COMPARATIVE PERFORMANCE ANALYSIS OF DRIVES FOR INDUCTION MOTORS

BY

ANDREW PHILIP FRIEDL

THESIS

Submitted in partial fulfillment of the requirements for the degree of Master of Science in Electrical and Computer Engineering in the Graduate College of the University of Illinois at Urbana-Champaign, 2009

Urbana, Illinois

Adviser:

Professor Philip T. Krein
ABSTRACT

The asynchronous polyphase induction motor has been the motor of choice in industrial settings for about the past half century because power electronics can be used to control its output behavior. Before that, the dc motor was widely used because of its easy speed and torque controllability. The two main reasons why this might be are its ruggedness and low cost. The induction motor is a rugged machine because it is brushless and has fewer internal parts that need maintenance or replacement. This makes it low cost in comparison to other motors, such as the dc motor. Because of these facts, the induction motor and drive system have been gaining market share in industry and even in alternative applications such as hybrid electric vehicles and electric vehicles.

The subject of this thesis is to ascertain various control algorithms’ advantages and disadvantages and give recommendations for their use under certain conditions and in distinct applications. Four drives will be compared as fairly as possible by comparing their parameter sensitivities, dynamic responses, and steady-state errors. Different switching techniques are used to show that the motor drive is separate from the switching scheme; changing the switching scheme produces entirely different responses for each motor drive.
ACKNOWLEDGMENTS

I have been incredibly fortunate to have Professor Philip T. Krein’s guidance for the past few years in my graduate studies. Without his support and assistance, this work would not have been possible. I thank him, the Grainger Foundation, and the National Science Foundation for their essential roles in directing me and shaping my future as an engineer.

I would also like to thank all of the people who have helped me through the good times and the bad, such as Robert Eagle, who has recently passed away, my undergraduate advisor, Prof. Gary Eden, and all of the amazing students here at the University of Illinois at Urbana-Champaign who put in long hours alongside of me. I would like to particularly thank numerous undergrad research assistants who all helped in some way or another with the results in this thesis.

Finally, everything I have done and accomplished so far in my career would have been impossible without the guidance of my loving parents, brother and extended family. I thank them for all the sacrifices they have made to help me become the person I am today.
# TABLE OF CONTENTS

1. INTRODUCTION .......................................................................................................... 1  
   1.1 Project Motivation .................................................................................................... 1  
   1.2 Objectives ................................................................................................................. 2  
   1.3 Software and Hardware Used ................................................................................... 3  

2. LITERATURE REVIEW AND BACKGROUND ........................................................ 5  
   2.1 Induction Motor Dynamic Model ............................................................................. 9  
   2.2 Induction Motor Parameterization .......................................................................... 10  
   2.3 Control Methods ..................................................................................................... 13  
   2.4 General Literature Review ...................................................................................... 14  
   2.5 Direct Torque Control ............................................................................................. 20  
   2.6 Indirect Field-Oriented Control .............................................................................. 20  
   2.7 Feedback Linearization: Input-Output Decoupling Control ................................... 23  
   2.8 Vectorized Volts-per-Hertz ..................................................................................... 25  
   2.9 Motor Control Continuum ...................................................................................... 26  

3. ANALYTICAL DEVELOPMENT .............................................................................. 28  
   3.1 Theoretical Basis for DTC ...................................................................................... 28  
      3.1.1 Singular perturbation theory and sliding-mode control .................................. 28  
      3.1.2 Direct torque control stability analysis ........................................................... 33  
      3.1.3 Direct torque control parameter sensitivity analysis .................................... 37  
   3.2 Indirect Field-Oriented Control .............................................................................. 38  
      3.2.1 Indirect field-oriented control stability analysis ............................................. 38  
      3.2.2 Indirect field-oriented control parameter sensitivity ................................... 40  
   3.3 Feedback Linearization: Input-Output Decoupling Control ................................... 41  
      3.3.1 Feedback linearization: input-output decoupling control stability analysis ... 41  
      3.3.2 Feedback linearization: input-output decoupling control parameter sensitivity analysis ................................................................................................... 42  
   3.4 Vectorized Volts-Per-Hertz .................................................................................... 43  
      3.4.1 Vectorized volts-per-hertz control parameter sensitivity analysis .................. 43  
   3.5 Load Modeling ........................................................................................................ 44  
   3.6 Flux Observer Design ............................................................................................. 45  

