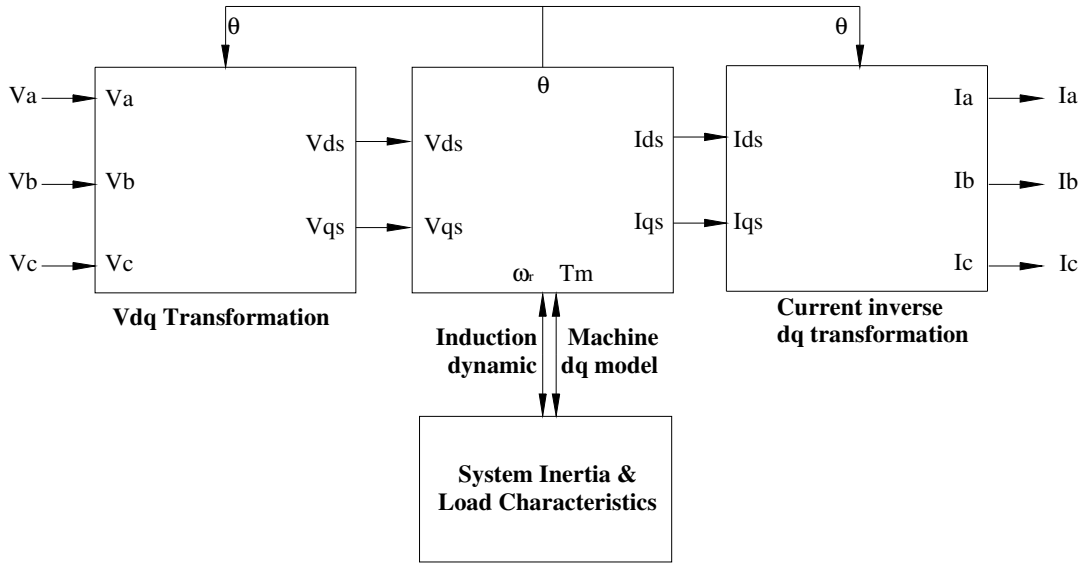


Figure 3.2. Block diagram of the induction machine plant

3.4.1. Voltage *d-q* Transformation Block

In an induction machine system, a controller issues the desired voltage commands for stator voltage components, $\overline{V}_{ds}$, and $\overline{V}_{qs}$ of the revolving frame, which are used to generate three-phase desired values. Using the three-phase desired voltages, the duty cycles needed to control the voltage source inverter are then calculated. Using the calculated duty cycles, and the pulse width modulated (PWM) technique, the voltage source inverter provides the input phase voltages, V_a, V_b, and V_c to the induction machine. This block utilizes the space vector transformations to convert the phase voltages V_a, V_b, and V_c in to the direct-axis voltage, V_{ds}, and quadrature-axis voltage, V_{qs}, of the rotating *d-q* frame. The transformations are performed according to the following equations:

$$\begin{bmatrix} V_{ds}^{s} \\ V_{qs}^{s} \end{bmatrix} = \begin{bmatrix} \frac{2}{3} & -\frac{1}{3} & -\frac{1}{3} \\ 0 & \frac{1}{\sqrt{3}} & -\frac{1}{\sqrt{3}} \end{bmatrix} \begin{bmatrix} V_a \\ V_b \\ V_c \end{bmatrix} \tag{3.26}$$

and

$$\begin{bmatrix} V_{ds} \\ V_{qs} \end{bmatrix} = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} V_{ds}^{s} \\ V_{qs}^{s} \end{bmatrix} \tag{3.27}$$

The *d-q* voltage components are then provided as inputs to the "induction machine *d-q* model block". The rotor flux angular position, θ, needed for the phase voltage transformation into voltage *d-q* components, is computed in the "induction machine dynamic *d-q* model block".

3.4.2. Current Inverse *d-q* Transformation Block.

In this block, using space vector transformations shown in Chapter 2, the *d-q* current components, I_{ds} and I_{qs} of the revolving frame are inverse transformed to the three-phase AC currents. A current sensor is usually used to measure the three-phase currents of the induction machine, and provides current feedback to the controller. The inverse transformations are performed according to the following equations:

$$\begin{bmatrix} I_{ds}^{s} \\ I_{qs}^{s} \end{bmatrix} = \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix} \begin{bmatrix} I_{ds} \\ I_{qs} \end{bmatrix} \tag{3.28}$$

and

$$\begin{bmatrix} I_a \\ I_b \\ I_c \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ -\frac{1}{2} & \frac{\sqrt{3}}{2} \\ -\frac{1}{2} & -\frac{\sqrt{3}}{2} \end{bmatrix} \begin{bmatrix} I_{ds}^{s} \\ I_{qs}^{s} \end{bmatrix} \tag{3.29}$$

The rotor flux angular position, θ, needed for the *d-q* current transformation to three-phase currents, is computed in the "induction machine dynamic *d-q* model block" and used in the transformation block.

3.4.3. Induction Machine Dynamic *d-q* Model Block.

The direct-axis voltage, V_{ds} and quadrature-axis voltage, V_{qs} are provided as inputs to the "induction machine dynamic *d-q* model block". The dynamic equations of the indirect rotor field-oriented induction machine are implemented in this block to simulate the dynamic response of the induction machine. The outputs of the "induction machine dynamic *d-q* model block" are the direct-axis current, I_{ds}, quadrature-axis current, I_{qs}, rotor electrical angular speed, ω_r, and the rotor flux position, θ. Figure 3.3 shows the subset blocks involved that make up the "induction machine dynamic *d-q* model block".

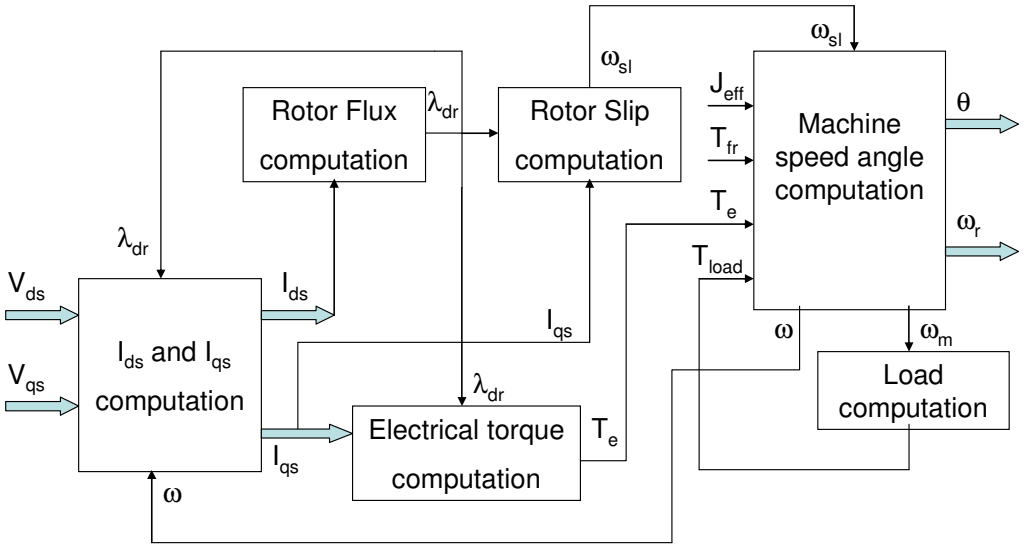


Figure 3.3. Block diagram of induction machine dynamic *d-q* model

3.4.3.1. I_{ds} and I_{qs} Computation

Figure 3.4 shows the block diagram of the I_{ds} and I_{qs} current computation block. In this block, the direct-axis voltages, V_{ds} and V_{qs}, are the inputs provided to the I_{ds} and I_{qs} computational block. The rotor flux λ_{dr} and synchronous speed, ω, which are determined in the rotor flux computation and the machine speed angle computation block respectively, provide the necessary feedback required to compute the direct-axis current, I_{ds}, and quadrature-axis current, I_{qs}.

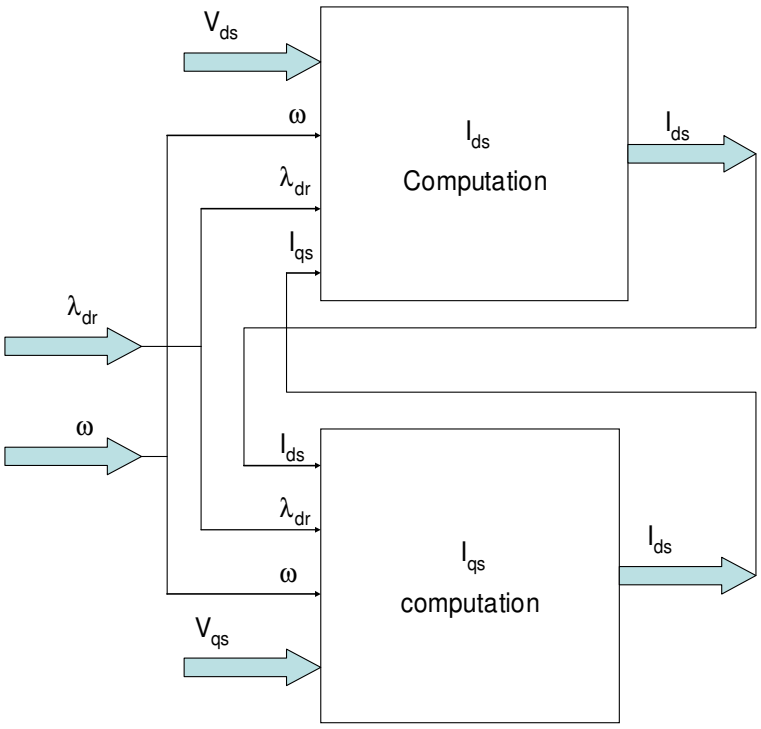


Figure 3.4. I_{ds} and I_{qs} computational block

The stator component currents, I_{ds} and $I_{qs,}$ of the revolving frame are calculated according to the following equations:

$$\sigma L_s \frac{d}{dt} I_{qs} = V_{qs} - R_s I_{qs} - \omega \sigma L_s I_{ds} - \frac{L_m}{L_r} \omega \lambda_{dr} \tag{3.30}$$

$$\sigma L_s \frac{d}{dt} I_{ds} = V_{ds} - R_s I_{ds} + \omega \sigma L_s I_{qs} - \frac{L_m}{L_r} \frac{d}{dt} \lambda_{dr} \tag{3.31}$$

where,

$$\sigma = 1 - \frac{L_m^2}{L_s L_r} \tag{3.32}$$

3.4.3.2. Rotor Flux Computation

In this block, the direct-axis current, I_{ds}, is used to compute the rotor flux. The rotor flux is computed as follows:

$$\frac{d}{dt}\lambda_{dr} = \frac{R_r}{L_r} L_m I_{ds} - \frac{R_r}{L_r}\lambda_{dr} \tag{3.33}$$

3.4.3.3. Rotor Slip Computation

The rotor flux, λ_{dr}, and quadrature-axis current, I_{qs}, are used to compute the rotor slip frequency. The computations are performed according to the following equation:

$$\omega_{sl} = \frac{L_m}{L_r \lambda_{dr}} R_r I_{qs} \tag{3.34}$$

3.4.3.4. Electrical Torque Computation

With the values of rotor flux, λ_{dr}, and quadrature-axis current, $I_{qs,}$ determined, the electrical torque is computed in this block according to the following equation:

$$T_e = \frac{3}{2} p_p \frac{L_m}{L_r} i_{qs} \lambda_{dr} \tag{3.35}$$

3.4.3.5. Load Computation

The load torque at the electric machine is needed to model the induction machine behavior. However, the ability to simulate various load conditions for the induction machine must be maintained. As an example, if one were to study the performance of

induction machines in hybrid or electric vehicles, the vehicle dynamic load would need to be modeled. Other applications of interest where a controlled induction machine dynamic behavior is desired are: elevator/hoist applications, spindle and servo applications and constant load applications. For this research, the load is modeled such that the robustness of the proposed controller under various situations, such as a step load torque change, and variable load torque can be studied. The following equation was used to model the load characteristics:

$$T_{load} = b_0 + b_1\omega_m + b_2\omega_m^2 \tag{3.36}$$

where b_o, b_1 and b_2 are constants, and ω_m is the rotor mechanical speed.

As can be seen from the equation above, by simply adjusting the values of the constants, the induction machine behavior can be studied for different load conditions using a single model.

3.4.3.6. Machine Speed Angle Computation

In this block, friction torque, T_{fr}, load torque, T_{load}, and effective system inertia, J_{eff}, are used to compute the rotor mechanical angular velocity, ω_m. The rotor electrical speed can be determined from the rotor mechanical speed by multiplying it with P_p poles pairs. The followings equations are used in this block to compute friction torque, T_{fr}, rotor mechanical angular velocity, ω_m, rotor electrical angular velocity, ω_r, synchronous speed, ω, and rotor flux position, θ:

$$T_e - T_{fr} - T_{load} = T_m - T_{load} = J_{eff}\frac{d\omega_m}{dt} \tag{3.37}$$

$$T_{fr} = k_f\omega_m + k_r\omega_m^2 \tag{3.38}$$

$$\omega_r = P_p \omega_m \tag{3.39}$$

$$\omega = \frac{d\theta}{dt} \tag{3.40}$$

$$\omega = \omega_{sl} + \omega_r \tag{3.41}$$

In equations (3.37) through (3.41), T_{load} is the load torque (Nm), T_{fr} represents the friction, k_f and k_r are the friction coefficients, T_m is the mechanical machine torque (Nm), J_{eff} is the total effective inertia (kg-m^2), ω_r is the electrical rotor speed (rad/s), ω_m is the rotor mechanical speed (rad/s), and θ is the electrical angular displacement of the rotor flux (rads).

3.5. ARCHITECTURE DEVELOPMENT

MATLAB-SIMULINK tools are used for model architecture development and simulations. This model is detailed enough that it can be used for evaluating the transient and steady-state response of the induction machine. Table 3.1 shows the technical specifications of the induction machine used for the plant model. For the purpose of the induction machine dynamics study and control system development, it is necessary to integrate the developed dynamic model with operator input, and an induction machine controller. In Chapter 4, a model of the induction machine controller that uses a fuzzy logic-based *d-q* control system to accurately control the dynamic response of the induction machine system is developed. In Chapter 5, through simulations, the performance of the proposed fuzzy controller is compared to that of the classical controller.

Table 3. 1. Technical specifications of the induction machine system

Parameter	**Value**
Rated Power	75 Hp
Rated speed	6000 r/min
Rated voltage	460V
Maximum speed	10000 r/min
No. of pole pairs	2
Stator Inductance, L_s	5.837 mH
Rotor Inductance, L_r	5.868 mH
Mutual inductance, L_m	5.650 mH
Stator resistance, R_s	0.0340 Ω
Rotor resistance, R_r	0.0227 Ω
Total effective inertia, J	0.0639 kg-m^2
Rotor flux at rated speed, λ_r	0.289 Wb

4. FUZZY LOGIC-BASED INDUCTION MACHINE CONTROL SYSTEM

4.1. INTRODUCTION

AC induction machines enjoy several advantages such as: ruggedness, simplicity, reliability, and low cost over DC machines. However, their control in high dynamic performance applications remains a challenge due to the non-linear model of the induction machine. Furthermore, motor parameters such as stator and rotor resistance, and mutual inductance vary with the operating conditions. Field-oriented control of the induction machine appears attractive, since under this scheme the de-coupling of the torque and flux is achieved. The de-coupling control method transforms the non-linear induction machine model to a set of linear equations, which can then be controlled by PI controllers. This method, however, greatly depends on the accurate mathematical model of the induction machine [5][4]. Conventional approaches use linear control algorithms to control the induction machine, which can result in undesired dynamic behavior. This issue arises from the fact that a complete high-fidelity mathematical model for the induction machine system, along with parametric variations cannot be accurately modeled inside the controller. Unfortunately, as a result of parameter changes and/or mismatches, the de-coupling of the torque and flux is not completely achieved. The dynamic performance of the induction machine then greatly deteriorates. Because of the non-linear nature of the induction machine, a highly robust control is needed to improve

the dynamic response of the induction machine. Therefore, a controller adaptable to non-linear behaviors and not requiring detailed knowledge of the mathematical model of the plant is required to address such issues [22].

The fuzzy logic-based induction machine controller, based on its non-linear approach, is an attractive choice which can accommodate the parameter variations of the induction machine. For a fuzzy logic controller, an accurate mathematical model of the induction machine is not required. Classical controllers with limitations have been used to control induction machines in achieving desired dynamic response. Fuzzy controls can provide a way to cope with the limitations of the classical controllers [13].

In this study, an innovative fuzzy logic-based *d-q* controller is presented, which improves the dynamic behavior of the induction machine when compared to the classical approach. This can improve two key attributes of the induction machine dynamic behavior; torque response and, I_{ds} and I_{qs}, current response. The objective of this work was to develop an induction machine control system with a non-linear controller using the fuzzy control paradigm to provide accurate control of I_{ds} and I_{qs} current under various transient conditions. The controller minimizes the stator currents, I_{ds} and I_{qs}, overshoots and undershoots and thus results in improved dynamic control of the torque and flux of the induction machine. Furthermore, this approach to current regulation would also mean that the torque response of the induction machine system could be optimized under dynamic conditions. The novel approach presented in this study uses a fuzzy logic-based *d-q* controller to determine output stator voltages commands based on the system's operating conditions.

4.2. SPEED AND TORQUE CONTROL OF INDUCTION MACHINE

A fuzzy logic-based induction machine controller must be able to provide two key attributes of the induction machine control. The attributes are speed control and torque control. Vector control techniques presented in Chapter 3 allow for speed and torque control of the induction machine in both the steady-state and transient operating conditions. Whether the induction machine system operates in speed control or torque control depends on the specific application. As an example, if the induction machine were to be used in an electric vehicle, the driver inputs (such as accelerator and brake pedal commands) would be considered torque commands and thus the controller would operate the induction machine in torque control mode. In the same way, when the electric vehicle is operated in cruise control, the controller provides speed regulation of the electric vehicle and subsequently controls the speed of the induction machine. In such a scenario, the induction machine operates in speed control mode.

4.2.1. Speed Control Based on Desired Speed

Figure 4.1 shows the system architecture from the perspective of induction machine speed control. The speed of the motor, ω_m, is usually determined as a time derivative of the rotor angle, θ_m. A very fast PI controller is then used, which compares the actual speed, $\omega_{m,}$ to the desired or reference speed and issues the torque command to achieve desired speed. Since a classical PI-type controller running at task rates of 100 μs or less with very aggressive proportional and integral terms are used, the error between desired induction machine speed and the actual machine speed is minimal. With such

accurate control of actual induction machine speed to the desired speed, the undesirable or objectionable (oscillatory) affects on induction machine speed are minimal and can be ignored.

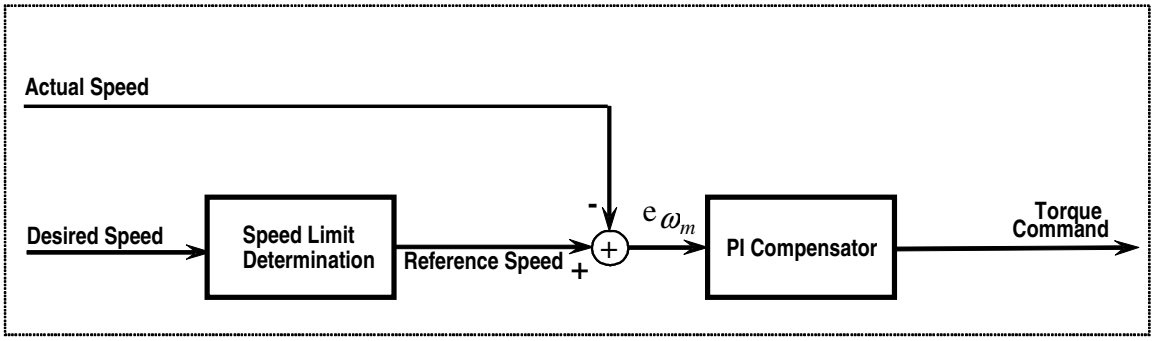


Figure 4.1. Induction machine speed control

4.2.2. Torque Control Based on Desired Torque

As mentioned earlier in Section 4.2, for a given torque demand (e.g. through accelerator and brake pedal requests in an electric vehicle) and machine operating conditions, the fuzzy logic-based induction machine control system commands the desired machine torque and maintains the optimal system dynamic response. Using field-oriented control methods, the controller achieves the independent control of torque and flux producing component of the stator current. Figure 4.2 shows the block diagram of an indirect field-oriented induction machine aligned with the rotor flux vector [1][3]. It is very clear from the block diagram of Figure 4.2 that in order to maintain a robust torque and rotor flux control, the direct-axis current, I_{ds}, and the quadrature-axis current, I_{qs}, of the rotating frame aligned with the rotor flux must be controlled.

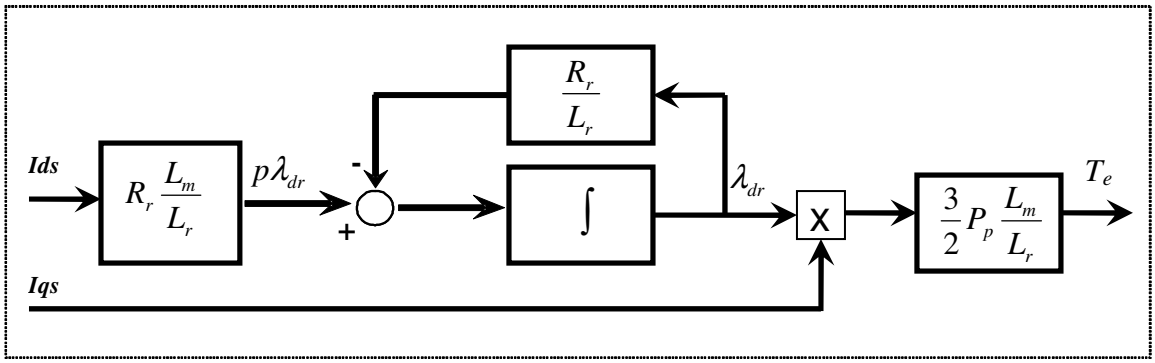


Figure 4.2. Induction machine torque control

It is the responsibility of the fuzzy logic-based induction machine control system to achieve the desired torque response of the induction machine, by controlling the actual three-phase currents to the induction machine. The desired dynamic response is achieved by constantly monitoring the current and speed feedback and calculating the desired three-phase voltages for the induction machine. Based on the operator inputs of mechanical torque requirements, and the angular speed feedback, the fuzzy logic-based induction machine control system determines the desired machine electrical torque and a desired rotor flux. The desired electrical torque and the desired rotor flux are then used to calculate desired direct-axis current, $\bar{I}_{ds}$, and the quadrature-axis current, $\bar{I}_{qs}$, of the rotating frame aligned with the rotor flux. The desired direct-axis current, $\bar{I}_{ds}$, and the quadrature-axis current, $\bar{I}_{qs}$ are used by the fuzzy logic-based *d-q* controller to calculate desired stator voltage components, $\bar{V}_{ds}$ and $\bar{V}_{qs}$. With the estimation of the angular position of rotor flux vector, θ, the desired component voltages, $\bar{V}_{ds}$ and $\bar{V}_{qs}$ of the revolving frame are transformed into desired PWM signals (duty cycle values), which are

then used by the inverter to generate the three-phase voltages for the induction machine. Using the feedback on the instantaneous values of stator component currents I_{ds} and I_{qs}, and comparing to the desired values of stator component currents $\bar{I}_{ds}$ and $\bar{I}_{qs}$ respectively, the fuzzy logic-based *d-q* controller controls the desired stator component voltages, $\overline{V}_{ds}$ and $\overline{V}_{qs}$, of the revolving frame. Use of the classical controller for controlling the desired component voltages, $\overline{V}_{ds}$ and $\overline{V}_{qs}$, could result in undesired induction machine response under certain conditions. Such an undesired response is not acceptable, as the classical approach using a PI controller with de-coupling method is unable to control the dynamic response of the induction machine. Since the classical PI-based control system is mostly effective for linear control processes, non-linear PI controllers are needed to satisfactorily control non-linear plants, or time-varying plants [22]. Since the induction machine exhibit both non-linear and time-varying behaviors, it is desirable to use a non-linear controller. Since fuzzy control systems do not require an accurate mathematical model of the system to be controlled, and allow for the development of a knowledge-based non-linear controller; a better candidate for improving the induction machine behavior is to use a fuzzy logic-based control system. Fuzzy logic-based *d-q* controllers can be used for non-linear control of such plants [14].

4.3. *MODEL OF FUZZY LOGIC-BASED INDUCTION MACHINE CONTROL SYSTEM*

As mentioned in Section 4.2.2, the fuzzy logic-based induction machine control system is responsible for achieving the desired torque and speed of the induction

machine. Based on the operator inputs, including acceleration or brake pedal in an electric vehicle or operator torque command in a dynamometer application and operating conditions, the controller determines the desired machine electrical torque, and the desired rotor flux. Based on the desired machine electrical torque, and the desired rotor

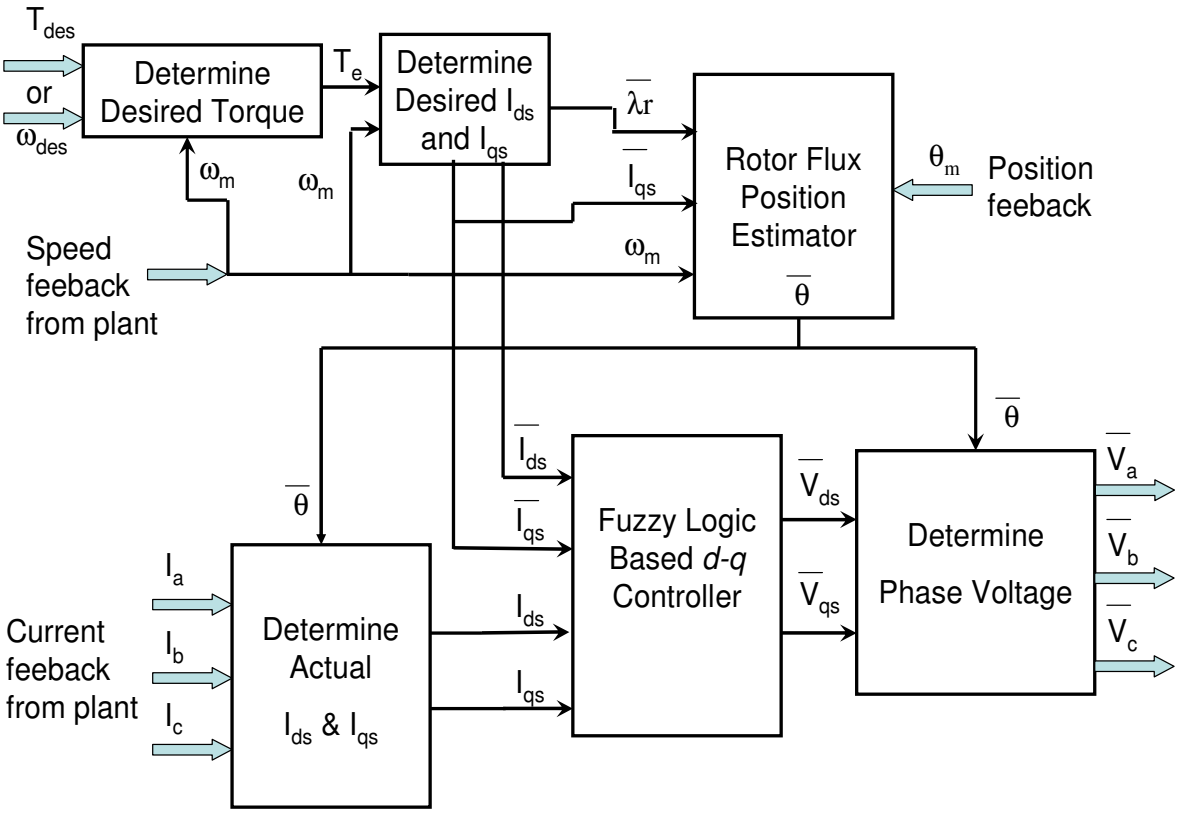


Figure 4.3. Fuzzy logic-based induction machine control system

flux, the controller determines desired stator component currents, $\bar{I}_{ds}$ and $\bar{I}_{qs}$ of the rotating frame. Using a fuzzy logic-based *d-q* controller, the controller determines the desired stator voltages, $\overline{V}_{ds}$ and $\overline{V}_{qs}$, of the revolving frame. Using the estimated rotor

flux angular position, θ, these voltages are then transformed into desired PWM signals (duty cycle values), which are then fed to the PWM inverter to generate the three-phase voltages for the induction machine. Since the scope of this research deals with improving the dynamic response of the induction machine, the inverter PWM switching effects are neglected. Instead a sinusoidal voltage source of variable magnitude represents the fundamental component of the actual PWM inverter waveform. Nevertheless, this study still provides the understanding of the induction machine dynamic behavior and presents an improved way of controlling its dynamic response. Figure 4.3 shows the basic blocks of the fuzzy logic-based induction machine control system essential for the induction machine control.

4.3.1. Determine Desired Torque Controller Block

This is the block which translates the operator demands. In this block, the controller decides whether speed control or torque control is to be used. Operator requests such as speed commands require the controller to operate the induction machine in speed control mode. In this mode, as explained in Section 4.2.1, the speed of the motor, ω_m, is compared with the desired reference speed, $\overline{\omega}_m$. A very fast PI controller is then used which compares the actual speed, $\omega_{m,}$ to the reference speed, $\overline{\omega}_m$ and produces a reference value, $\overline{T}_m$ of motor torque to achieve target speed. When the operator request is in the form of a desired torque, $\overline{T}_m$, the controller operating in torque control mode determines the desired electrical torque according to the following equation:

$$\overline{T}_e = \overline{T}_m + k_f \omega_m + k_r \omega_m^2 \tag{4.1}$$

where k_f and k_r are the friction coefficients.

The desire electrical torque, $\overline{T}_e$, and the rotor angular speed, ω_m, are then forwarded to the "determine I_{ds} and I_{qs} block".

4.3.2. Determine Desired I_{ds} and I_{qs} Block

This block is responsible for computing the desired stator component currents, $\bar{I}_{ds}$ and $\bar{I}_{qs}$, of the revolving frame based on the desired electrical torque, $\overline{T}_e$ and desired rotor flux, $\overline{\lambda}_r$. The manufacturer of the induction machine usually provides the rated flux over an operating speed range. In the absence of such, a desired rotor flux vector, $\overline{\lambda}_r$, can be computed using the steady-state equivalent circuit of the induction machine. In most practical applications, the rotor flux is controlled according to the following formula [1]:

$$\overline{\lambda}_r = \begin{cases} \lambda_r & for \quad \omega_m \le \omega_{m,rated} \\ \dfrac{\omega_{m,rated}}{\omega_m} \lambda_r & for \quad \omega_m > \omega_{m,rated} \end{cases} \tag{4.2}$$

where $\omega_{m,\,rated}$ denotes the rated motor speed. Equation (4.2) ensures that the stator voltage under the field-weakening conditions will not exceed the rated value. For this purpose, this block uses a look-up table based on desired rotor flux, $\overline{\lambda}_r$ and corresponding rotor speed, ω_m. The desired rotor flux at rated speed is calculated from the steady-state equivalent circuit, and the subsequent rotor flux values in field-weakening region are calculated according to Equation (4.2). This ensures that stator

voltage under field-weakening conditions will not exceed the rated value. Similarly, based on the induction machine ratings, a lookup table for electrical torque, $\overline{T}_e$, and rotor angular speed, ω_m, is created to ensure that at any given point, the desired electrical torque, $\overline{T}_e$, does not exceed the motor rated capabilities. Finally, the stator component currents, $\bar{I}_{ds}$ and $\bar{I}_{qs}$ of the revolving frame are calculated as follows:

$$\bar{I}_{ds} = \frac{1}{L_m}\left(\overline{\lambda}_r + \frac{L_r}{R_r}\frac{d}{dt}\overline{\lambda}_r\right) \tag{4.3}$$

In steady-state, $\frac{d}{dt}\overline{\lambda}_r = 0$, and $\bar{I}_{ds}$ is,

$$\bar{I}_{ds(steady\ state)} = \frac{\overline{\lambda}_r}{L_m} \tag{4.4}$$

and

$$\bar{I}_{qs} = \frac{2}{3P_p}\left(\frac{L_r}{L_m}\right)\frac{\overline{T}_e}{\overline{\lambda}_r} \tag{4.5}$$

The desired currents, $\bar{I}_{ds}$ and $\bar{I}_{qs}$, are the inputs used in both the "rotor flux position estimator block" as well as the "fuzzy logic-based *d-q* controller block".

4.3.3. Rotor Flux Position Estimator Block

This block uses desired current, $\bar{I}_{qs}$, desired rotor flux, $\overline{\lambda}_r$, and rotor angular speed, ω_m, and displacement, θ_m, as its inputs. The main function of this block is to estimate the angular speed, $\overline{\omega}$, and the angular displacement, $\overline{\theta}$, of the rotor flux vector. Based on the induction machine parameters, this block estimates the rotor slip frequency

and uses it to compute the rotor angular displacement, $\bar{\theta}$. The estimated rotor flux angular speed is computed according to the following equation:

$$\bar{\omega} = \bar{\omega}_{sl} + \omega_r \tag{4.6}$$

where,

$$\bar{\omega}_{sl} = \frac{R_r}{L_r} L_m \frac{\bar{I}_{qs}}{\bar{\lambda}_r} \tag{4.7}$$

and

$$\omega_r = P_p \omega_m \tag{4.8}$$

The estimated rotor flux angular position, $\bar{\theta}$, of the revolving reference frame is given as:

$$\bar{\theta} = \int_0^t \bar{\omega}_{sl} dt + P_p \theta_m \tag{4.9}$$

where θ_m is the angular displacement of the rotor and $\bar{\omega}_{sl}$ is the estimated rotor slip frequency. The angular displacement of rotor, θ_m, is usually measured by a shaft position sensor such as a digital encoder.

4.3.4. Determine Actual I_{ds} and I_{qs} Block

The main function of this block is to transform the measured phase currents (feedback obtained on the instantaneous value of the phase currents through the current sensor) into the stator component currents, I_{ds} and I_{qs}, of the revolving frame. This transformation is similar to that of the inverse transformation performed in Section 3.4.2, except that this block performs a forward transformation and uses estimated angular

displacement, $\bar{\theta}$, of the rotor flux vector. The transformations used in this block are given as:

$$\begin{bmatrix} I_{ds}^{s} \\ I_{qs}^{s} \end{bmatrix} = \begin{bmatrix} \frac{2}{3} & -\frac{1}{3} & -\frac{1}{3} \\ 0 & \frac{1}{\sqrt{3}} & -\frac{1}{\sqrt{3}} \end{bmatrix} \begin{bmatrix} I_a \\ I_b \\ I_c \end{bmatrix} \tag{4.10}$$

and

$$\begin{bmatrix} I_{ds} \\ I_{qs} \end{bmatrix} = \begin{bmatrix} \cos\bar{\theta} & \sin\bar{\theta} \\ -\sin\bar{\theta} & \cos\bar{\theta} \end{bmatrix} \begin{bmatrix} I_{ds}^{s} \\ I_{qs}^{s} \end{bmatrix} \tag{4.11}$$

4.3.5. Determine Phase Voltage Block

The main function of this block is to transform the desired stator component voltages, $\bar{V}_{ds}$ and $\bar{V}_{qs}$ to desired phase voltages, $\bar{V}_a$, $\bar{V}_b$, and $\bar{V}_c$ to be supplied to the induction machine. This transformation is similar to that of the transformation performed in Section 3.4.1, except that this block performs an inverse transformation and uses estimated angular displacement, $\bar{\theta}$, of the rotor flux vector. The transformations used in this block are given as:

$$\begin{bmatrix} \bar{V}_{ds}^{s} \\ \bar{V}_{qs}^{s} \end{bmatrix} = \begin{bmatrix} \cos(\bar{\theta}) & -\sin(\bar{\theta}) \\ \sin(\bar{\theta}) & \cos(\bar{\theta}) \end{bmatrix} \begin{bmatrix} \bar{V}_{ds} \\ \bar{V}_{qs} \end{bmatrix} \tag{4.12}$$

and

$$\begin{bmatrix} \bar{V}_a \\ \bar{V}_b \\ \bar{V}_c \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ -\frac{1}{2} & \frac{\sqrt{3}}{2} \\ -\frac{1}{2} & -\frac{\sqrt{3}}{2} \end{bmatrix} \begin{bmatrix} \bar{V}_{ds}^{s} \\ \bar{V}_{qs}^{s} \end{bmatrix} \tag{4.13}$$

4.3.6. Fuzzy Logic-based *d-q* Controller

The main function of this block is to calculate the output stator component voltages commands, $\overline{V}_{ds}$ and $\overline{V}_{qs}$ of the revolving reference frame such that the actual stator currents I_{ds} and I_{qs} could be controlled. In this block, feedback on the actual values of the I_{ds} and I_{qs} is compared with the desired $\bar{I}_{ds}$ and $\bar{I}_{qs}$. Any overshoot or unexpected behavior in the actual I_{ds} and I_{qs} currents can result in a poor torque and current response of the induction machine. Such poor responses are not acceptable in high performance drive applications.

In rotor field-oriented control, the direct-axis stator current, I_{ds} and the quadrature-axis current, I_{qs} of the synchronous revolving *d-q* frame must be controlled independently which would allow independent control of torque and rotor flux of the induction machine. The stator component voltages from equations (3.22) and (3.23) in Chapter 3, are given as:

$$V_{ds} = R_s I_{ds} + \sigma L_s \frac{d}{dt} I_{ds} - \omega \sigma L_s I_{qs} + \frac{L_m}{L_r} \frac{d}{dt} \lambda_{dr} \tag{4.14}$$

and

$$V_{qs} = R_s I_{qs} + \sigma L_s \frac{d}{dt} I_{qs} + \omega \sigma L_s I_{ds} + \frac{L_m}{L_r} \omega \lambda_{dr} \tag{4.15}$$

It can be seen from equations (4.14) and (4.15), that the stator voltage equations are linked and are non-linear. In other words, the equations of the stator voltage components are coupled. The direct-axis voltage, V_{ds} also depends on stator current, I_{qs} and the quadrature-axis voltage, V_{qs} also depends on stator current I_{ds}. In classical approaches, the stator currents I_{ds} and I_{qs} can only be independently controlled if the stator voltage

equations are de-coupled [4][5]. By manipulating the fundamental equations (3.22) and (3.23) in Chapter 3 with $\lambda_{qr} = 0$ and $\lambda_{dr} =$ constant, the stator voltage equations can be written as[5]:

$$V_{ds} = R_s I_{ds} + \sigma L_s \frac{d}{dt} I_{ds} - \omega \sigma L_s I_{qs} \tag{4.16}$$

$$V_{qs} = R_s I_{qs} + \sigma L_s \frac{d}{dt} I_{qs} + \omega L_s I_{ds} \tag{4.17}$$

Hence, classical approach uses de-coupling schemes using feed-forward terms together with PI controllers. In these approaches, the output stator voltages command, $\overline{V}_{ds}$ and $\overline{V}_{qs}$, are calculated using two PI controllers with de-coupling terms ($\omega L_s I_{ds}$, $\omega \sigma L_s I_{qs}$) which compares the error of the actual currents , I_{ds} and I_{ds}, to that of desired currents, $\bar{I}_{ds}$ and $\bar{I}_{qs}$, respectively. The classical approach using PI controllers with the de-coupling compensation method is represented through the following equations [4][5]:

$$P_{ds}(n) = K_{p\_ds}(\bar{I}_{ds}(n) - I_{ds}(n)) = K_{p\_ds} e_{ds}(n) \tag{4.18}$$

$$S_{ds}(n) = K_{i\_ds} \sum_{i=1}^{n} (\bar{I}_{ds}(i) - I_{ds}(i)) T_s = K_{i\_ds} \sum_{i=1}^{n} e_{ds}(i) T_s \tag{4.19}$$

$$\overline{V}_{ds}(n) = P_{ds}(n) + S_{ds}(n) - \overline{\omega}(n) \tilde{\sigma} \tilde{L}_s I_{qs}(n) \tag{4.20}$$

and

$$P_{qs}(n) = K_{p\_qs}(\bar{I}_{qs}(n) - I_{qs}(n)) = K_{p\_qs} e_{qs}(n) \tag{4.21}$$

$$S_{qs}(n) = K_{i\_qs} \sum_{i=1}^{n} (\bar{I}_{qs}(i) - I_{qs}(i)) T_s = K_{i\_qs} \sum_{i=1}^{n} e_{qs}(i) T_s \tag{4.22}$$

$$\overline{V}_{qs}(n) = P_{qs}(n) + S_{qs}(n) + \overline{\omega}(n) \tilde{L}_s I_{qs}(n) \tag{4.23}$$

where K_{p_ds} and K_{i_ds} denote proportional and integral gains for direct-axis, and $\tilde{\sigma}$, $\tilde{L}_s$ are estimates of σ, L_s respectively. The final direct-axis stator voltage command output $\overline{V}_{ds}(n)$ is the sum of the direct-axis proportional term, $P_{ds}(n)$, and the direct-axis integral term, $S_{ds}(n)$. Note that, $e_{ds}(n)$ is the error between the desired direct-axis current, $\bar{I}_{ds}$, and the actual direct-axis current, I_{ds}, and T_s is the sampling time. Similarly K_{p_qs} and K_{i_qs} denote proportional and integral gains for quadrature-axis. The final quadrature-axis stator voltage command output $\overline{V}_{qs}(n)$ is the sum of the quadrature-axis proportional term, $P_{qs}(n)$, and the quadrature-axis integral term, $S_{qs}(n)$ with $e_{qs}(n)$ representing the error between the desired quadrature-axis current, $\bar{I}_{qs}$, and the actual quadrature-axis current, I_{qs}. Notice that the feed-forward compensation terms in the above equations, $-\overline{\omega}(n)\tilde{\sigma}\tilde{L}_s I_{qs}(n)$ and $\overline{\omega}(n)\tilde{L}_s I_{ds}(n)$, are used to cancel out the coupling terms of the induction machine. The classical approach is primarily based on linear control methods, and using the feed forward de-coupling scheme, the non-linear model of induction machine is transformed to a set of linear equations, which are then controlled by the PI controllers. Figure 4.4 shows the diagram of a classical *d-q* controller employing PI control system with feed-forward compensation. The classical *d-q* controller assumes an ideal situation, where there are no parameter mismatches, and therefore the classical *d-q* controller with feed forward de-coupling terms is able to achieve the desired induction machine response. However, in real world applications, there are always parameter mismatches and the estimated values of the induction machine parameters are different from the actual values. Furthermore, induction machine parameters are not fixed but