4. COMPUTER SIMULATION ....................................................................................... 47  
   4.1 Simulation Modeling .............................................................................................. 47  
   4.2 Simulation of Direct Torque Control ...................................................................... 49  
      4.2.1 Direct torque control parameter sensitivity results ........................................ 51  
      4.2.2 DTC performance with changing sector count ............................................. 54  
      4.2.3 DTC performance analysis .......................................................................... 59  
   4.3 Simulation of Indirect Field-Oriented Control ....................................................... 63  
   4.4 Simulation of Vectorized Volts-per-Hertz .............................................................. 70  
   4.5 Simulation of Feedback Linearization .................................................................... 74  
   4.6 Comparison of Motor Drives .................................................................................. 81  

iv
5. ALGORITHM IMPLEMENTATION ................................................................. 91
   5.1 Modular Inverter .................................................................................. 91
       5.1.1 Front end ...................................................................................... 92
       5.1.2 Power stage .................................................................................. 93
       5.1.3 Control stage ............................................................................... 95
   5.2 Dynamometer ...................................................................................... 97
   5.3 Induction Motor .................................................................................. 98
   5.4 Role of the Computer ......................................................................... 99
   5.5 Hardware Results .............................................................................. 99

6. CONCLUSIONS AND FUTURE WORK ...................................................... 103

REFERENCES .............................................................................................. 105

APPENDIX A: ADDITIONAL DATA .............................................................. 108

APPENDIX B: SIMULATION MODELS, CONSTANTS, AND VARIABLES ...... 110
1. INTRODUCTION

1.1 Project Motivation

The need for comparing polyphase induction motor (IM) control methods has been around for quite some time. Some methods, such as field-oriented control (FOC), were introduced in the late 1970s, while others, such as direct torque control (DTC), were developed in the mid-1980s. A flurry of various controllers was introduced in the late 1980s through today. This was due to the fact that variable frequency drives have become extremely popular since the advent of power electronics in the 1960s. The use of the bipolar junction transistor (BJT) and then the field effect transistor (FET) have enabled the conversion between different types of power sources. The use of power electronics has enabled the control of a motor’s frequency, which has eliminated the need to start and run motors from line frequency. Because of the fact that there are so many different types and variations of controllers, there is a need to separate groups of controllers into application specific groups; a few of these groups include: hybrid-electric vehicles (HEVs) or electric vehicles (EVs); motor drives for industrial processes; general industrial uses such as pumps, compressors, and air conditioning units; and finally household appliances such as washing machines. The bulk of this work will attempt to survey the existing literature in addition to compiling relevant results and then compare and contrast four different motor controllers. A minor attempt will be made in this paper to group control algorithms with their respective applications and to give them a particular performance status, such as “high-performance” or “low-performance” by use of various operational metrics: one example is a parameter sensitivity analysis of each algorithm. DTC and IFOC will be studied in depth.
1.2 Objectives

Four types of IM control will be analyzed dynamically and in steady state, simulated, and implemented. Recommendations will be given for their use in different applications on the basis of their controller’s mathematical underpinnings and findings. The four control algorithms to be analyzed are DTC, IFOC, feedback linearization, and vectorized volts-per-hertz. The simulation for each type of control algorithm will be conducted using Matlab-Simulink. The hardware that will be used to implement each control algorithm is an in-house research-grade inverter designed at UIUC in the mid-2000s. It allows for a wide range of control schemes which all require different various feedback variables.

While the inverter is at the heart of the control schemes, the most important part of the system is the electromagnetic actuator, the induction motor. Magnetic induction occurs when current flowing on the stator coil of the motor induces a current on the rotor windings. Essentially, an induction motor behaves like a moving transformer. Through the use of an inverter and induction motor, four different types of motor controllers will be compared via simulation.

This work is broken down into six different chapters: Introduction, Literature Review and Background, Analytical Development, Computer Simulation, Algorithm Implementation, and finally Conclusions and Future Work. The Introduction gives a brief overview of what this document will contain and an overview of the project. The Literature Review and Background chapter contains a thorough literature review of the subject matter and necessary background for the rest of the thesis. The literature review is extensive because of the sheer number of publications that are similar to what this
research entails. The Analytical Development chapter contains the bulk of the theory for the composition. The chapter that contains the simulations comparing the motor drives is aptly named Computer Simulation. The Algorithm Implementation chapter gives an in-depth description of the hardware and software used for the physical realization of the theory. Because of time constraints, the simulation results could not be fully verified by fabricated hardware. A potential driving cycle of an EV is demonstrated with IFOC hardware results and matching simulation results. The IFOC hardware results are given in order to add credibility to the other motor drive simulations covered in this work that have not been fully verified by hardware. Finally, Conclusions and Future Work contain concluding remarks and possible future work.