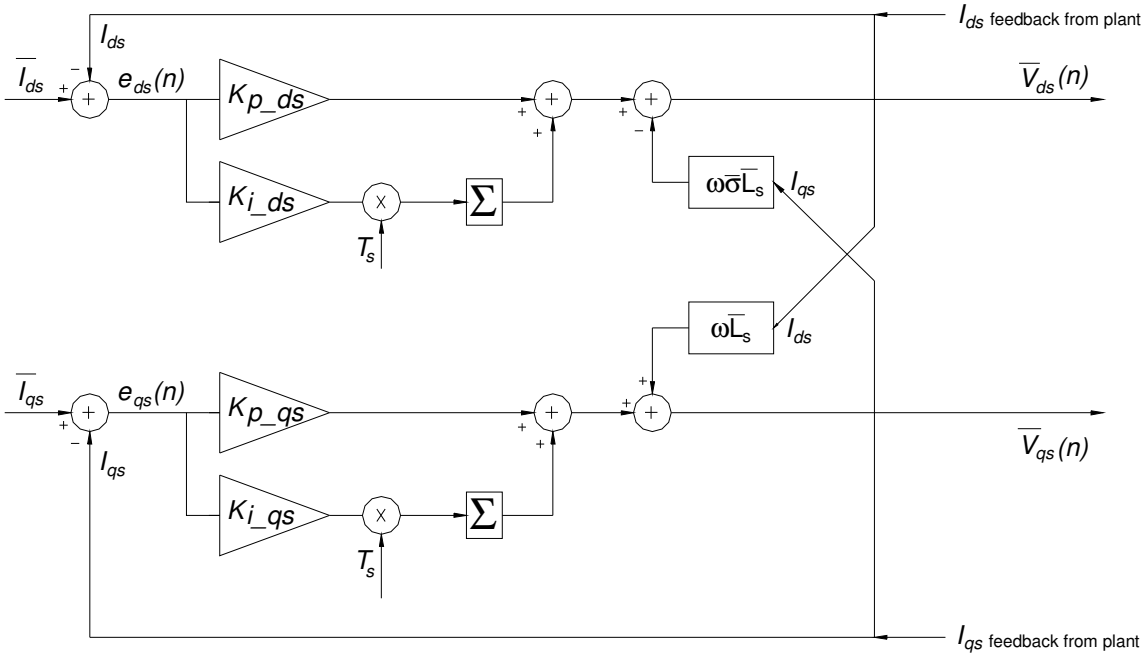


Figure 4.4. Classical *d-q* controller

rather dependent on other variables. As an example, the rotor resistance, R_r, varies with the temperature of the machine, and stator inductance, L_s, and mutual inductance, L_m, vary depending on the magnitude and frequency of the stator current. These parameter errors can be as much as 30% [4][5]. Under such a scenario, the de-coupling scheme does not function well. Under parameter mismatches, the de-coupling scheme will have compensation errors, $-\overline{\omega}(\sigma L_s - \tilde{\sigma}\tilde{L}_s)I_{qs}$ and $\overline{\omega}(L_s - \tilde{L}_s)I_{ds}$. Note that the errors become large at high speeds and simply can not be ignored. The approach presented in this research does not use the feed-forward de-coupling scheme but instead uses a fuzzy logic-based *d-q* controller, to calculate the desired stator voltage components, $\overline{V}_{ds}$ and $\overline{V}_{qs}$. The fuzzy logic-based *d-q* controller shown in Figure 4.5 is responsible for

achieving improved induction machine dynamic behavior. The fuzzy logic-based control system improves the induction machine dynamic behavior by utilizing the human control knowledge and experience to intuitively construct an intelligent controller, so that the resulting controller will emulate the desired control behavior to a certain extent [22].

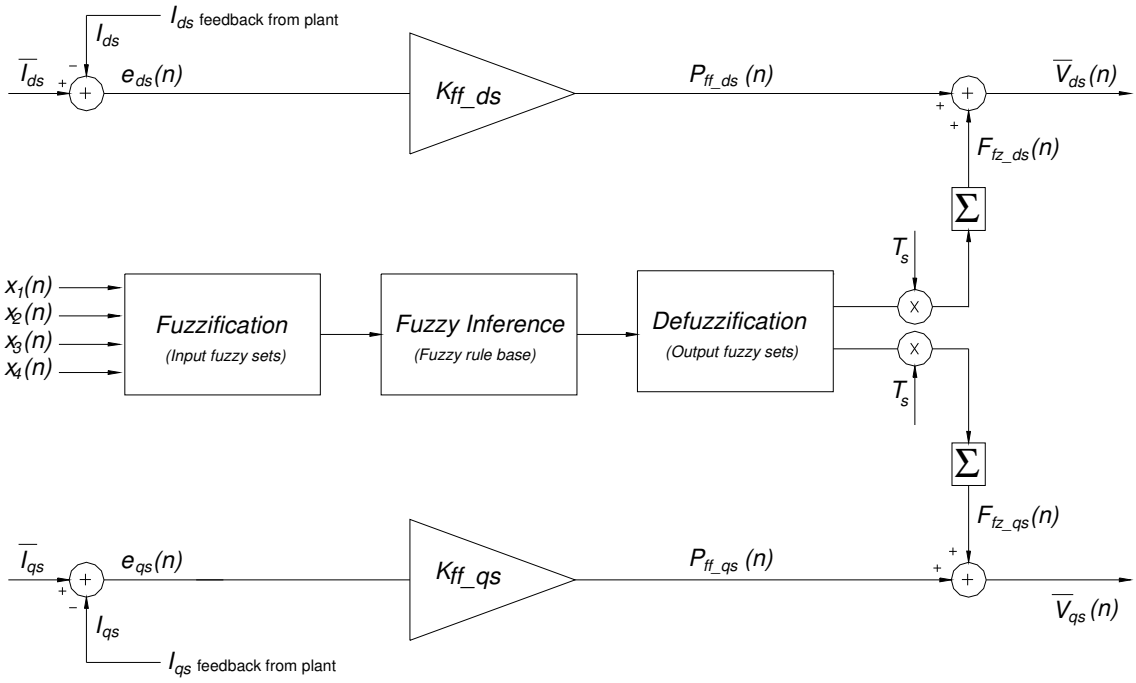


Figure 4.5. Fuzzy *d-q* controller

For this purpose, a multiple-input multiple-output (MIMO) Mamdani fuzzy logic [22] based *d-q* control system was utilized. The formulation of the proposed fuzzy logic-based *d-q* control system in Figure 4.5 can be described as follows:

$$P_{ff\_ds}(n) = K_{ff\_ds}(\bar{I}_{ds}(n) - I_{ds}(n)) = K_{ff\_ds} e_{ds}(n) \tag{4.24}$$

$$F_{fz\_ds}(n) = \sum_{i=1}^{n} (\bar{I}_{ds}(i) - I_{ds}(i)) \Delta u_{ds} T_s = \sum_{i=1}^{n} e_{ds}(i) \Delta u_{ds} T_s \tag{4.25}$$

$$\overline{V}_{ds}(n) = P_{ff\_ds}(n) + F_{fz\_ds}(n) \tag{4.26}$$

and

$$P_{ff\_qs}(n) = K_{ff\_qs}(\bar{I}_{qs}(n) - I_{qs}(n)) = K_{ff\_qs}e_{qs}(n) \tag{4.27}$$

$$F_{fz\_qs}(n) = \sum_{i=1}^{n}(\bar{I}_{qs}(i) - I_{qs}(i))\Delta u_{qs}T_s = \sum_{i=1}^{n}e_{qs}(i)\Delta u_{qs}T_s \tag{4.28}$$

$$\bar{V}_{qs}(n) = P_{ff\_qs}(n) + F_{fz\_qs}(n) \tag{4.29}$$

It is clear that this controller consists of a MIMO fuzzy logic controller together with a proportional error compensator. K_{ff_ds} is the proportional direct-axis stator voltage error compensation gain and $P_{ff\_ds}(n)$ is the direct-axis proportional error compensated stator voltage command term. Δu_{ds} is the direct-axis fuzzy commanded change in the stator voltage, which is integrated to obtain direct-axis fuzzy stator voltage feedback command, $F_{fz\_ds}(n)$. Note that, $e_{ds}(n)$ is the error between the desired direct-axis current, $\bar{I}_{ds}$, and the actual direct-axis current, I_{ds}, and T_s is the sampling time. Similarly, K_{ff_qs} is the proportional quadrature-axis stator voltage error compensation gain and $P_{ff\_qs}(n)$ is the quadrature-axis proportional error compensated stator voltage command term. Δu_{qs} is the quadrature-axis fuzzy commanded change in the stator voltage, which is integrated to obtain quadrature-axis fuzzy stator voltage feedback command, $F_{fz\_qs}(n)$. Note that, $e_{qs}(n)$ is the error between the desired quadrature-axis current, $\bar{I}_{qs}$, and the actual quadrature-axis current, I_{qs}, and T_s is the sampling time. Finally, $P_{ff\_ds}(n)$ and $F_{fz\_ds}(n)$ are added to determine direct-axis output stator voltage command, and $P_{ff\_qs}(n)$ and $F_{fz\_qs}(n)$ are added to determine quadrature-axis output stator voltage command. At this point, it is important to emphasize that if a classical *d-q* controller was only used, then the equations (4.24), (4.25), (4.26), (4.27), (4.28) and (4.29) governing the behavior

of the proposed fuzzy logic-based *d-q* controller will be replaced by equations (4.18), (4.19), (4.20), (4.21), (4.22) and (4.23) to obtain the classical controller.

Next, to design the fuzzy *d-q* controller, input variables, output variables, and input and output fuzzy sets are defined [22]. To control the actual stator currents I_{ds} and I_{qs} during steady-state events, the magnitude of the errors between $\bar{I}_{ds}$ and $\bar{I}_{qs}$ and the actual I_{ds} and I_{qs} respectively are selected as two of the inputs to the fuzzy logic-based *d-q* controller. To control induction machine behavior during transient events, the magnitudes of the change of these errors are used as two of the other inputs. The combination of the error and change in error provides the prediction mechanism of undesired response. The input variables for the controller can be written as follows,

$$x_1(n) = \left|e_{ds}(n)\right| = \left|\bar{I}_{ds}(n) - I_{ds}(n)\right| \tag{4.29}$$

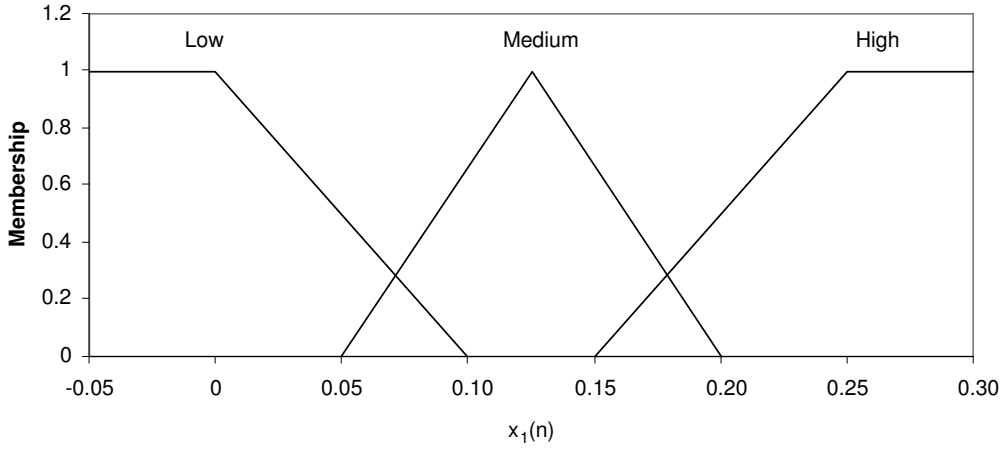
$$x_2(n) = \left|\Delta e_{ds}(n)\right| = \left|e_{ds}(n) - e_{ds}(n-1)\right| \tag{4.30}$$

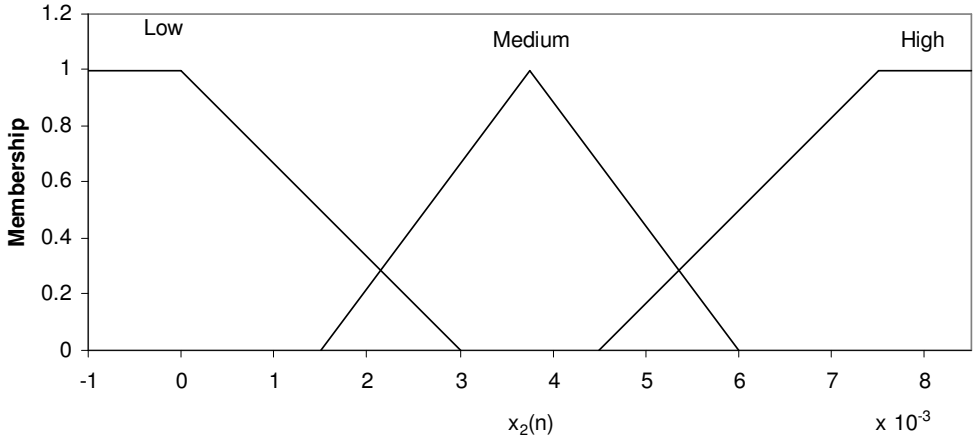
$$x_3(n) = \left|e_{qs}(n)\right| = \left|\bar{I}_{qs}(n) - I_{qs}(n)\right| \tag{4.31}$$

$$x_4(n) = \left|\Delta e_{qs}(n)\right| = \left|e_{qs}(n) - e_{qs}(n-1)\right| \tag{4.32}$$

The input fuzzy sets or membership functions for $x_1(n)$, $x_2(n)$, $x_3(n)$ and $x_4(n)$ that are used for fuzzification are shown in Figure 4.6. The output fuzzy sets used for de-fuzzification are the singleton types as shown in Figure 4.7. The fuzzy rules for this fuzzy *d-q* controller are described in Table 4.1. The fuzzy rules are laid out in a manner such that the they can distinguish between various steady-state, predictive and transient conditions and decide on the current and future states for control purposes. This enables the fuzzy rules to cover conditions where parameter estimation error can easily be compensated. In addition, the fuzzy rules provide the ability for the controller to

anticipate conditions where inappropriate response may occur, thereby providing a mechanism to avoid possible undesirable responses. The description of these fuzzy rules is provided in Table 4.1. Among these fuzzy rules, some of the rules are intended to cover steady-state and transient conditions, and the others are used for predictive conditions. For example, rules 1 and 4, where $x_1(n)$ is either low or medium and $x_2(n)$ are both low, and $x_3(n)$ and $x_4(n)$ are irrelevant (none or don't care), depict a steady-state condition where there is no need for further corrective actions for direct-axis and





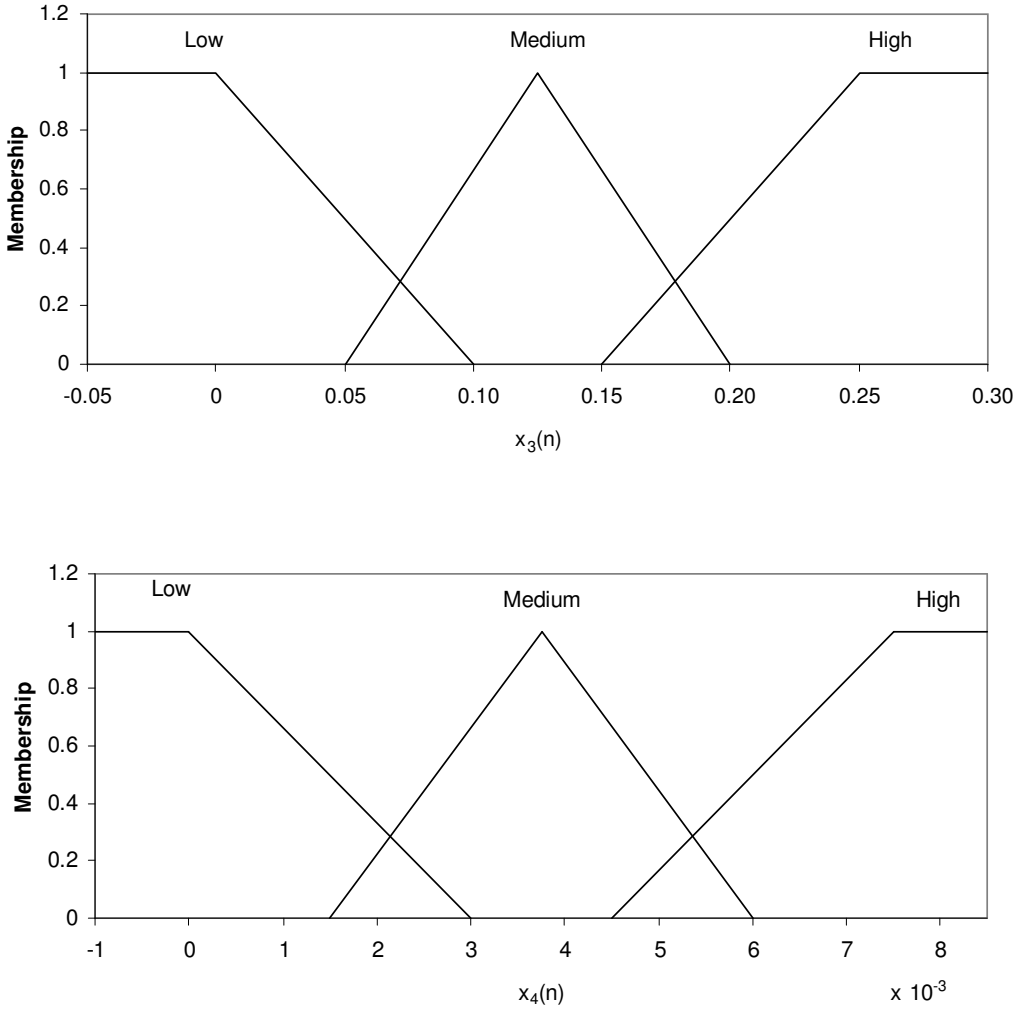


Figure 4.6. Input variables and their fuzzy sets

hence use a low value (h_{ds_L}) for the change of stator direct voltage command, $\Delta u_{ds}(n)$. Rule 3, where $x_1(n)$ is low, $x_2(n)$ is high, $x_3(n)$ and $x_4(n)$ are irrelevant (none or don't care), indicates a predictive condition for direct-axis where major correction may need to be made in the future, and therefore uses a high value (h_{ds_H}) for the change of stator direct-axis voltage command, $\Delta u_{ds}(n)$. Rules 5 and 6, where $x_1(n)$ are both medium, $x_2(n)$ is either medium or high, $x_3(n)$ and $x_4(n)$ are irrelevant (none or don't care), depicts an immediate corrective condition for direct-axis requiring feedback

correction, while at the same time predicts that the situation may deteriorate in the future. Rules 5 and 6 therefore correct for this issue by increasing the value for the change of

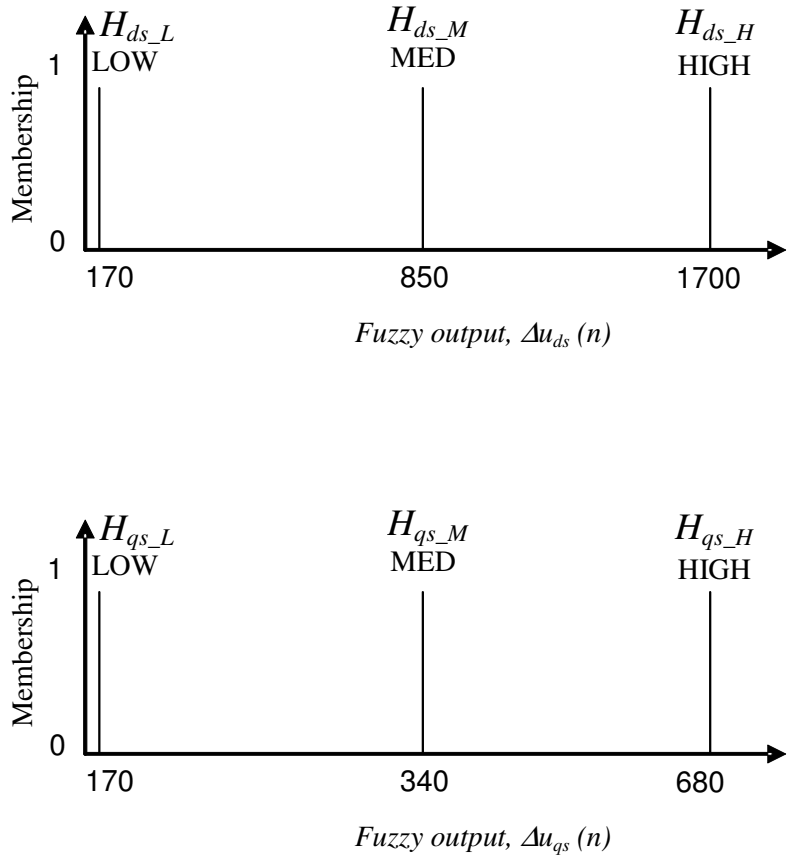


Figure 4.7. Output variable and its singleton fuzzy sets

stator direct voltage command, $\Delta u_{ds}(n)$ to high value (h_{ds_H}). Similarly rule 10, where $x_1(n)$ and $x_2(n)$ are irrelevant (none or don't care), $x_3(n)$ is low and $x_4(n)$ is low, depicts a quadrature-axis steady-state condition with minimal errors in desired and actual quadrature current and hence there is no need for further corrective actions for quadrature-axis. Therefore the controller uses a low value (h_{qs_L}) for the change of stator quadrature voltage command, $\Delta u_{qs}(n)$. Rule 14, where $x_1(n)$ and $x_2(n)$ are irrelevant

Table 4.1. Fuzzy rules for *d-q* controller

Rule No.	If $x_1(n)$ is	if $x_2(n)$ is	if $x_3(n)$ is	if $x_4(n)$ is	Then $\Delta u_{ds}(n)$ is	Then $\Delta u_{qs}(n)$ is	Explanation
1	low	low	none	none	low	none	Direct-axis steady-state condition.
2	low	med	none	none	med	none	Predictive condition for direct-axis where in future correction may need to be made.
3	low	high	none	none	high	none	Predictive condition for direct-axis where major correction may need to be made in future.
4	med	low	none	none	low	none	Direct-axis close to steady-state condition.
5	med	med	none	none	high	none	Corrective condition for direct-axis requiring feedback correction.
6	med	high	none	none	high	none	Corrective condition for direct-axis requiring feedback correction, but at the same time predicts that situation may deteriorate in future.
7	high	low	none	none	med	none	Direct-axis close to steady-state condition but due to high steady-state error, there is a need to increase corrective action during this condition.
8	high	med	none	none	high	none	Corrective condition for direct-axis requiring major feedback correction.
9	high	high	none	none	high	none	Fast transient corrective condition for direct-axis requiring major feedback correction.
10	none	none	low	low	none	low	Quadrature-axis steady-state condition with minimal errors.
11	none	none	low	med	none	low	Predictive condition for quadrature-axis, where current condition is close to steady-state with minimal errors, but there is a chance for overshoot and hence provides minimal corrective actions.
12	none	none	low	high	none	low	Fast transient condition for quadrature-axis with minimal errors, but there is a major chance for overshoot and hence provides minimal corrective actions.
13	none	none	med	low	none	med	Quadrature-axis current close to steady-state condition with some errors, and hence increase corrective action to improve response.
14	none	none	med	med	none	Med	Predictive condition for quadrature-axis with some errors and there is some chance for overshoot, hence keep some corrective action to improve response.
15	none	none	med	high	none	Low	Fast transient condition for quadrature-axis with some errors and there is major chance for overshoot, hence keep minimal corrective action to improve response.
16	none	none	high	low	none	high	Quadrature-axis close to steady-state condition with major errors, and hence use major corrective action to improve response.
17	none	none	high	med	none	Med	Predictive condition for quadrature-axis with major errors and there is some chance for overshoot, hence keep some corrective action to improve response.
18	none	none	high	high	none	Low	Fast transient condition for quadrature-axis with major errors and there is major chance for overshoot, hence keep minimal corrective action to improve response.

(none or don't care), $x_3(n)$ is medium and $x_4(n)$ is medium, represents a predictive condition for quadrature-axis with some error in desired and actual quadrature current. Also, due to the existence of medium value of the change of error, there is some chance for overshoot, hence rule 14 provides some corrective action to improve response by using a medium value (h_{qs_M}) for the change of stator quadrature voltage command, $\Delta u_{qs}(n)$. Finally, rule 15, where $x_1(n)$ and $x_2(n)$ are irrelevant (none or don't care), $x_3(n)$ is medium and $x_4(n)$ is high, represents a fast transient condition for quadrature-axis with some error in desired and actual quadrature current. There is also a major chance for overshoot due to the existence of medium value of the change of error, hence rule 15 uses minimal corrective action by using a low value (h_{qs_L}) for the change of stator quadrature voltage command. In summary, these rules provide a method to determine change of stator direct and quadrature voltage commands that are then integrated to obtain direct and quadrature-axis fuzzy stator voltage feedback commands according to the induction machine operating conditions.

If Ω represents the total number of fuzzy rules (Ω=18 in our case) and $\mu_j(x_i, \tilde{\mathbf{A}}_{i,j})$ represents the combined membership value from the antecedent of the jth rule, the outputs, $\Delta u_{ds}(n)$ and $\Delta u_{qs}(n)$, of the fuzzy *d-q* controller can be written as follows when the centroid defuzzifier is employed [22]:

$$\Delta u_{ds}(n) = \frac{\sum_{j=1}^{\Omega} \mu_j(x_1, \tilde{\mathbf{A}}_{1,j}) \mu_j(x_2, \tilde{\mathbf{A}}_{2,j}) \mu_j(x_3, \tilde{\mathbf{A}}_{3,j}) \mu_j(x_4, \tilde{\mathbf{A}}_{4,j}) \tilde{h}_{ds\_j}}{\sum_{j=1}^{\Omega} \mu_j(x_1, \tilde{\mathbf{A}}_{1,j}) \mu_j(x_2, \tilde{\mathbf{A}}_{2,j}) \mu_j(x_3, \tilde{\mathbf{A}}_{3,j}) \mu_j(x_4, \tilde{\mathbf{A}}_{4,j})} \tag{4.33}$$

$$\Delta u_{qs}(n) = \frac{\sum_{j=1}^{\Omega} \mu_j(x_1, \tilde{\mathbf{A}}_{1,j}) \mu_j(x_2, \tilde{\mathbf{A}}_{2,j}) \mu_j(x_3, \tilde{\mathbf{A}}_{3,j}) \mu_j(x_4, \tilde{\mathbf{A}}_{4,j}) \tilde{h}_{qs\_j}}{\sum_{j=1}^{\Omega} \mu_j(x_1, \tilde{\mathbf{A}}_{1,j}) \mu_j(x_2, \tilde{\mathbf{A}}_{2,j}) \mu_j(x_3, \tilde{\mathbf{A}}_{3,j}) \mu_j(x_4, \tilde{\mathbf{A}}_{4,j})} \tag{4.34}$$

where x_i represents all the inputs ($i = 1..4$) and $\tilde{\mathbf{A}}_{i,j}$ is a vector involving all the input fuzzy sets and $\tilde{h}_{ds\_j}$ represents the direct-axis output fuzzy set for the j-th rule and $\tilde{h}_{qs\_j}$ represents the quadrature-axis output fuzzy set for the j-th rule. Analyzing the rules shown in Table 4.1, it is clear that Δu_{ds} does not utilize inputs, $x_3(n)$ and $x_4(n)$, and Δu_{qs} does not utilize inputs, $x_1(n)$ and $x_2(n)$, hence equations (4.33) and (4.34) can be simplified as follows:

$$\Delta u_{ds}(n) = \frac{\sum_{j=1}^{\Omega'=9} \mu_j(x_1, \tilde{\mathbf{A}}_{1,j}) \mu_j(x_2, \tilde{\mathbf{A}}_{2,j}) \tilde{h}_{ds\_j}}{\sum_{j=1}^{\Omega'=9} \mu_j(x_1, \tilde{\mathbf{A}}_{1,j}) \mu_j(x_2, \tilde{\mathbf{A}}_{2,j})} \tag{4.35}$$

$$\Delta u_{qs}(n) = \frac{\sum_{j=10}^{\Omega''=18} \mu_j(x_3, \tilde{\mathbf{A}}_{3,j}) \mu_j(x_4, \tilde{\mathbf{A}}_{4,j}) \tilde{h}_{qs\_j}}{\sum_{j=10}^{\Omega''=18} \mu_j(x_3, \tilde{\mathbf{A}}_{3,j}) \mu_j(x_4, \tilde{\mathbf{A}}_{4,j})} \tag{4.36}$$

Using (4.24) to (4.29) and (4.35) and (4.36), the complete fuzzy controller system for the induction machine control can be described by the following equation:

$$\bar{V}_{ds}(n) = K_{ff\_ds}(\bar{I}_{ds}(n) - I_{ds}(n)) + \sum_{i=1}^{n} \left((\bar{I}_{ds}(n) - I_{ds}(n)) \frac{\sum_{j=1}^{\Omega'=9} \mu_j(x_1, \tilde{\mathbf{A}}_{1,j}) \mu_j(x_2, \tilde{\mathbf{A}}_{2,j}) \tilde{h}_{ds\_j}}{\sum_{j=1}^{\Omega'=9} \mu_j(x_1, \tilde{\mathbf{A}}_{1,j}) \mu_j(x_2, \tilde{\mathbf{A}}_{2,j})} T_s \right) \tag{4.37}$$

$$\bar{V}_{qs}(n) = K_{ff\_qs}(\bar{I}_{qs}(n) - I_{qs}(n)) + \sum_{i=1}^{n}\left((\bar{I}_{qs}(n) - I_{qs}(n))\frac{\sum_{j=10}^{\Omega''=18}\mu_j(x_3, \tilde{\mathbf{A}}_{3,j})\mu_j(x_4, \tilde{\mathbf{A}}_{4,j})\tilde{h}_{qs\_j}}{\sum_{j=10}^{\Omega''=18}\mu_j(x_3, \tilde{\mathbf{A}}_{3,j})\mu_j(x_4, \tilde{\mathbf{A}}_{4,j})}\right)T_s \quad (4.38)$$

To evaluate the improvements from the fuzzy *d-q* controller, a simulation environment to study the classical *d-q* controller and the fuzzy *d-q* controller is needed. It is important to emphasize that if a classical controller was instead used, then the fuzzy logic-based *d-q* controller in Figure 4.5 will be replaced by the classical *d-q* controller block from Figure 4.4.

4.4. ARCHITECTURE DEVELOPMENT

MATLAB-SIMULINK tools are used for model architecture development and simulations. This model is detailed enough that it can be used for evaluating the robustness of the fuzzy controller for the transient and steady-state behavior of the induction machine. In the next chapter, a description of the simulation environment is presented, and followed by the simulation results using the classical controller and the fuzzy controller.

5. IMPLEMENTATION OF THE FUZZY CONTROLLER AND SIMULATION RESULTS

5.1. INTRODUCTION

In the previous chapters, the model of induction machine plant and the model of induction machine control system were presented. In this chapter, using MATLAB-SIMULINK, simulations are performed to evaluate the performance of the proposed fuzzy logic controller. The fuzzy controller is evaluated for its control of torque and current response of the induction machine. The performance of the proposed fuzzy logic controller is then compared to that of the classical controller.

5.2. SIMULATION ENVIRONMENT

In order to study the improvements from the fuzzy logic-based *d-q* controller; it is necessary to create a simulation environment and integrate the fuzzy logic-based *d-q* controller with operator input, and the developed plant model. The complete simulation environment for the induction machine system will therefore consist of: the operator inputs, induction machine controller model and the plant model of the induction machine. The model for the complete induction machine system under simulation is developed by integrating the developed subsystems' mathematical representations and models in a hierarchical architecture, shown in Figure 5.1.

The operator inputs block in Figure 5.1 represents the operator choice of either

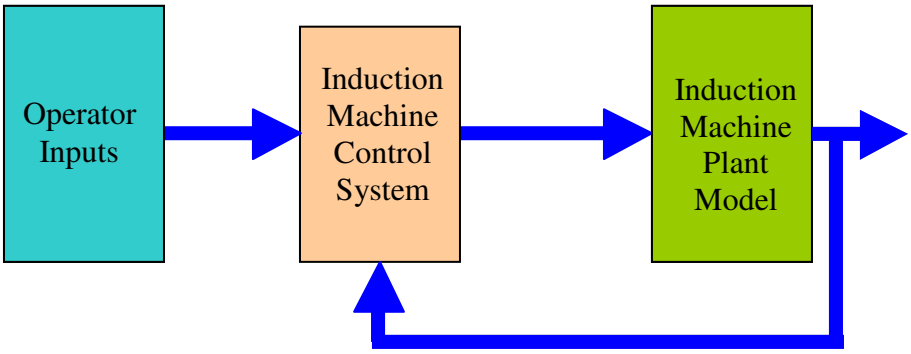


Figure 5.1. Induction machine system simulation model

operating the induction machine in torque control mode or speed control mode. This operator selection is usually dependent on the specific application requirement. In an electric vehicle, the driver inputs such as accelerator and brake pedal commands are considered torque commands and the controller would operate the induction machine in torque control mode. In the same way, when the electric vehicle is operated in cruise control, the controller provides speed regulation of the electric vehicle and controls the induction machine in speed control mode. Through the operator inputs block, operator commands such as desired speed for speed control or desired torque for torque control are issued. The operator inputs block can be configured to follow a wide selection of machine loads based on applications such as elevators, electric vehicles, high-speed spindles, etc. The induction machine plant model block contains the induction machine dynamic *d-q* model, load characteristics model, and necessary voltage and current *d-q* forward and inverse transformation blocks. The induction machine control system consists of the proposed fuzzy logic-based *d-q* controller, rotor flux estimator block, the *d-q* transformation and inverse transformation blocks and the necessary blocks for

determination of desired motor torque, $\overline{T}_m$, required rotor flux, $\overline{\lambda}_{dr}$, and desired stator component currents, $\overline{I}_{ds}$ and $\overline{I}_{qs}$.

5.3. PERFORMANCE EVALUATION OF FUZZY CONTROLLER USING SIMULATION ENVIRONMENT

To study the improvements of the fuzzy controller, it is imperative to compare it to the classical PI controller. For the performance evaluation of the classical controller, the fuzzy logic-based *d-q* controller of Figure 4.5 is replaced with the classical *d-q* controller of Figure 4.4. In order to evaluate the effectiveness of the controllers, simulations are performed with the induction machine operating in speed and torque control mode.

In the simulation environment, the induction machine characteristics shown in Table 3.1 are used. The load characteristics are chosen according to equation (3.36). For the simulation tests in speed control mode, the constants are chosen as follows: $b_0 = 43$, $b_1 = 0$ and $b_3 = 0$. This results in a constant load of 43 Nm, which is applied at 0.5 s. For the simulation tests in the torque control mode, the constants are chosen as follows: $b_0 = 0.1$, $b_1 = 0.002$ and $b_3 = 0.00006$. For the purpose of this simulation, the friction torque, T_{fr} in equation (3.38), is neglected for both speed and torque control mode. The total effective system inertia for the plant model used in the simulations is 0.0639 kg-m^2.

After numerous iterations to fine tune the gain parameters for the classical controller, the proportional and integral gains for the conventional PI controller are set as

follows: $K_{p\_ds} = 3$, $K_{p\_qs} = 3$, $K_{i\_ds} = 170$ and $K_{i\_qs} = 170$. For the fuzzy controller, the following proportional gains are used; $K_{ff\_ds} = 3$, and $K_{ff\_qs} = 3$. Note that these gain values are used as reference for simulation purposes to compare the classical and fuzzy logic-based controller. The fuzzy outputs, F_{fz_ds} and F_{fz_qs}, of the fuzzy controller are based on rules mentioned in Table 4.1, and per equations (4.25) and (4.28).

As mentioned in Chapter 4, there are always parameter mismatches between the actual machine parameters and the estimated machine parameters used in the controller. This parameter mismatch can be as large as 30%. Under such conditions, the performance of the classical controller deteriorates and is unacceptable for high performance applications. In order to evaluate the robustness of the proposed fuzzy controller under such parameter mismatches, the estimated values of the stator inductance, $\tilde{L}_s$, rotor inductance, $\tilde{L}_r$, and mutual inductance, $\tilde{L}_m$, used in the controller are increased by 30% compared to the actual values in Table 3.1. Finally, simulations using MATLAB-SIMULINK are performed using the classical and fuzzy controller. The performance of the controllers is then evaluated for the two operational conditions; speed control mode and torque control mode.

5.4. PERFORMANCE EVALUATION UNDER SPEED CONTROL

To test the effectiveness of the fuzzy controller versus the classical controller under speed control mode, simulations are performed using a custom step load test. In this test, the controller is set at speed control mode with the desired speeds of 2000 r/min,

5000 r/min and 8000 r/min. In this test, the induction machine is initially operating a constant load of 3 Nm at the desired r/min. Next, a step change in load to 43 Nm is applied. This causes the controller to increase desired motor torque to maintain desired speed. With the instant increase in desired motor torque to maintain desired speed, the desired stator component currents, $\bar{I}_{ds}$ and $\bar{I}_{qs}$, are determined and need to be maintained by commanding appropriate, $\overline{V}_{ds}$ and $\overline{V}_{qs}$, through the use of appropriate *d-q* controller (classical and fuzzy *d-q* controllers). This custom step load test emphasizes the behavior of actual mechanical motor torque, T_m, and actual stator component currents, I_{ds} and I_{qs}, in speed control mode.

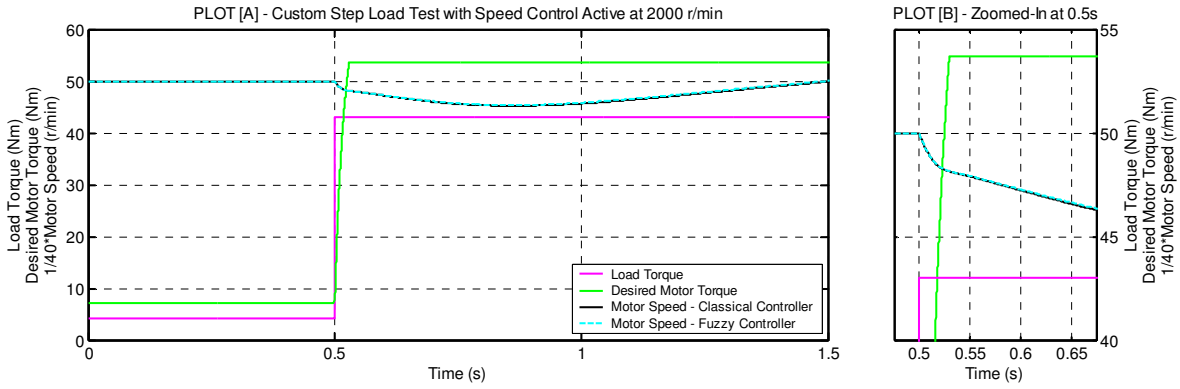


Figure 5.2. Response of desire torque, $\overline{T}_m$, and motor speed, ω_m, for a step load at 2000 r/min

Figure 5.2 shows the results from such a step load test at 2000 r/min. Plot [A] of Figure 5.2 shows the desired motor speed, desired motor torque and the load torque during this test for both classical and fuzzy controllers. Note that before 0.5 s, the load is at 3 Nm and the motor speed is at 2000 r/min. At 0.5 s, the load is stepped from 3 Nm to 43 Nm. This causes the speed of the motor to drop and the speed control portion of "determine desired torque controller block" to increase the desired motor torque to

achieve the desired speed target of 2000 r/min. Plot [B] of Figure 5.2 shows the zoomed-in portion of the test around 0.5 s. It is clear from this test that the speed controller is not impacted by the use of the classical or fuzzy controller, which is a pre-requisite for the performance evaluation of key elements of classical *d-q* vs. fuzzy *d-q* controller.