1.3 Software and Hardware Used

As stated above, the main software program that will be used is Mathwork’s Matlab, version 2008a for the simulations, and 7.0.4 for the inverter platform. Version 7.0.4 was used for the inverter because it is the version that was used when the inverter was first created. It works, so as the saying goes, “don’t fix the wheel that isn’t broken.”

As for the inverter, it has three main stages: a Front End, a Power Stage, and a Control Stage. The Front End is, in its most basic form, a 3φ ac line rectifier and then a boost converter. The Power Stage is where the dc input from the Front End comes in and external input signals are used to drive power transistors. The power transistors are then controlled to create a 3φ dc-ac converter, also known as a “power inverter.” The final stage is the Control Stage, where the algorithm for the switching scheme and motor drive
lies. Because of the fact that it uses modules for each of the stages, it is also called a “modular inverter.” More will be said about the modular inverter in Chapter 5.

Other hardware used included a Magtrol Hysteresis Brake Dynamometer (HD) model HD-715-7N, Magtrol dynamometer controller model DSP6001, and a 2048 line resolution encoder that allows directional sensing and indexing. As for the induction motor, a 1.5 hp Dayton induction motor was chosen to be the representative electromagnetic actuator, but some simulations were done with a slightly larger 3 hp motor in an attempt to be more in the center of the 3φ IM range.
2. LITERATURE REVIEW AND BACKGROUND

Nikola Tesla invented the asynchronous polyphase induction motor in 1882 and patented it in 1888. This invention led to an increase in the mechanization rate of industry. The electric machine that we know today as the squirrel-cage induction motor was invented by Mikhail Dolivo-Dobrovolsky in 1891. Because of these two inventors along with the utility of power electronics, today we study the control of induction motors using advanced control techniques.

After considering the history of the induction motor, it is useful to give background and a basis to the following discussion. As stated earlier, the induction motor can be considered, in an electrical sense, to be a $3\phi$ transformer with an air gap and a winding that is in motion, called the rotor. Assuming a balanced $3\phi$ voltage set is applied to the stator, a few concepts can be obtained, namely that a magnetic field is found to be moving at synchronous speed in the motor air gap, and that this in turn will induce a current on the shorted rotor winding that has an angular electrical frequency associated with how much “slip” is occurring on the rotor. This induced current creates its own magnetic field which interacts with the stator magnetic field to create a force that repels the stator field. Under normal operating conditions, every time the stator field attempts to line itself up with the rotor, the rotor field restores the repulsion force. It is important to note that both fields rotate at synchronous speed around the axis of the motor. It is now necessary to define the synchronous speed as the “angular speed that the rotor spins multiplied by the number of poles divided by two,” as shown in (1) from [1].

$$\omega_e = 2\pi f_e = \frac{P}{2} \omega_{\text{rm(at synchronous)}}$$  \hspace{1cm} (1)
Slip can be better interpreted as the relative difference in motion between the stator and rotor. Slip is essentially caused by any torque against the motor on the shaft which slows the rotor down below synchronous speed. In normal motoring operation, it consists of any power that is lost or consumed via 3 main routes: frictional, windage, and torque load. Motor slip is defined as “the ratio of the difference between the synchronous angular electrical frequency and rotor angular electrical frequency to the synchronous angular electrical frequency,” as seen in (2).

\[ s = \frac{\omega_e - \omega_r}{\omega_e} \]  

(2)

By this definition, slip is 0 at synchronous speed, and 1 when not moving, also known as the “blocked rotor condition.” The motor starts out in the blocked rotor condition with the induced rotor currents equal to the synchronous angular electrical frequency. As the motor speeds up, the rotor frequency decreases until it reaches steady state—somewhere below synchronous—speed depending on the slip. It never reaches synchronous speed because of the losses inherent in the rotor’s movement as stated previously. From [1], the slip frequency is now defined to be the “difference between the synchronous electrical angular frequency and the rotor electrical angular frequency” (3).

\[ \omega_{slip} = \omega_e - \omega_r = s\omega_e \]  

(3)

The next logical step in describing the motor is by the torque it can produce. Before going over the torque and its constituent equations, reference frame theory must be reviewed. In reference frame theory, a change of coordinates occurs whereby the time-varying variables such as voltage, current, and flux linkage get translated into a rotating reference frame, called the “arbitrary reference frame.” This is commonly known as Park’s transform, after R. H. Park who formulated the transformation
specifically for electric machines in the late 1920s. The rotation speed of the reference frame is, as its name implies, arbitrary, where the common speeds are \( \theta, \omega_s, \omega_r, \) and \( \omega \) which correspond to the stationary, synchronous, rotor and arbitrary reference frames, respectively. The general form for the change of variables transformations is given in (4), where \( f_{qd0s} \) is the variable expressed in the qd0 arbitrary coordinate frame and \( f_{abcs} \) is the time varying stator variable being transformed [1].

\[
f_{qd0s} = K_s f_{abcs}
\]

where

\[
K_s = \frac{2}{3} \begin{pmatrix}
\cos(\theta) & \cos\left(\theta - \frac{2\pi}{3}\right) & \cos\left(\theta + \frac{2\pi}{3}\right) \\
\sin(\theta) & \sin\left(\theta - \frac{2\pi}{3}\right) & \sin\left(\theta + \frac{2\pi}{3}\right) \\
\frac{1}{2} & \frac{1}{2} & \frac{1}{2}
\end{pmatrix}
\]

and

\[
\omega = \frac{d\theta}{dt} \text{ or } \theta = \int_0^t \omega(\xi) \, d\xi + \theta(0)
\]

It should be noted that the subscripts \( q, d, \) and \( \theta \) refer to the quadrature, direct and zero components, of any reference frame, respectively, but are typically taken to be in the arbitrary reference frame. It is also necessary to convert the qd0 variable into continuous time variables. This can be accomplished by using the following inverse transform (5), also known as the inverse Park transform [1]:

\[ f_{abcd} = (K_s)^{-1} f_{qd0s} \]

where
\[
(K_s)^{-1} = \begin{pmatrix}
\cos(\theta) & \sin(\theta) & 1 \\
\cos\left(\theta - \frac{2\pi}{3}\right) & \sin\left(\theta - \frac{2\pi}{3}\right) & 1 \\
\cos\left(\theta + \frac{2\pi}{3}\right) & \sin\left(\theta + \frac{2\pi}{3}\right) & 1
\end{pmatrix}
\]

Now that reference frame theory has been reviewed, the rationalization of torque can be better illustrated. Co-energy is used instead of actual energy in the analysis of the energy stored in the coupling field. The coupling field can be obtained, as in (7), by the integration of the differential equation in (6). In Equation (7), \( i_{abcd} \) is the stator line current vector, \( L_s \) is the stator inductance matrix, \( L_{ls} \) is the stator leakage inductance term, \( L_{sr} \) is the referred mutual inductance. In the special case where the electric machine is assumed to be magnetically linear, the energy stored in the coupling field, \( W_f \), is equal to the negative of the co-energy, \( W_c \) [1].

\[
\frac{dW_m}{dt} = i \frac{d\lambda}{dt} - f \frac{dx}{dt}
\]

\[
W_f = \frac{1}{2}(i_{abcd})^T(L_s - L_{ls} I)i_{abcd} + (i_{abcd})^T L_M i_{abcd} + \frac{1}{2}(i_{abcd})^T(L_r - L_{sr} I)i_{abcd}
\]

\[
W_f = -W_c
\]

With the previous equations, the stage has been set for the following torque expressions. The basic torque equation is given by (9) from [1] which states that “the partial of the co-energy with respect to the rotor position is the torque of electrical origin.” The co-energy from (9) is then broken down in (10). From (10), it is broken
down further still in (11) into its scalar components in the dq0 plane, where the 0 axis on the dq0 plane has been neglected due to the fact that a balanced 3φ load and source are assumed. Two other equivalent expressions for the electrical torque are found in (12) corresponding to whether the calculation is done from the stator or rotor perspective. For the purposes of this paper, the last expression in (2.12) will be used.