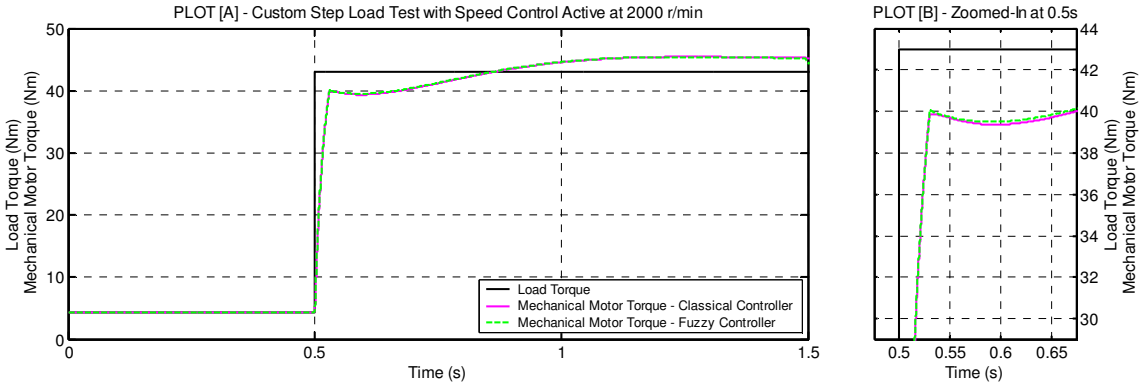


Figure 5.3. Response of motor torque, T_m, for a step load at 2000 r/min

Since speed control is not impacted, it would be imperative to compare the actual mechanical torque using both classical and fuzzy controllers in speed control mode. Figure 5.3 shows the simulation results of actual motor mechanical torque using classical PI controller and fuzzy *d-q* controller. Plot [A] of Figure 5.3 shows the actual mechanical torque behavior during this test. It is clear from plot [A] of Figure 5.3, that the actual torque response for the classical and the fuzzy controller is very close at 2000 r/min, even in the presence of parameter mismatch of 30%. However, plot [B] of Figure 5.3 shows that the actual motor mechanical torque response during the initial transient around 0.5 s with the fuzzy controller is only very slightly better than the classical controller under these conditions. Since the behavior of the actual motor torque is known, the torque producing component of the stator current (quadrature-axis current),

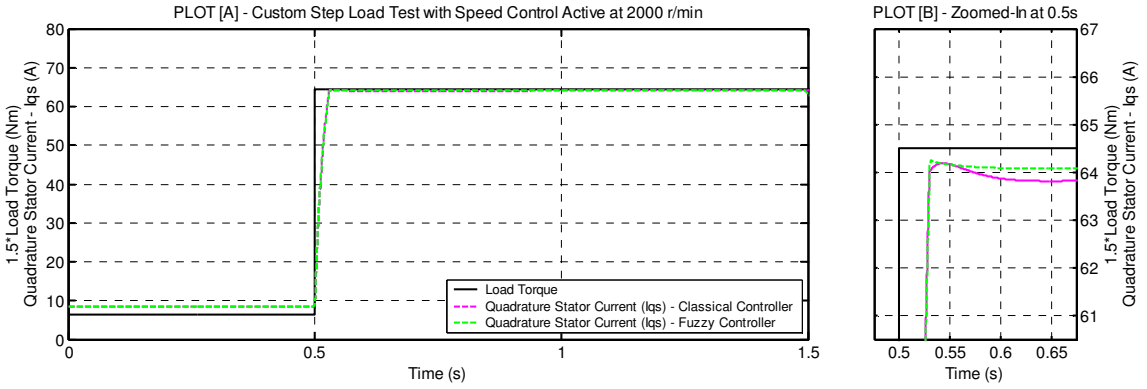


Figure 5.4. Response of stator quadrature-axis current, I_{qs}, for a step load at 2000 r/min

I_{qs}, must be examined. Figure 5.4 shows the behavior of stator quadrature-axis current, I_{qs}, for the custom step load test at 2000 r/min. Plot [A] of Figure 5.4 shows that the I_{qs} current response is very similar for the classical controller and the fuzzy controller. Again, looking at the zoomed-in plot [B] of Figure 5.4, it is seen that the response of the fuzzy controller is slightly better than that of the classical control. Also, the fuzzy controller exhibits strong non-linear behavior, such that the current, I_{qs}, rises quickly in almost a straight-line manner around the target set point and then changes its direction and settles to the desired set point in less than 0.02s. Conversely, the classical controller exhibits typical linear behavior with overshoots, undershoots and relatively large settling time of greater than 0.15s. The direct-axis current, I_{ds}, which is the flux producing component of the stator current, can have a slight impact on the motor torque, since the motor flux can vary in transient. The goal of both classical and fuzzy controller is to minimize the changes in I_{ds}. Figure 5.5 shows the behavior of stator direct-axis current,

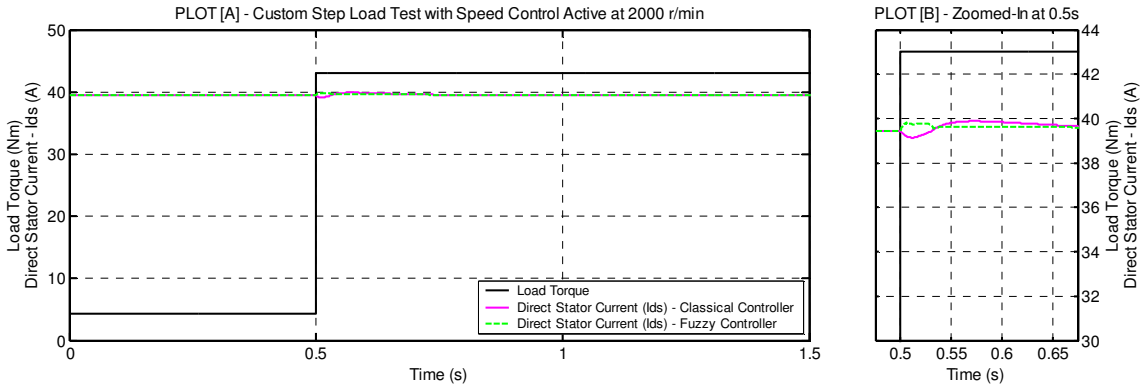


Figure 5.5. Response of stator direct-axis current, I_{ds}, for a step load at 2000 r/min

I_{ds}, for the custom step load test at 2000 r/min. As can be seen from the plot [A] of Figure 5.5, there is a small change in the actual value of the stator current, I_{ds}, using the classical controller. The zoomed-in plot [B] of Figure 5.5 clearly shows that the stator direct-axis current, I_{ds}, for the fuzzy controller has an improved behavior with a very small overshoot of less than 0.3% and a small settling time of less than 0.03 s. The classical controller on the other hand has a settling time of about 0.15 s and exhibits the expected linear control issues of current undershoots and overshoots of approximately 1.3% and 0.6% respectively. The improvements in the stator component current, I_{ds} and I_{qs}, using the fuzzy controller are achieved directly as a result of the stator component voltage output commands, $\overline{V}_{ds}$ and $\overline{V}_{qs}$ of the fuzzy controller. Figures 5.6 and 5.7 show the comparison of stator component voltage commands, $\overline{V}_{ds}$ and $\overline{V}_{qs}$, for the fuzzy controller and the classical controller.

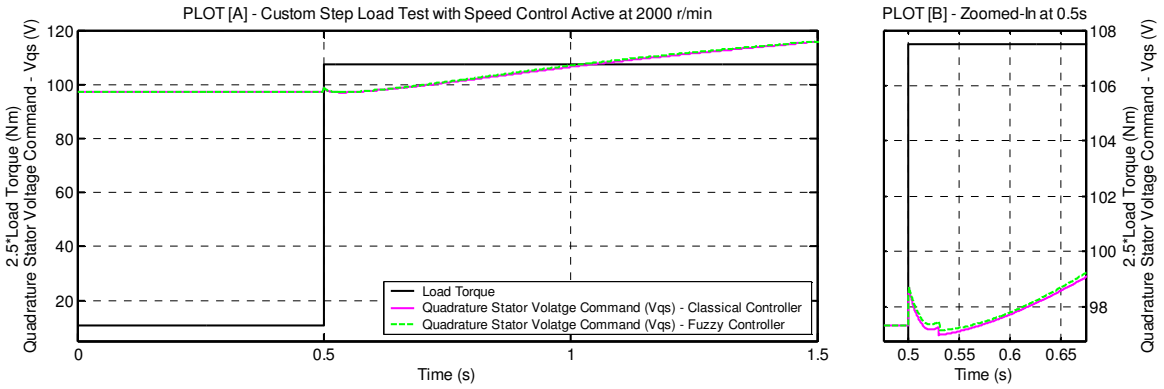


Figure 5.6. Stator quadrature voltage command, $\overline{V}_{qs}$, for a step load at 2000 r/min

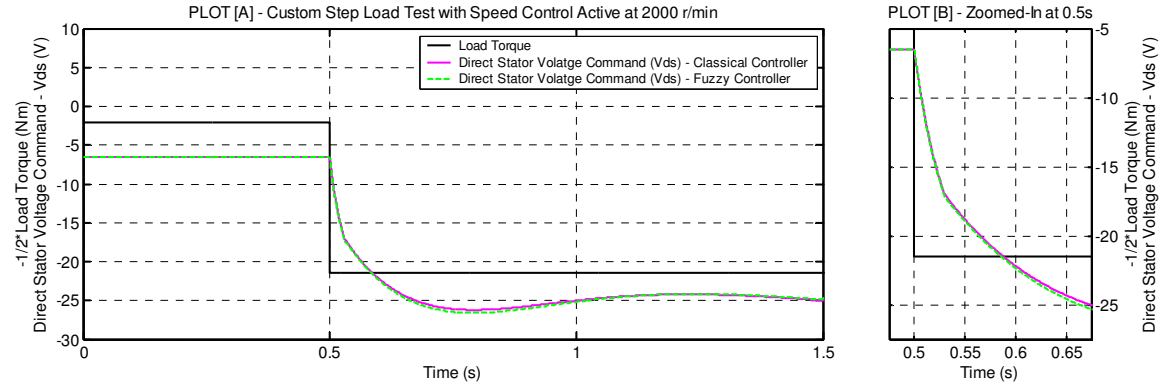


Figure 5.7. Stator direct voltage command, $\overline{V}_{ds}$, for a step load at 2000 r/min

The use of the fuzzy controller during the custom step load at low machine speed of 2000 r/min resulted in slight improvement over the classical controller, so more improvements are expected at medium speeds. Next, the same type of test is conducted at the medium motor speed of 5000 r/min.

Figure 5.8 shows the results from a custom step load test conducted at 5000 r/min. Figure 5.8 again clearly shows that the speed controller is not impacted by the use of the

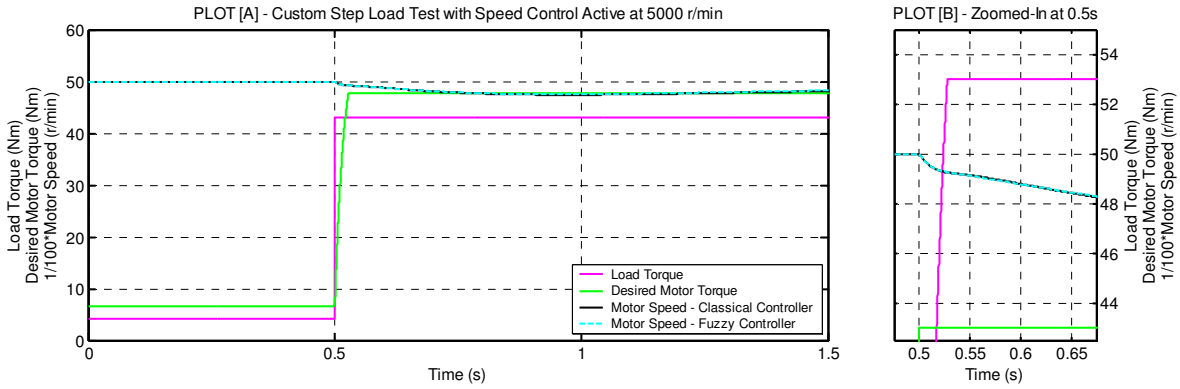


Figure 5.8. Response of desire torque, $\overline{T}_m$, and motor speed, ω_m, for a step load at 5000 r/min

classical or fuzzy controller, and this allows the comparison of the performance of key elements of classical *d-q* controller vs. fuzzy *d-q* controller in speed control mode. Figure 5.9 shows the simulation results of actual motor mechanical torque using classical *d-q* controller and fuzzy *d-q* controller at 5000 r/min. Plot [A] of Figure 5.9 shows the actual mechanical torque behavior during this test. At 5000 r/min test, it can be noticed

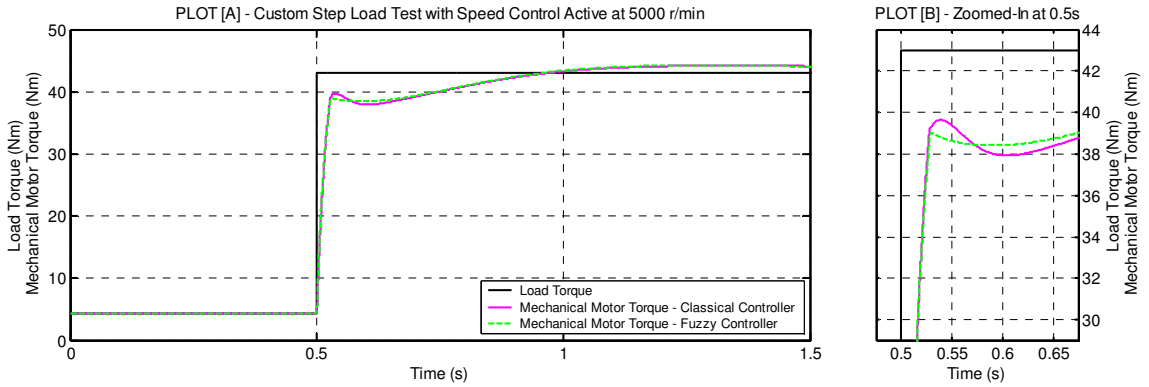


Figure 5.9. Response of motor torque, T_m , for a step load at 5000 r/min

from the plot [A] of Figure 5.9 that the actual torque response for the fuzzy controller is comparatively better than that of the classical controller. It can be clearly seen from plot

[B] of Figure 5.9 that the actual motor mechanical torque response for the fuzzy controller is robust with no overshoots or undershoots. It is shown that the torque rises quickly in almost a straight-line manner around the target set point, then changes its direction and settles to the desired set point in less than 0.05s. In comparison, the actual motor mechanical torque response for the classical controller exhibits overshoot of about 1 Nm or 2% and undershoot of about 0.5 Nm or 1.2% in the torque response with much larger settling times. Next, the response of the torque and flux producing components of stator currents, I_{qs} and I_{ds}, respectively, for the fuzzy controller vs. the classical controller is compared. Figure 5.10 shows the response of the stator component current, I_{qs}, for custom step load test at 5000 r/min using the fuzzy controller and the classical controller. Plot [A] of Figure 5.10 shows that the I_{qs} current response for the fuzzy controller shows greater improvements compared to the classical controller.

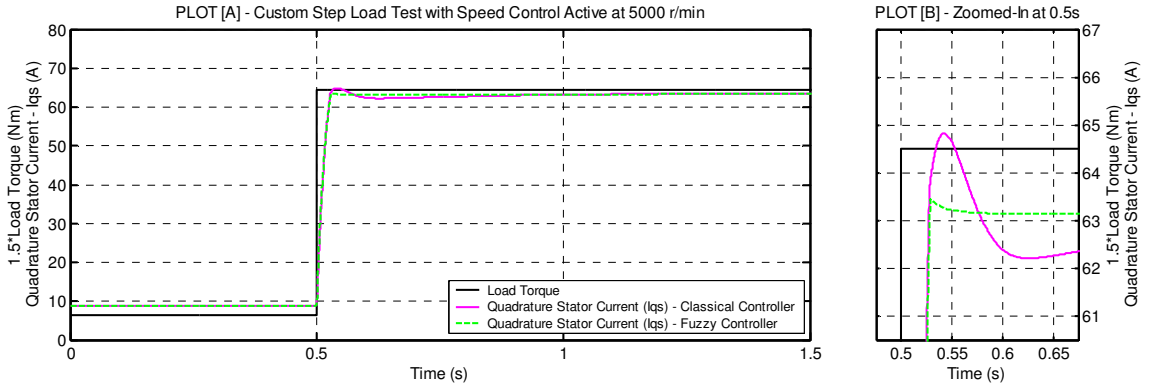


Figure 5.10. Response of stator quadrature-axis current, I_{qs}, for a step load at 5000 r/min

Again, looking at the zoomed-in plot [B] of Figure 5.10, we see that the response of the fuzzy controller is robust with minimal overshoot (0.3%). On the other hand, the classical controller exhibits I_{qs} current overshoot of about 1.5 amps or 2.5%, and

undershoot of about 1 amps or 1.4%, with settling time in excess of 0.4 s. Again, the fuzzy controller showed strong non-linear behavior; the stator component current, I_{qs}, rose quickly in almost a straight-line manner, then changed its direction and settled to the desired set point in less than 0.02s.

The response of the stator component current, I_{ds}, for the custom step load test at 5000 r/min, is shown in Figure 5.11. Again, from Figure 5.11 a noticeable improvement

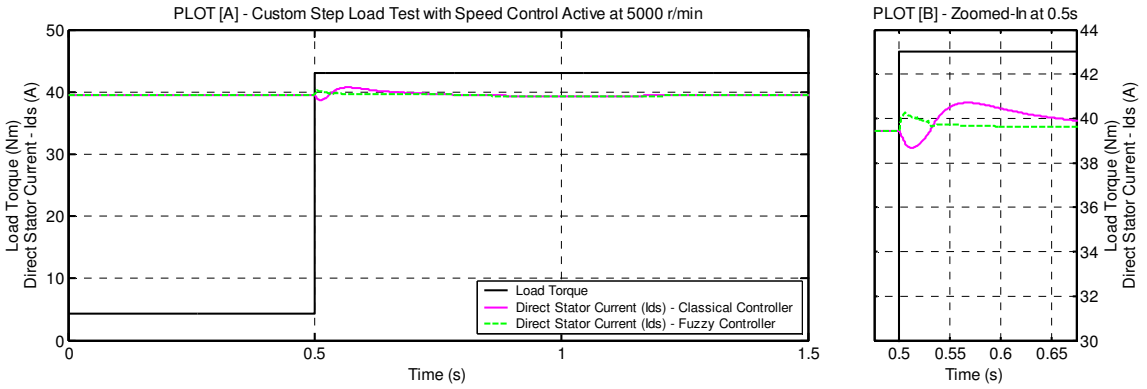


Figure 5.11. Response of stator direct-axis current, I_{ds}, for a step load at 5000 r/min

can be seen in the response of stator component current, I_{ds}, for the fuzzy controller when compared to the classical controller. From the plot [B] of Figure 5.11, it is apparent that for the fuzzy controller, the change in stator component current, I_{qs}, caused an initial small disturbance of less than 0.1% in the stator component current, I_{ds}. However, the fuzzy controller immediately corrected this error and brought the stator component current, I_{ds}, to a constant value in a very short time. The zoomed-in plot [B] of Figure 5.11 clearly shows that the response of stator direct-axis current, I_{ds}, for the fuzzy controller was much improved and steady. The classical controller, on the other hand, had issues of current undershoot and overshoot of as much as 1 amp or 2.5%, and showed

a large settling time of greater than 0.15 s. Finally, Figures 5.12 and 5.13 show the comparison of stator component voltage commands, $\overline{V}_{ds}$ and $\overline{V}_{qs}$, for the fuzzy controller and the classical controller.

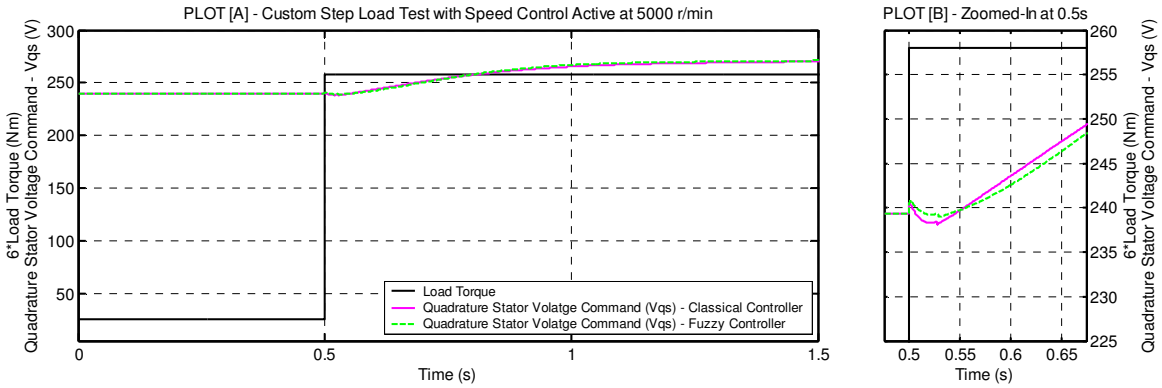


Figure 5.12. Stator quadrature voltage command, $\overline{V}_{qs}$, for a step load at 5000 r/min

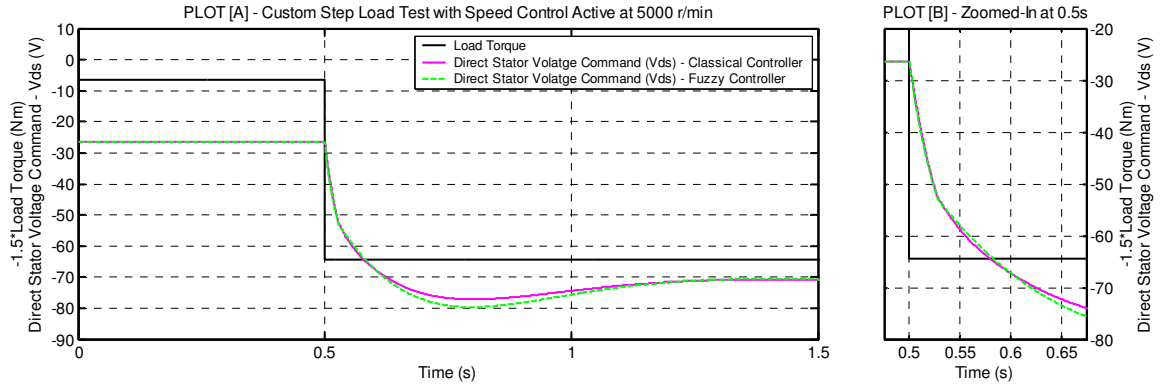


Figure 5.13. Stator direct voltage command, $\overline{V}_{ds}$, for a step load at 5000 r/min

Using the fuzzy controller during the custom step load revealed a noticeable improvement over the classical controller at medium machine speed of 5000 r/min. It is anticipated that even greater improvements in torque and stator current behavior will occur for the machine operating at high-speed of 8000 r/min.