\[
T_e (i, \theta_r) = \frac{\partial W_e (i, \theta_m)}{\partial \theta_m} = \left(\frac{P}{2}\right) \frac{\partial W_e (i, \theta_r)}{\partial \theta_r} \tag{9}
\]

\[
T_e = \frac{P}{2} (i_{abc})^T \frac{\partial}{\partial \theta_r} \left[ L_{mr} \right] i_{abc} \tag{10}
\]

\[
T_e = \frac{3}{2} \frac{P}{2} L_M \left( i_{qr} i_{dr} - i_{qs} i_{qs} \right) \tag{11}
\]

\[
T_e = \frac{3}{2} \frac{P}{2} \left( \lambda_{qr} i_{dr} - \lambda_{qs} i_{qs} \right) \tag{12}
\]

and

\[
T_e = \frac{3}{2} \frac{P}{2} \left( \lambda_{ds} i_{qs} - \lambda_{qs} i_{ds} \right)
\]

2.1 Induction Motor Dynamic Model

The following set of seven time-varying equations in (13) from [1] are known collectively as the induction motor dynamic model in the arbitrary reference frame. The variables with the subscript \( s \) indicate that it is a stator variable, while the subscript \( r \) shows it is a rotor quantity. It should be noted that this model is exclusively for variables in the dq0 plane rather than the time-variant plane. The motor terminal voltages are indicated by \( v \), the currents by \( i \), the flux linkages by \( \lambda \), and the rotor speed by \( \omega_r \). The number of pole pairs is given by \( n_p \), the load torque by \( T_{load} \), and the shaft moment of inertia by \( J \). These equations will be used in the simulations that follow and are also inherent in governing the induction motor dynamic behavior in the implementation.
2.2 Induction Motor Parameterization

Parameters in the 3φ induction motor per-phase circuit shown in Figure 1 can be found using 4 main tests: blocked rotor, no load, dc, and inertia tests. The first 3 of the tests, the blocked rotor, no load and dc tests, are used to parameterize the equivalent lumped circuit elements, such as $\lambda_{qs}$, $\lambda_{ds}$, $r_s$, $L_s$, $L_r$, $M$, $R_L$, and $R$. The last test, the inertia test, finds the inertial constant, $J$, for the shaft. The typical procedure for electrical testing is as follows: perform the no load test with the stator coil impedance assumed to be zero, then the blocked rotor test assuming the core impedance assumed to be negligible, and finally the dc test. This process is then iterated not assuming that the core or the stator coil impedances are negligible. The values found after this second iteration are usually very good approximates. If more precision is necessary, a third iteration is performed with the new second iteration circuit values.

\[
\begin{align*}
\frac{d\lambda_{qs}}{dt} & = -r_s i_{qs} - \omega \lambda_{ds} + v_{qs} \\
\frac{d\lambda_{ds}}{dt} & = -r_s i_{ds} + \omega \lambda_{qs} + v_{ds} \\
\frac{d\lambda_{0s}}{dt} & = -r_s i_{0s} + v_{0s} \\
\frac{d\lambda_{qr}}{dt} & = -r_s i_{qr} - (\omega - n_p \omega_r) \lambda_{dr} + v_{qr} \\
\frac{d\lambda_{dr}}{dt} & = -r_s i_{dr} + (\omega - n_p \omega_r) \lambda_{qr} + v_{dr} \\
\frac{d\lambda_{0r}}{dt} & = -r_s i_{0r} + v_{0r} \\
\frac{d\omega_r}{dt} & = \frac{3 n_p}{2 J} (\lambda_{ds} i_{qs} - \lambda_{qr} i_{ds}) - \frac{T_{load}}{J}
\end{align*}
\]
Figure 1. The 3φ Induction Motor Electrical Equivalent Per-Phase Circuit

Being more specific, the no-load test is carried out first on the motor by applying rated voltage across all three windings after bringing the motor up to synchronous speed—usually by the use of an attached dynamometer. The total 3φ power is obtained and the line voltage and currents are measured. The per-phase power is then used to create an estimate of \( R_c \) by Equation (14). The per phase reactive power, \( X \), is used to estimate the magnetizing reactance, \( X_M \), and thus the magnetizing inductance, \( L_M \), as shown in (15) below.