Figure 5.14 shows the results from a custom step load test at 8000 r/min. It can be seen from the Figure 5.14 that the performance of the speed controller is same for both

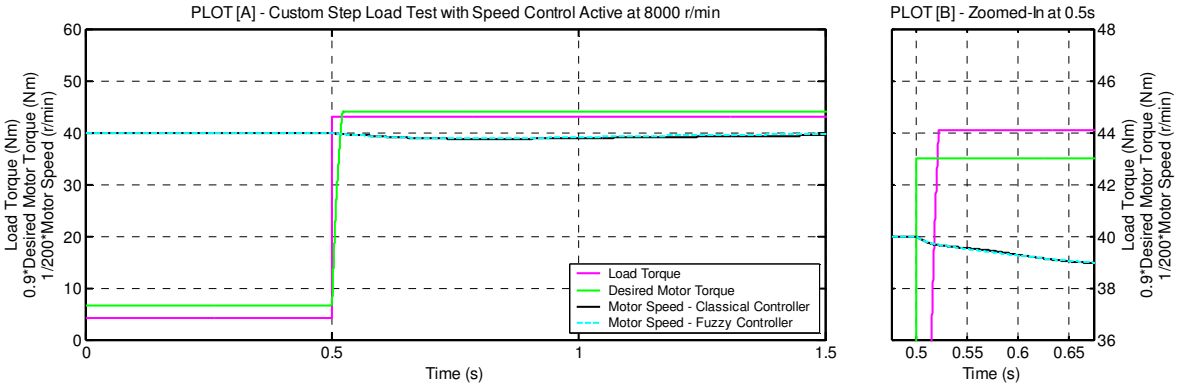


Figure 5.14. Response of desire torque, $\overline{T}_m$, and motor speed, ω_m, for a step load at 8000 r/min

the classical and the fuzzy controllers, and this enables the comparison of the performance of key elements of classical *d-q* controller vs. fuzzy *d-q* controller in speed control mode. Figure 5.15 shows the simulation results of actual motor torque using the classical *d-q* controller and the fuzzy *d-q* controller at 8000 r/min. Plot [A] of Figure 5.15 shows the actual mechanical torque behavior during this test. For the step load test

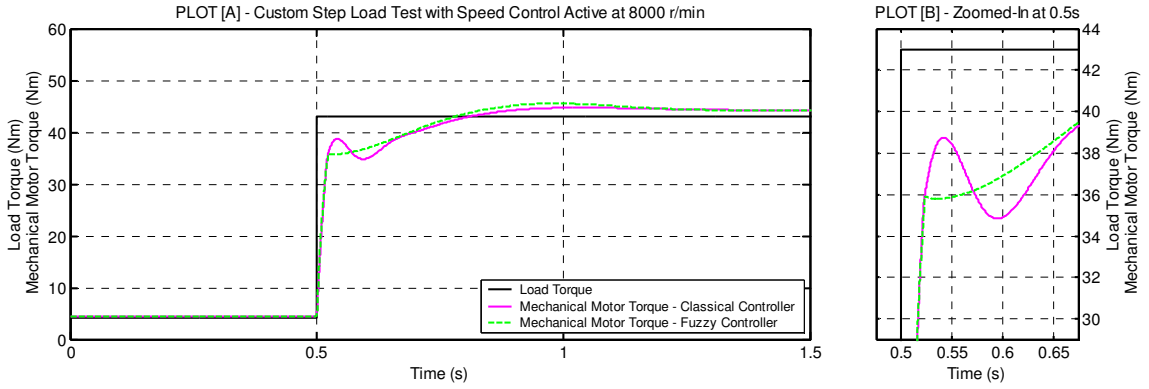


Figure 5.15. Response of motor torque, T_m , for a step load at 8000 r/min

at 8000 r/min, it can be noticed from plot [A] of Figure 5.15, that the actual torque response for the fuzzy controller is significantly better than that of the classical controller. It can be clearly seen from plot [B] of Figure 5.15 that that the actual motor mechanical torque response for the fuzzy controller was robust with no overshoots or undershoots. Whereas, the actual motor mechanical torque response for the classical controller exhibited large overshoot of about 3 Nm or (8%) and undershoot of about 2 Nm or (5%), with larger settling times of around 0.17 s.

Next, the response of the torque and flux producing components of stator current, I_{qs} and I_{ds}, respectively, for the fuzzy controller is compared with the classical controller. Figure 5.16 shows the response of the stator component current, I_{qs}, for custom step load test at 8000 r/min using the fuzzy controller and the classical controller. Plot [A] of

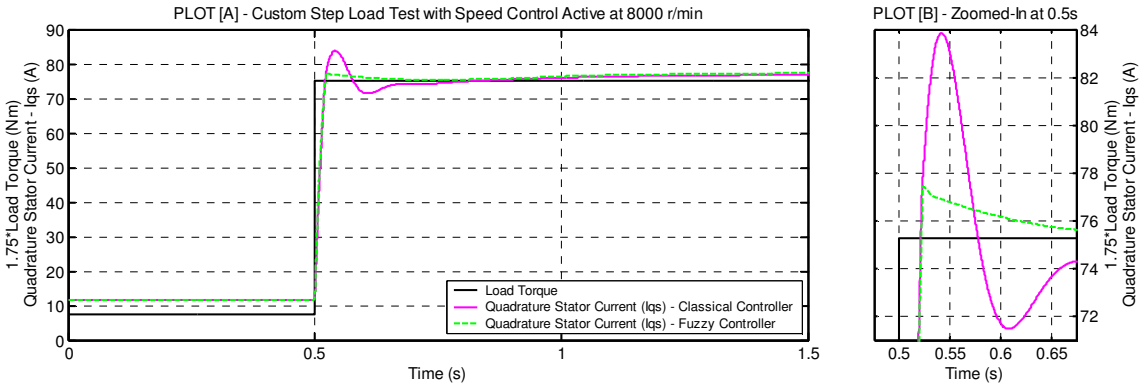


Figure 5.16. Response of stator quadrature-axis current, I_{qs}, for a step load at 8000 r/min

Figure 5.16 shows that the I_{qs} current response for the fuzzy controller significantly improved when compared to that of the classical controller. Again, the zoomed-in plot [B] of Figure 5.16 shows that the response of the fuzzy controller is robust with no overshoots. On the other hand, the classical controller exhibited large I_{qs} current

overshoot of about 7 amps or 9%, and undershoots of about 4.5 amps or 6%, with settling time in excess of 0.4 s. Once again, the fuzzy controller displayed a strong non-linear behavior. The stator component current, I_{qs}, rose quickly in almost a straight-line manner, then changed its direction and settled to the desired set point in less than 0.02s. The response of the stator component current, I_{ds}, for the custom step load test at 8000 r/min, is shown in Figure 5.17.

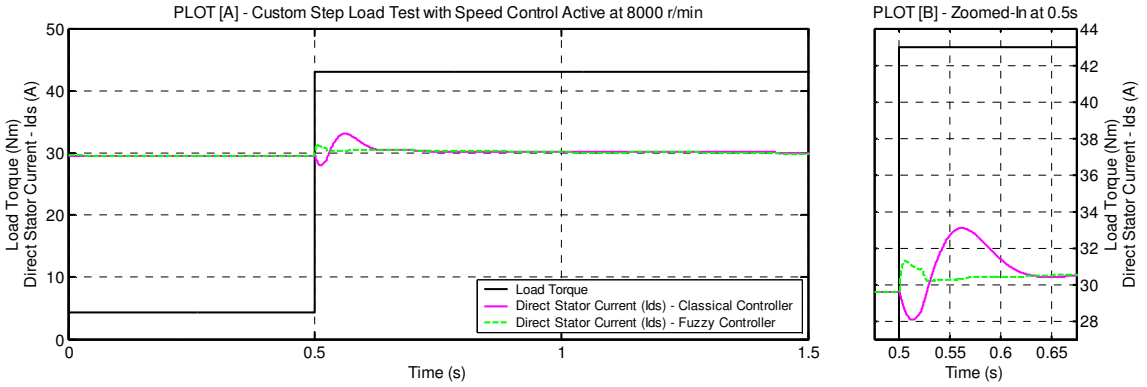


Figure 5.17. Response of stator direct-axis current, I_{ds}, for a step load at 8000 r/min

Again, from the Figure 5.17 a significant improvement is seen in the response of stator component current, I_{ds}, for fuzzy controller when compared to a classical controller. Plot [B] of Figure 5.17 shows that for the fuzzy controller, the change in stator component current, I_{qs}, caused an initial disturbance in the stator component current, I_{ds}, of about 0.75 amps or 2%. However, the fuzzy controller immediately corrected this error and brought the stator component current, I_{ds}, to a constant value in about .03 s. The classical controller, on the other hand, had an initial current undershoot of about 2 amps or 7%, followed by an overshoot of about 3 amps or 10%, with a large settling time of greater than 0.12 s. The difference in the performance of the two

controllers clearly demonstrates the effectiveness of the fuzzy controller. Finally, Figures 5.18 and 5.19 show the comparison of stator component voltage commands, $\overline{V}_{ds}$ and $\overline{V}_{qs}$, for the fuzzy controller and the classical controller.

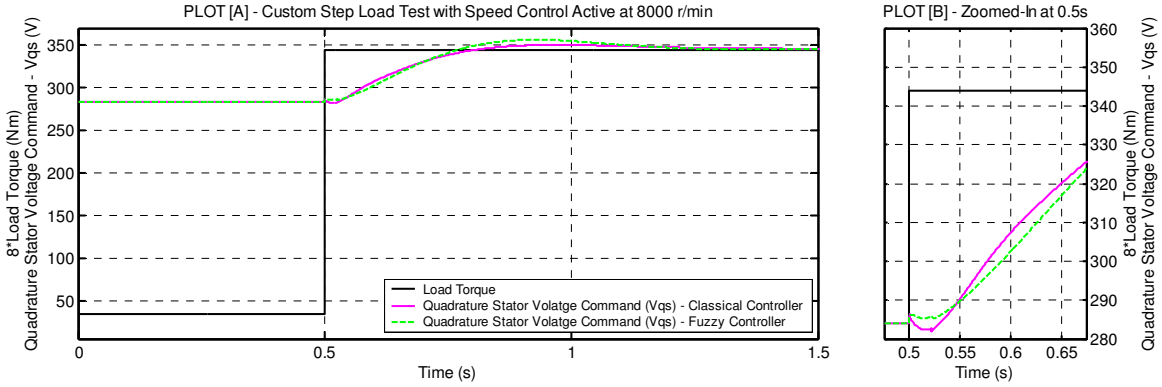


Figure 5.18. Stator quadrature voltage command, $\overline{V}_{qs}$, for a step load at 8000 r/min

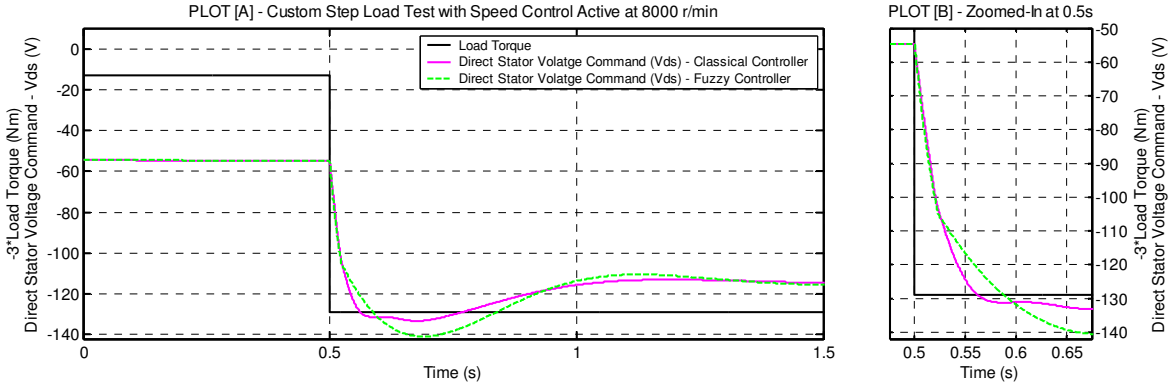


Figure 5.19. Stator direct voltage command, $\overline{V}_{ds}$, for a step load at 8000 r/min

From the custom step load tests conducted at 2000 r/min, 5000 r/min and 8000 r/min and with a 30% parameter mismatch, a clear pattern emerges which shows that as the actual motor speed increased, the performance of fuzzy controller remained robust, while that of the classical controller deteriorated. This is especially true, since the

classical controller uses de-coupling compensation terms to calculate the stator components voltages, $\overline{V}_{ds}$ and $\overline{V}_{qs}$. Clearly, the error in the compensation terms, $-\overline{\omega}(\sigma L_s - \tilde{\sigma}\tilde{L}_s)I_{qs}$ and $\overline{\omega}(L_s - \tilde{L}_s)I_{ds}$, due to parameter mismatches increases as $\overline{\omega}$ increases and together with the use of linear PI controllers, is responsible for the poor response of the classical controller at high-speed. The fuzzy controller instead uses fuzzy logic control, together with a proportional error compensator to determine the appropriate stator component voltages, $\overline{V}_{ds}$ and $\overline{V}_{qs}$. Since this proposed fuzzy controller is non-linear in nature, it can robustly control the stator component currents, I_{ds} and I_{qs}.

It has been shown that in the speed control mode, the proposed fuzzy controller provided a robust control of the actual motor mechanical torque and the stator component currents, I_{ds} and I_{qs}. Next, the robustness of our proposed fuzzy controller is compared with the classical controller in the torque control mode.

5.5. PERFORMANCE EVALUATION UNDER TORQUE CONTROL

To test the effectiveness of the fuzzy controller versus the classical controller under torque control mode, simulations are performed using a custom step torque command. In this test, the controller is set to torque control mode with the induction machine operating at speeds of 2000 r/min, 5000 r/min and 8000 r/min. For the test, the induction machine is operating with an initial speed of 2000 r/min, 5000 r/min and 8000 r/min with the load torque defined by coefficients b_0, b_1 and b_2 in Section 5.2. Next, a step change in motor torque command is applied. This causes the controller to increase desired motor torque, resulting in acceleration of the induction machine. With the instant

increase in desired motor torque for accelerating the induction machine, the desired $\bar{I}_{ds}$ and $\bar{I}_{qs}$ are determined and maintained by commanding appropriate stator component voltages, $\overline{V}_{ds}$ and $\overline{V}_{qs}$, through the use of appropriate *d-q* controller (Classical or Fuzzy *d-q* controllers). This custom step torque command test emphasizes the behavior of actual mechanical motor torque, T_m, and actual stator component currents, I_{ds} and I_{qs}, in torque control mode.

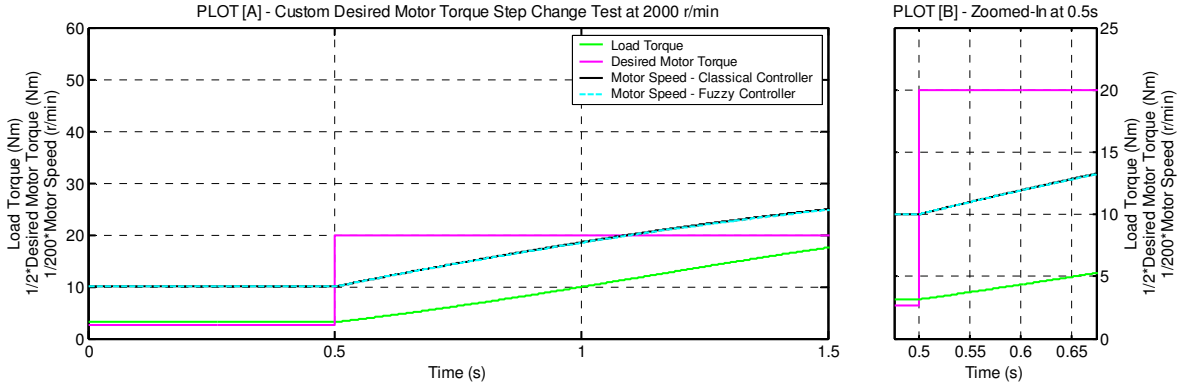


Figure 5.20. Response of desire torque, $\overline{T}_m$, and motor speed, ω_m, for a torque command at 2000 r/min

Figure 5.20 shows the results from such a step torque command test at 2000 r/min. Plot [A] of Figure 5.20 shows that the desired motor torque is stepped from 6 Nm to 40 Nm causing the motor to accelerate from 2000 r/min to 5000 r/min in 1 s. During this acceleration period, the load torque is changed from about 3 Nm to about 18 Nm. Plot [B] of Figure 5.20 shows the zoomed-in portion of the test around 0.5 s. It is clear from this test that the motor speed behavior is not impacted by the use of the classical or

fuzzy controller. This allows for the performance evaluation of key elements of classical *d-q* controller vs. fuzzy *d-q* controller.

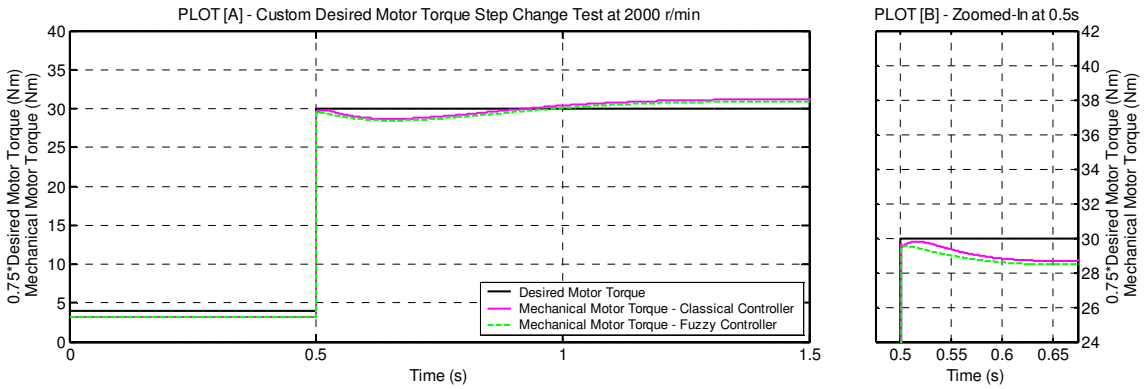


Figure 5.21. Response of motor torque, T_m, for a torque command at 2000 r/min

Figure 5.21 shows the simulation results of actual motor mechanical torque using the classical controller and the fuzzy controller. Plot [A] of Figure 5.21 shows the actual mechanical torque behavior during this test. It is clear from plot [A] of Figure 5.21 that the actual torque response for the classical and fuzzy controllers, even in the presence of parameter mismatch of 30%, is very similar at 2000 r/min. However, plot [B] of Figure 5.21 shows that the actual motor mechanical torque response with the fuzzy controller is slightly better than the classical controller during the initial transient at around 0.5 s.

Next, the torque-producing component of the stator current, which is quadrature-axis current, I_{qs}, is examined. Figure 5.22 shows the behavior of stator quadrature-axis current, I_{qs}, for the step torque command at 2000 r/min. Plot [A] of Figure 5.22 shows that the I_{qs} current response is very similar for the classical controller and the fuzzy controller. However, looking at the zoomed-in plot [B] of Figure 5.22, we see that the response of the fuzzy controller is much better than that of the classical controller.

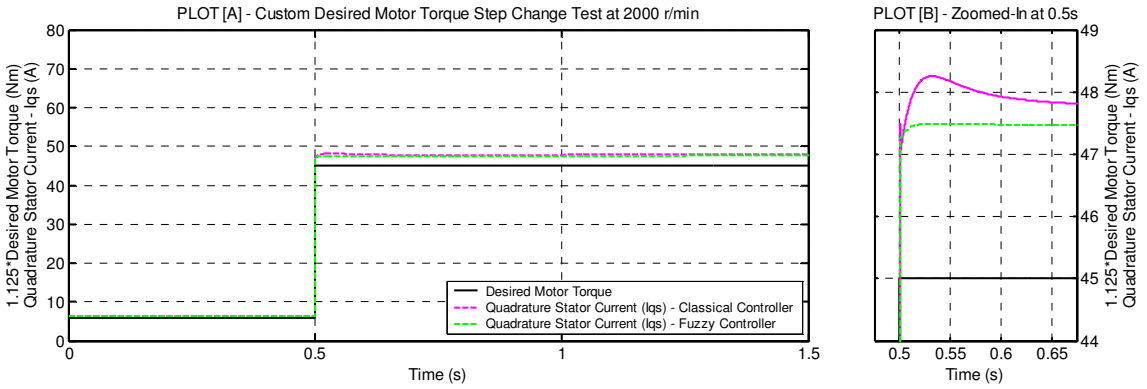


Figure 5.22. Response of stator quadrature-axis current, I_{qs}, for a torque command at 2000 r/min

The stator component current, I_{qs}, rose quickly in almost a straight-line manner around the target set point and then changed its direction and settled to the desired set point in less than 0.02s. Whereas, with the classical controller, there was an initial spike in the I_{qs} current, followed by a relatively small undershoot, and then an overshoot of about 0.75 amps or 1.5%. The controller then slowly corrected the I_{qs} current response with a relatively large settling time of around .3 s.

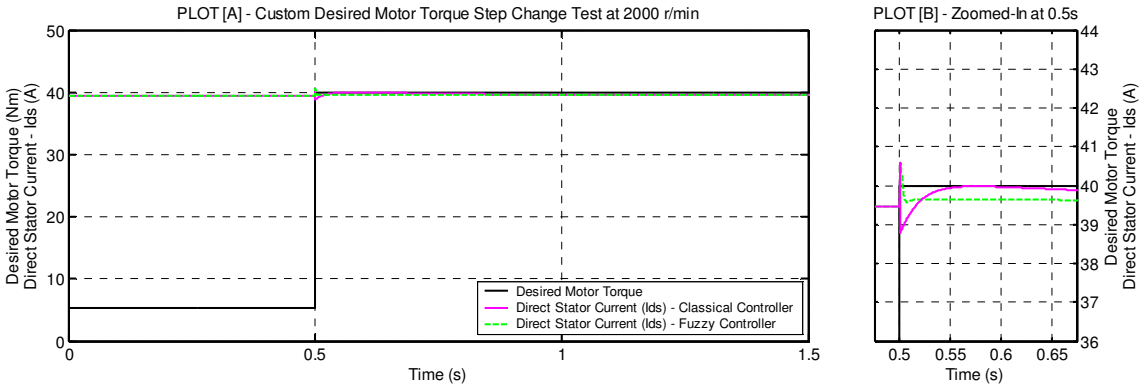


Figure 5.23. Response of stator direct-axis current, I_{ds}, for a torque command at 2000 r/min

Figure 5.23 shows the behavior of stator direct-axis current, I_{ds}, for the step torque command at 2000 r/min. As can be seen from the plot [A] of Figure 5.23, there is a small initial spike in the actual value of the stator current, I_{ds}, with both classical and fuzzy controllers. This initial spike is caused by a large step change in I_{qs} current due to a step change in torque command. However, the zoomed-in plot [B] of Figure 5.23 clearly shows that the stator direct-axis current, I_{ds}, for the fuzzy controller was corrected almost instantly, with a small settling time of less than 0.02 s. The classical controller on the other hand, caused the I_{ds} current to undershoot by about 2.5% after an initial spike,

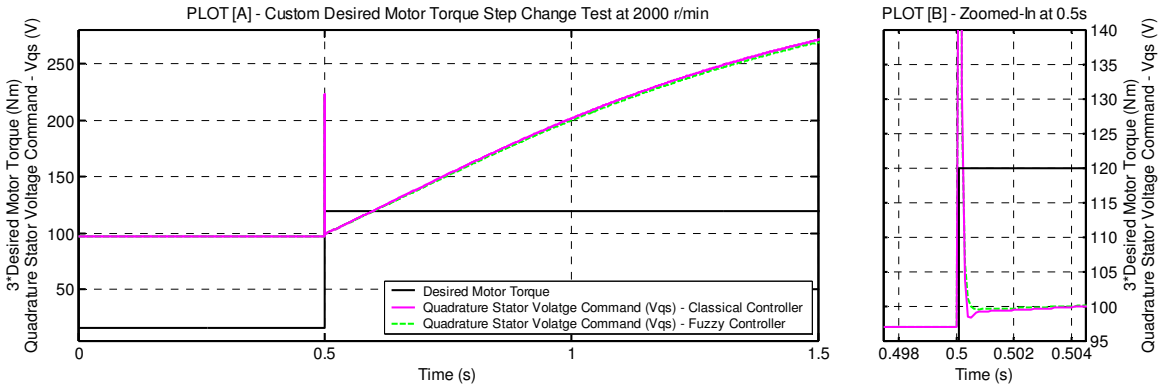


Figure 5.24. Stator quadrature voltage command, $\overline{V}_{qs}$, for a torque command at 2000 r/min

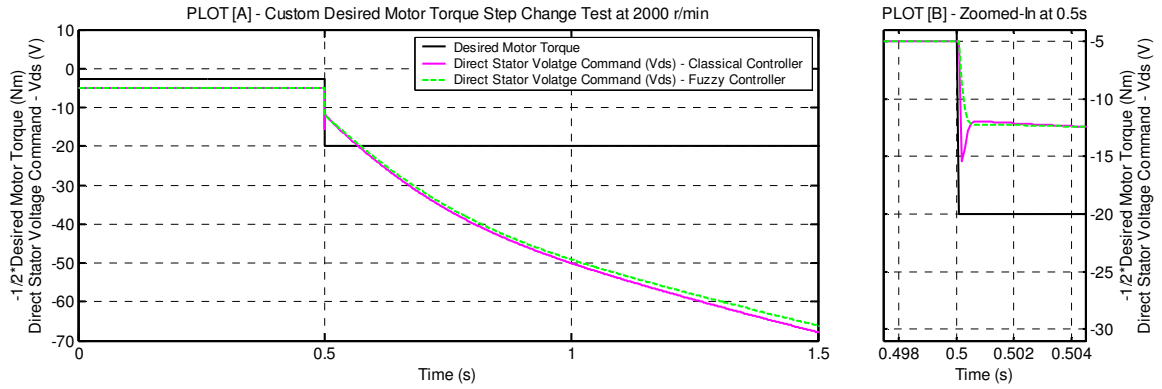


Figure 5.25. Stator direct voltage command, $\overline{V}_{ds}$, for a torque command at 2000 r/min

then slowly corrected the I_{ds} response with a settling time of about 0.7 s. This shows the robustness of the proposed fuzzy controller compared to the classical controller. The improvements in the stator component current, I_{ds} and I_{qs} using the fuzzy controller are achieved directly as a result of the stator component voltage output commands, $\overline{V}_{ds}$ and $\overline{V}_{qs}$, of the fuzzy controller. Figures 5.24 and 5.25 show the comparison of stator component voltage commands, $\overline{V}_{ds}$ and $\overline{V}_{qs}$, for the fuzzy controller and the classical controller.

The use of the fuzzy controller during the custom step torque command at low machine speed of 2000 r/min resulted in slight improvement over the classical controller. Next, the same type of test is conducted at the medium motor speed of 5000 r/min.

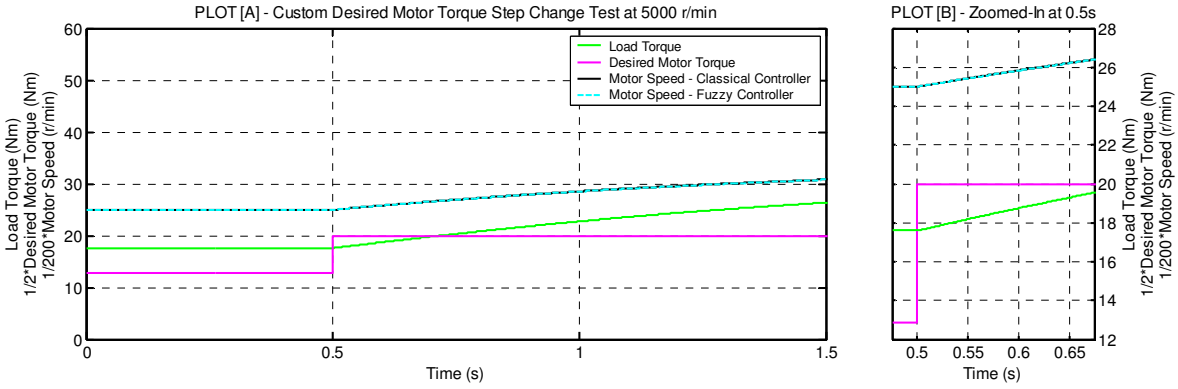


Figure 5.26. Response of desire torque, $\overline{T}_m$, and motor speed, ω_m, for a torque command at 5000 r/min

Figure 5.26 shows the results from such a step torque command test at 5000 r/min. Plot [A] of Figure 5.26 shows that the desired motor torque is stepped from 14 Nm to 40 Nm. This results in the motor accelerating from 5000 r/min to 6000 r/min in less than 1 s. During this acceleration period, the load torque is changed from about 18

Nm to about 27 Nm. Plot [B] of Figure 5.26 shows the zoomed-in portion of the test around 0.5 s. Again, it can be seen that the motor speed behavior is not impacted by the use of the classical or fuzzy controller and this enables an evaluation of the performance of key elements of the classical and fuzzy controllers.

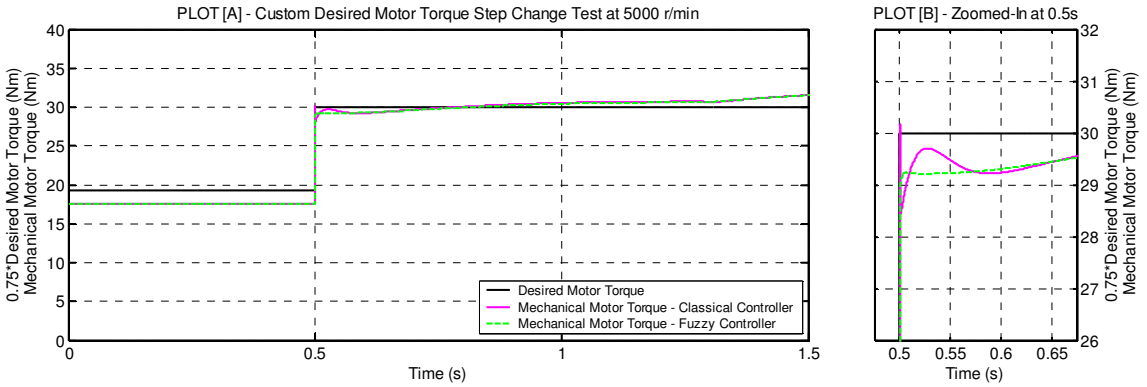


Figure 5.27. Response of motor torque, T_m, for a torque command at 5000 r/min

Figure 5.27 shows the simulation results of actual motor mechanical torque using the classical controller and the fuzzy controller. It is clear from plot [A] of Figure 5.27, that there is a significant difference in the actual torque response for the classical and fuzzy controller at 5000 r/min. The zoomed-in plot [B] of Figure 5.27 shows that the actual motor mechanical torque response for the fuzzy controller does not contain any overshoots or undershoots. The torque response for the classical controller clearly contained torque spikes, as well as undershoot and overshoot of about 2.5% and 1.7% respectively. Thus, better torque response for the fuzzy controller was noticed compared to the classical controller.

Figure 5.28 shows the behavior of stator quadrature-axis current, I_{qs}, for the step torque command at 5000 r/min. Plot [A] of Figure 5.28 shows that the I_{qs} current

response for the fuzzy controller does not contain any overshoots. Looking at the zoomed-in plot [B] of Figure 5.28, it can be seen that the response of the I_{qs} current for the fuzzy controller is much better than that of the classical controller. It can be seen

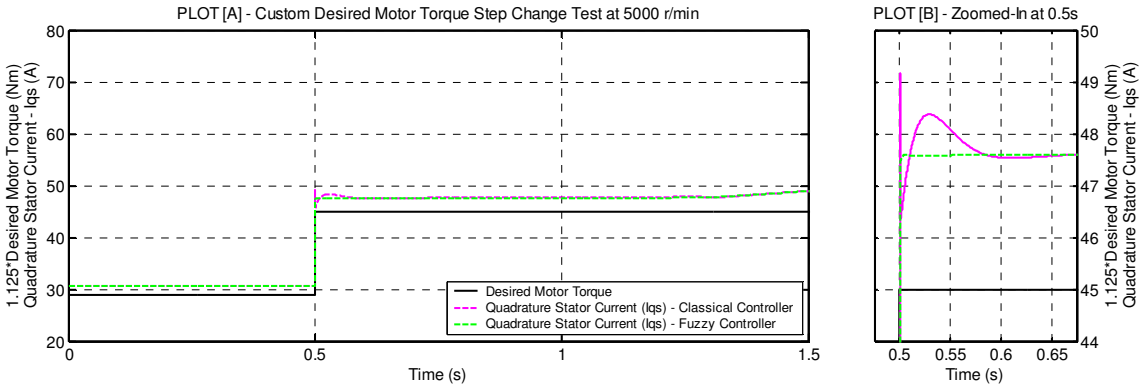


Figure 5.28. Response of stator quadrature-axis current, I_{qs}, for a torque command at 5000 r/min

that stator component current, I_{qs}, rose quickly in almost a straight-line manner around the target set point and then changed its direction and settled to the desired set point in less than 0.02s. The classical controller, on the other hand, caused the quadrature-axis current, I_{qs}, to have an initial spike of about 1.5 amps or 3%, followed by an undershoot

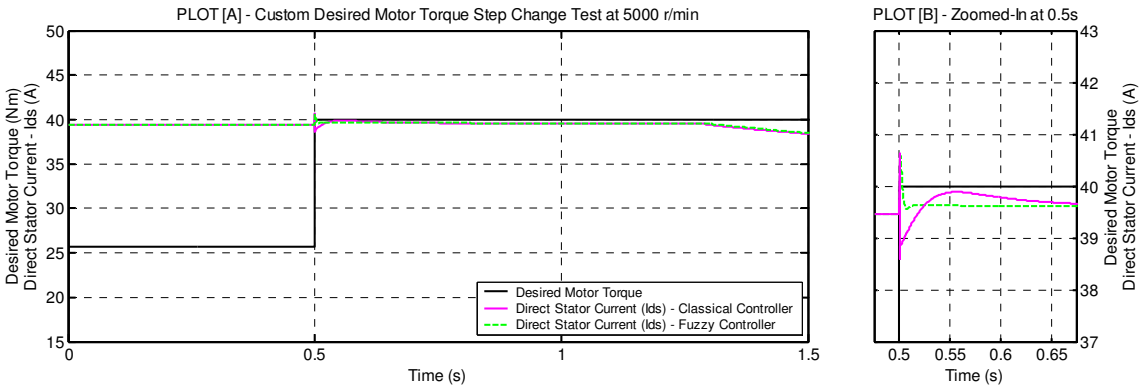


Figure 5.29. Response of stator direct-axis current, I_{ds}, for a torque command at 5000 r/min

of about 1.3 amps or 2%, and then an overshoot of about 1 amps or 1.5%. The controller then slowly corrected the I_{qs} current response with a relatively large settling time of around 0.15 s. It is also noticed that around 6000 r/min, the I_{qs} current starts to increase slowly with motor speed for both the fuzzy controller and the classical controller. This is due to the fact that above 6000 r/min, the motor operates in the field-weakening region, and thus the stator component current, I_{qs} must increase for the same commanded torque.

Figure 5.29 shows the behavior of stator direct-axis current, I_{ds}, for the step torque command at 5000 r/min. As can be seen from the plot [A] of Figure 5.29, there is an initial spike in the actual value of the stator current, I_{ds}, with both classical and fuzzy controllers, caused by a step increase in the I_{qs} current. The zoomed-in plot [B] of Figure 5.29 clearly shows that the stator direct-axis current, I_{ds}, for the fuzzy controller was corrected almost instantly, with a small settling time of less than 0.02 s. As in the case at 2000 r/min, the classical controller caused the I_{ds} current undershoot of about 1 amps or 2.5% after an initial spike, which was followed by an overshoot and another undershoot, then was slowly corrected with a settling time of greater than 0.3 s. Again, it was noticed that around 6000 r/min, the I_{ds} current started to decrease slowly, for both the fuzzy controller and the classical controller, with the increase in motor speed. This is due to the fact that above 6000 r/min, the motor operates in the field-weakening region, and thus the flux producing component of the stator current, I_{ds}, is decreased as well.

Figures 5.30 and 5.31 show that the stator voltage commands, $\overline{V}_{ds}$ and $\overline{V}_{qs}$, for the fuzzy controller do not contain any overshoots or undershoots. Whereas, for the classical control, the stator voltage components, $\overline{V}_{ds}$ and $\overline{V}_{qs}$, contain large overshoots, as well as undershoots, which clearly indicates a poor controller response.

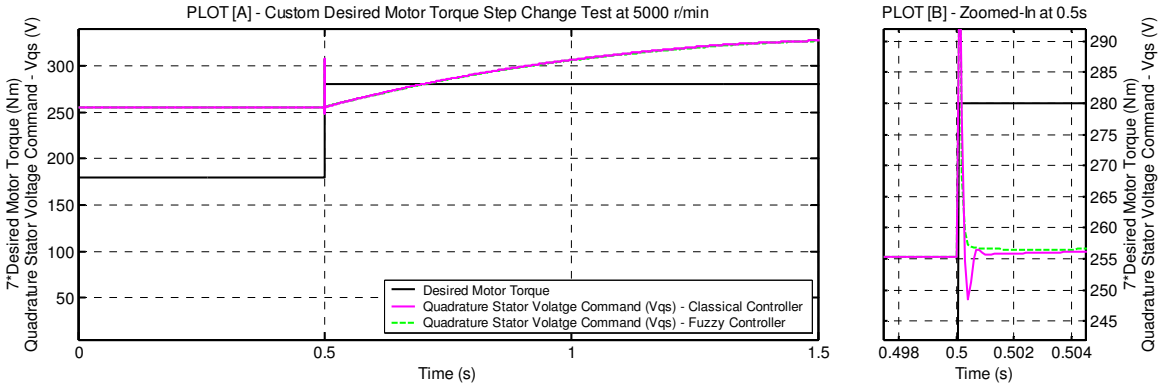


Figure 5.30. Stator quadrature voltage command, $\overline{V}_{qs}$, for a torque command at 5000 r/min

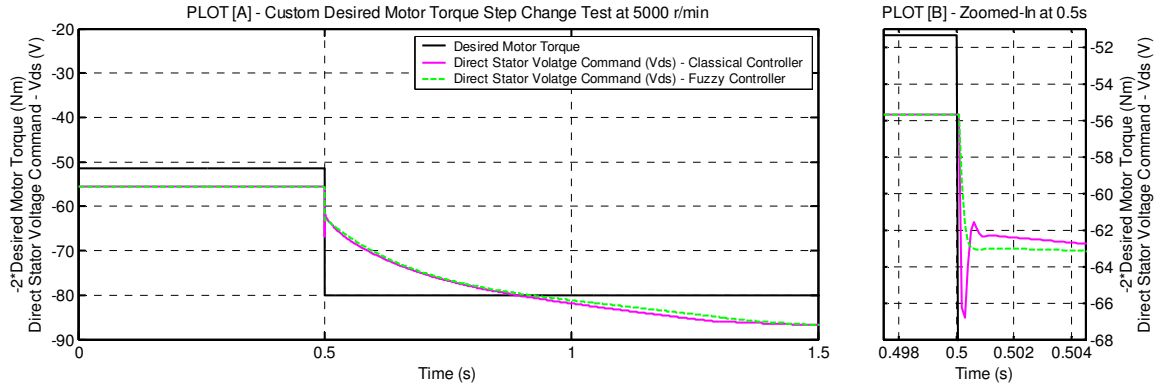


Figure 5.31. Stator direct voltage command, $\overline{V}_{ds}$, for a torque command at 5000 r/min

There were significant improvements for the fuzzy controller during the custom step torque command at medium motor speed of 5000 r/min over the classical controller. Even greater improvements are therefore expected for the step torque command test at 8000 r/min.

Figure 5.32 shows the results from such a step torque command test at 8000 r/min. Plot [A] of Figure 5.32 shows that the desired motor torque is stepped from 48 Nm to 55 Nm. This resulted in the motor accelerating from 8000 r/min to 9500 r/min in

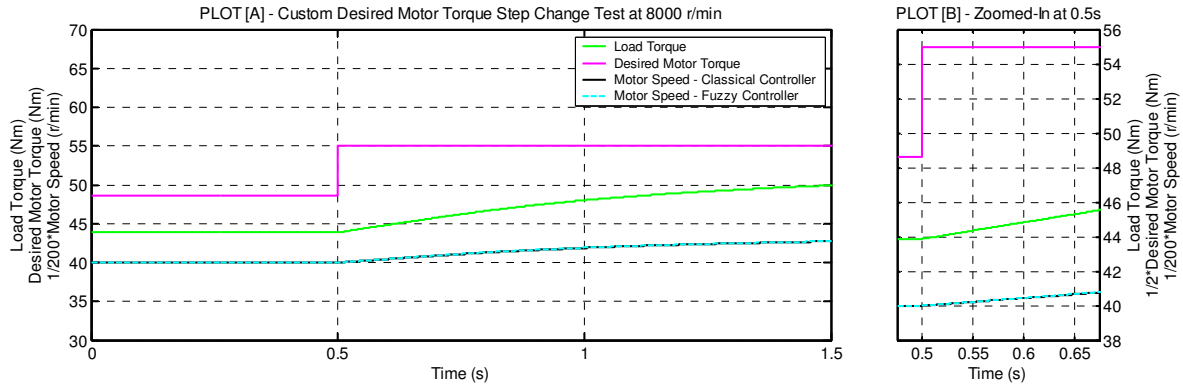


Figure 5.32. Response of desire torque, $\overline{T}_m$, and motor speed, ω_m, for a torque command at 8000 r/min

about 1 s. During this period, the load torque changed from about 44 Nm to about 50 Nm. Plot [B] of Figure 5.32 shows the zoomed-in portion of the test around 0.5 s. Again, the motor speed behavior is very similar for both the classical and the fuzzy controller and this enables an evaluation of the performance of key elements of the classical and the fuzzy controller.

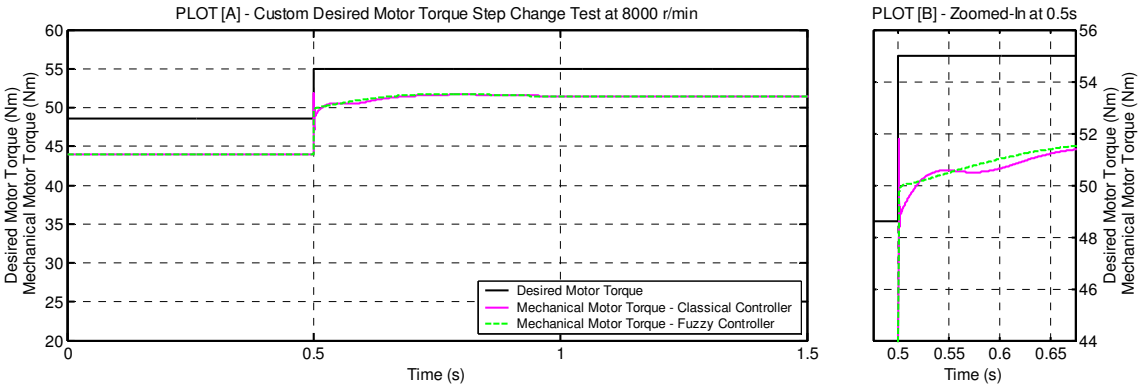


Figure 5.33. Response of motor torque, T_m , for a torque command at 8000 r/min

Figure 5.33 shows the simulation results of actual motor mechanical torque using the classical and the fuzzy controller. It is clear from plot [A] of Figure 5.33 that at the

high-speed of 8000 r/min, there is significant deterioration in the actual torque response for the classical controller. The zoomed-in plot [B] of Figure 5.33 shows that the actual motor mechanical torque response for the classical controller contains high initial torque spike of about 3 Nm or 6% followed by torque undershoot of about 2 Nm or 4%. The classical controller then slowly corrects the torque response with the settling time at around 0.3 s. In high performance applications, such behavior of the motor torque is not desired. With the fuzzy controller, there were no overshoots or undershoots in the torque response. Furthermore the torque response was non-linear in nature. The motor torque rose quickly in almost a straight-line manner around the target set point, and then changed its direction and settled to the desired set point in a short period of time. The torque response of the induction machine for our fuzzy controller was exceptionally better than that of the classical controller.

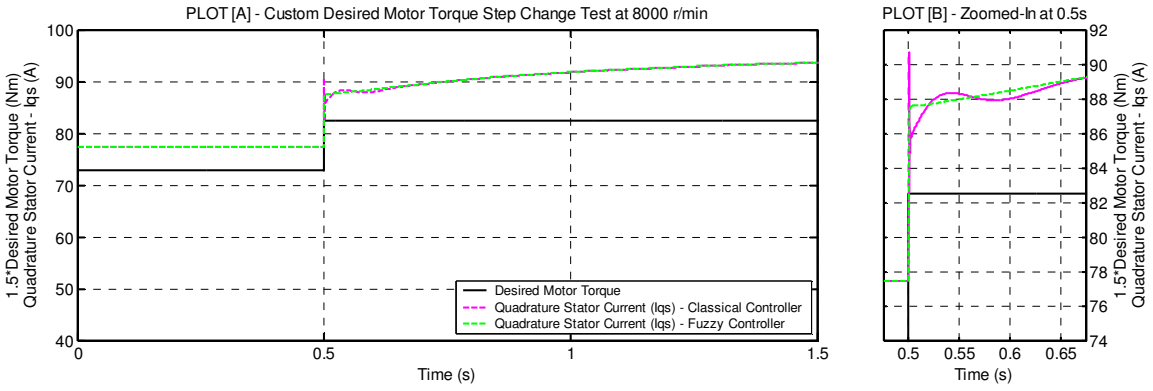


Figure 5.34. Response of stator quadrature-axis current, I_{qs}, for a torque command at 8000 r/min

Figure 5.34 shows the behavior of stator quadrature-axis current, I_{qs}, for the step torque command at 8000 r/min. Plot [A] of Figure 5.34 clearly shows an exceptional I_{qs} current response for the fuzzy controller since it does not contain any overshoots or

undershoots. The zoomed-in plot [B] of Figure 5.34 shows that the response of the I_{qs} current for the fuzzy controller is significantly better than that of classical controller. Again, we see that stator component current, I_{qs}, rose quickly in almost a straight-line manner around the target set point, then changed its direction and settled to the desired set point in less than 0.02 s. Again, the I_{qs} current response for the classical controller had an initial spike of about 3 amps or 3.4% followed by undershoot of about 2 amps or 2.3%. The controller then slowly corrects the I_{qs} current response with a relatively large settling time of around 0.3 s. It can be clearly seen that the I_{qs} current response for the fuzzy controller is exceptionally better than that of the classical controller. It is also seen that during this test, the stator component current, I_{qs}, continued to increase with motor speed. This is because of the fact that the motor is now completely operating in the field-weakening region and thus the stator component current, I_{qs}, is increased to meet the step torque command.

Figure 5.35 shows the behavior of stator direct-axis current, I_{ds}, for the step torque command at 8000 r/min. As can be from the plot [A] of Figure 5.35, there is an initial

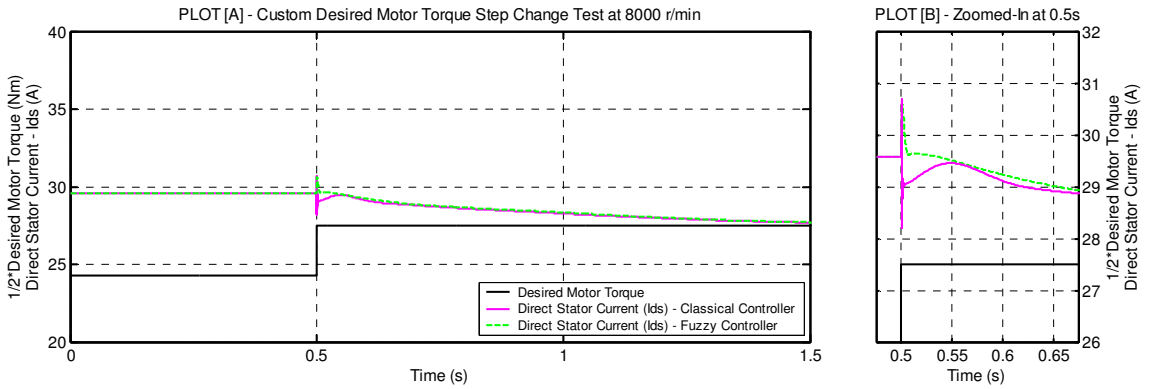


Figure 5.35. Response of stator direct-axis current, I_{ds}, for a torque command at 8000 r/min

spike in the actual value of the stator current, I_{ds}, with both classical and fuzzy controllers caused by a step increase in the I_{qs} current. The zoomed-in plot [B] of Figure 5.35 clearly shows that the stator direct-axis current, I_{ds}, for the fuzzy controller was corrected instantly with a small settling time of less than 0.02 s. Similar to the results shown for step torque command at 5000 r/min, the I_{ds} current for the classical controller had an undershoot of about 1.5 amps or 5% after the initial spike, which was slowly corrected at a settling time of greater than 0.4 s. It can be clearly seen from Figure 5.35, that I_{ds} current response for the fuzzy controller is exceptionally better than that of the classical controller. It is also noticed that the I_{ds} current continued to decrease with the increase in the motor speed for both the fuzzy and classical controllers. This is expected since the motor is operating in the field-weakening region and the stator component current, I_{ds}, is reduced accordingly. Figures 5.36 and 5.37 show the comparison of stator component voltage commands, $\overline{V}_{ds}$ and $\overline{V}_{qs}$, for the fuzzy controller and the classical controller. Again, it is seen that the stator component voltage commands, $\overline{V}_{ds}$ and

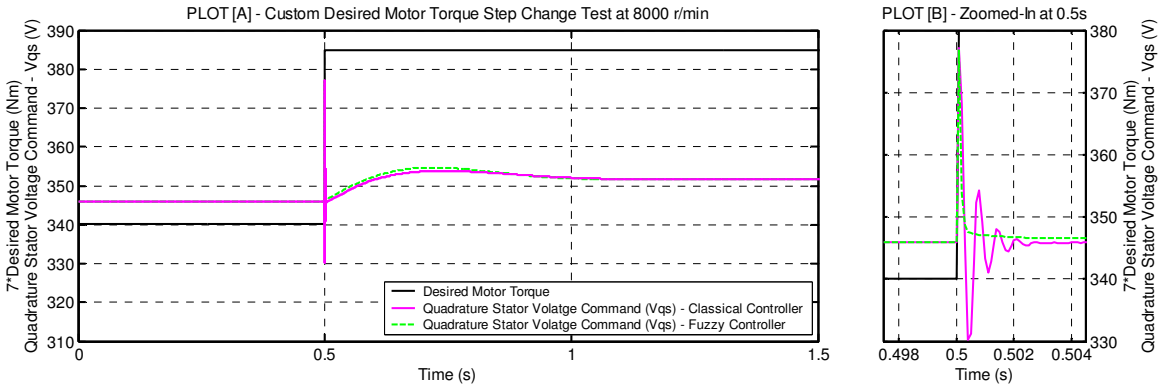


Figure 5.36. Stator quadrature voltage command, $\overline{V}_{qs}$, for a torque command at 8000 r/min

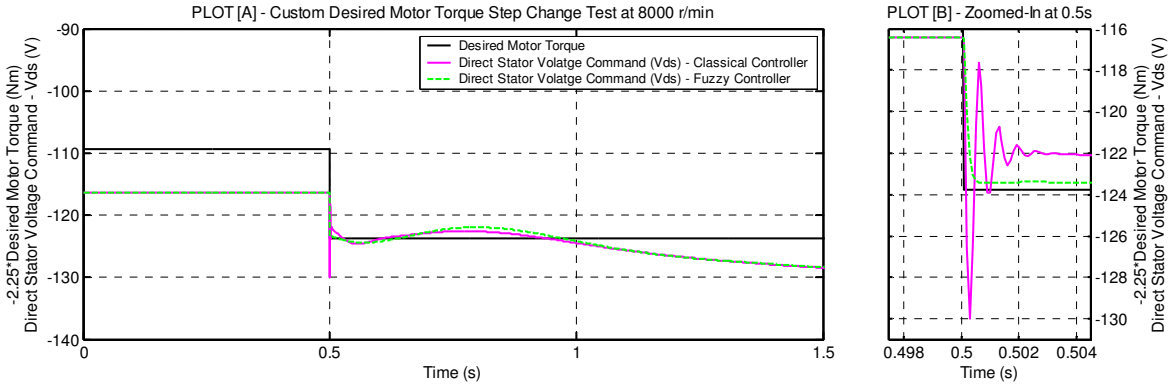


Figure 5.37. Stator direct voltage command, $\overline{V}_{ds}$, for a torque command at 8000 r/min.

$\overline{V}_{qs}$, for the fuzzy controller does not contain any overshoots or undershoots. Whereas, for the classical control, the voltage components, $\overline{V}_{ds}$ and $\overline{V}_{qs}$ contain numerous overshoots as well as undershoots which clearly indicates a very poor controller response.

From the custom torque command tests conducted at 2000 r/min, 5000 r/min and 8000 r/min and with a 30% parameter mismatch, it is clearly shown that as the actual motor speed increased, the performance of fuzzy controller remained robust, while that of the classical controller deteriorated. The errors in the classical controller are caused by de-coupling compensation terms, which are used to calculate the stator components voltages, $\overline{V}_{ds}$ and $\overline{V}_{qs}$. Clearly, the mismatch error increases as $\overline{\omega}$ increases, and this, together with the use of linear PI controllers, is responsible for the poor performance of the classical controller at high-speed. This poor control is clearly evident at high-speed and can be seen in the stator voltage component commands, $\overline{V}_{ds}$ and $\overline{V}_{qs}$, of Figure 5.36 and Figure 5.37 for the classical controller. Also note that under parameter mismatches, the performance of the classical controller deteriorated even further when operating in

torque control mode compared to speed control mode. Whereas, it is shown that the proposed controller maintained a robust control of the induction machine in torque control, as well as, speed control mode. It can be clearly seen from the results of our simulations, that the proposed fuzzy controller, due to its non-linear behavior, can robustly control the induction machine torque, and stator component currents, I_{ds} and I_{qs}, independent of the speed, and even with parameter mismatches.

6. CONCLUSION AND FURTHER RESEARCH

The development of high performance drives in the recent years has led to the use of AC machines in applications that were considered to be the sole domain of the DC machines. One of the advantages that AC induction machines offer over the DC machines, and AC permanent magnet machines, is that the induction machine due to its robust rotor construction can be operated at much higher rotational speeds. Well known high-speed applications include: electric vehicle drives systems, high-speed spindle motors, and dynamometer test stands. In these systems, the speed of the induction machines can be in excess of 10,000 r/min.

For high performance applications, classical *d-q* controllers are mostly used for controlling the induction machines. The approach of the classical controller is based on an accurate mathematical model of the induction machine at steady-state. However, in a real world environment, there are always errors between the estimated induction machine parameters and the actual parameters. Under such conditions, use of the classical controller can cause significant overshoots and undershoots in the torque and current response of the induction machine, resulting in deteriorated performance of the controller.

In this research, a novel approach using a fuzzy logic-based *d-q* control to improve the performance of the controller was presented. This fuzzy controller, due to its non-linear behavior, is capable of significantly improving the performance of the induction machine systems.

The fuzzy controller's performance was validated through simulations. The fuzzy logic-based controller's behaviors in simulation environment using custom tests were compared to that of the classical controller. The results clearly demonstrated the superior performance of the fuzzy controller versus the classical controller. From the results, it was evident that due to the uncertainty of the environmental conditions and non-linearity of induction machine dynamics, a classical controller due to its linear approach will not always be able to give a satisfactory response. These results clearly show that the fuzzy controller can significantly eliminate the overshoots and undershoots and improve the settling times by a significant margin in comparison to the classical controller. Thus, during conditions such as the ones presented which could result in overshoots, undershoots, and large settling times, the fuzzy controller provides a huge opportunity to further improve induction machine performance.

The fuzzy logic-based approach used fuzzy rules for improving the dynamic torque and current response of the induction machine. These rules are meant to cover all possible conditions which the specific control system may encounter. However, the rules base was composed of two distinct sets of rules, dealing with either direct-axis or quadrature-axis components of the current that are mutually exclusive. Hence, there is a possibility of further enhancing the rules by making them inclusive. At the same time, making the rules inclusive can result in increased complexity of this proposed fuzzy controller. Further research in this field needs to be performed to study the benefits of making these rules inclusive, and the impact on controller performance improvement versus controller complexity.

7. REFERENCES

[1] Andrzej M. Trzynadlowski, "Control of induction motors," Academic Press, San Diego, 2001.

[2] Hajku Lee, Jaedo Lee, Sejin, Seong, "Approach to fuzzy control of an indirect field-oriented induction motor drives," in proc IEEE ISIE, 2001, pp. 1119-1123, 2001.

[3] Irving M. Gottlieb, "Electric motors and control techniques," Second Edition, McGraw-Hill, Inc., New York, 1994.

[4] Jinhwan Jung, Kwanghee Nam, "A dynamic decoupling control scheme for high-speed operation of induction motors," IEEE Transactions on Industrial Electronics, vol. 46, issue 1, pp.100-110, Feb. 1999.

[5] J. Jung, S. Lim, K. Nam, "A PI type decoupling control scheme for high speed operation of induction motors," in Proc. IEEE PESC'97, pp. 1082-1085, 1997.

[6] A. E. Fitzgerald, Charles. Kingsley, Jr., Stephen D. Umans, "Electric machinery," Fifth Edition, McGraw-Hill, Inc., New York, 1990.

[7] Eun-Chul Shin, Tae-Sik Park, Won-Hyun Oh, Ji-Yoon Yoo, "A design method of PI controller for an induction motor with parameter variation," in Proc. IEEE PESC'2003, pp. 408-413.

[8] M. Zerikat, M. Bendjebbar, N. Benouzza, "Dynamic Fuzzy-neural network controller for induction motor drive," Proceedings of World Academy of Science, Engineering and Technology, vol. 10, pp. 278-283, 2005.

[9] M.J. Hossain, M.A. Hoque, M.A. Ali, M.A. Rahman, " Fuzzy-logic-based control for induction motor drive with the consideration of core loss," International Conference on Electrical and Computer Engineering 2006, pp. 333-336, Dec. 2006.

[10] T. S. Radwan, "Perfect Speed tracking of direct torque controlled-induction motor drive using Fuzzy logic," in proc. IEEE PEDS 2005, pp. 38-43, 2005.

[11] Yan Liu, Cheng Shao, "A torque control scheme of induction motor in hybrid electric vehicle," SICE-ICASE International Joint Conference 2006, pp. 540-544, Oct. 2006.

[12] L. A. Zadeh, "Fuzzy logic and the calculus of fuzzy if-then rules," in Proc. 22nd Intl. Symp. on Multiple-Valued Logic, Los Alamitos, CA: IEEE Computer Society Press, 1992, pp. 480-480.

[13] Jae-Sub Ko, Jung-Sik Choi, Dong-Hwa Chung, "Hybrid artificial intelligent control for speed control of induction motor," SICE-ICASE International Joint Conference 2006, pp. 678-683, Oct. 2006.

[14] Sanjiv Kumar, Bhim Singh, J. K. Chatterjee, "Hybrid speed controller for vector controlled cage induction motor drive," in proc. IEEE PESC'98, pp. 147-152, 1998.

[15] Mouloud Azzedine Denai, Sid Ahmed Attia, "Fuzzy and neural control of an induction motor," International Journal of Applied Math. And Comp. Science, vol.12, No.2, 2002.

[16] Abdelnassir Abdalla, "Torque ripple minimization in direct torque control of induction machines," University of Akron, May 2005.

[17] Jussi Puranen, “Induction motor versus permanent magnet synchronous motor in motion control applications: a comparative study,” Lappeenranta University of Technology, December 2006.

[18] Muhammad H. Rashid, “Power electronics circuits, devices, and applications,” Second Edition, Prentice Hall, New Jersey, 1993.

[19] P.C. Krause, “Analysis of electric machinery,” Mcgraw-Hill Book Company, 1986.

[20] Sompong Srilad, Satean Tunyasrirut, Tianchai Suksri, “ Implementation of a scalar controlled induction motor drives,” SICE-ICASE International Joint Conference 2006, pp. 3605-3610, Oct. 2006.

[21] Ramshaw, van Heeswijk, “Energy conversion electric motors and generators,” Sanders College Publishing, Philadelphia, 1990.

[22] Hao Ying, “Fuzzy Control and Modeling,” IEEE Press: New York, 2000.