\[
R_c = \frac{V_\phi^2}{P_\phi} \tag{14}
\]

\[
X_M = j\omega_s L_M = \frac{V_\phi^2}{Q_\phi}
\]

where \( Q_\phi = \sqrt{S_\phi^2 - P_\phi^2} \) \tag{15}

The blocked rotor test consists of locking the rotor in place and energizing the motor with rated current. By doing this, and assuming no core losses, it will give an equivalent \( R_{eq} \) and \( X_{eq} \) of the stator and rotor windings (16 and 17). The stator winding
resistance can be assumed to be half of the total equivalent resistance in order to get an initial approximation. The other half of the resistance is the referred rotor resistance. The split in stator and rotor reactances is a little more difficult, as it requires knowledge of the NEMA motor design class. Usually 60% of the equivalent reactance is accounted for in the rotor.

\[ R_{eq} = \frac{P_\phi}{I_\phi^2} = R_{as} + R_{bs} \]

where \( R_{as} \) is the a phase stator resistance and \( R_{bs} \) is the b phase stator resistance (16)

\[ X_{eq} = \frac{Q_\phi}{I_\phi^2} = X_{as} + X_{bs} \]

where \( X_{as} \) is the a phase stator reactance and \( X_{bs} \) is the b phase stator reactance (17)

Then the dc test is performed by energizing two of the three terminals with enough voltage to give rated current. Ohm’s law, Equation (18), is then invoked to find the resultant equivalent resistance, \( R_{eq} \), which is the summation of the two phase stator resistances. This is repeated for the other two combinations of windings. The system of three linear equations is then solved for, and the phase stator resistances found. These values are then used to give a much better approximation to the resistance values found in the blocked rotor test. As stated above, the three electrical tests are iterated as necessary.

\[ V_{DC} = I_{DC} R_{eq} = I_{DC} (R_{as} + R_{bs}) \]

(18)

The inertia test is performed independently of the electrical tests. The main point of this test is to determine the inertia, \( J \), of the motor’s shaft. A good approximation of
the inertia can be made by using a known mass, attaching it to the shaft, winding it up on
the shaft, and allowing it to drop from a certain height. The time it takes to fall can be
used in conjunction with the known mass in order to find the inertia by (19). By
Newton’s second law of motion, the force, \( F \), on the mass, \( m \), is known to be the product
of the mass times its acceleration, \( a \), in this case gravity, \( g \) (20). The product of the
radius of the shaft and the force gives torque. Torque is known, and so is the
acceleration, \( \frac{d\omega}{dt} \), thereby allowing the inertia, \( J \), to be found. The above is an
approximation in that it does not consider frictional losses of the motor, windage, or the
friction of the apparatus.

\[
\sum T = J \frac{d\omega}{dt} \tag{19}
\]

\[
F = ma = mg \tag{20}
\]

2.3 Control Methods

As mentioned earlier, the four control algorithms to be analyzed are DTC, IFOC,
feedback linearization, and vectorized volts-per-hertz. While there are many different
motor drives and also just as many, if not more, variations on these motor drives, these
four particular motor drives were chosen in order to get close to an assumed full
“spectrum” of performance, from no knowledge of motor parameters and open loop
control to precise parameter knowledge and closed-loop control. Vectorized volts-per-
hertz was chosen because it is the least complicated vector control, and is run as an
open-loop motor drive. DTC was chosen because of its ruggedness and simplicity of
design, while also maintaining closed-loop control. It also uses few parameters to
implement. IFOC was selected because it uses slightly more parameters than DTC, is

13
closed-loop, and tries to imitate the control action of a dc motor controller. The final type of motor drive that was selected was the feedback linearization, input-output decoupled control. This drive was picked because it uses a large number of motor parameters, is closed-loop, and is assumed to be one of the highest performing motor drives. Next, a literature review will be conducted to see what topics have already been researched. At first, the review will consist of mostly major multiple drive comparisons, and then a more detailed review of individual drives will be conducted. There will be an emphasis on IFOC vs. DTC comparisons, due to the sheer number of research topics comparing the two drives.

2.4 General Literature Review

It can be seen that there are no significant numerical comparisons of dynamic responses between IFOC and DTC available in the literature. Analyses of parameter sensitivities do not quantize the effect of parameter variations or errors on transient responses. Most of the literature deals with steady-state performance measures [2, 3, 4, 5], while [3, 4, 5, 6] provide some comparisons of dynamic responses. A detailed comparison of different induction motor drives is given in [2], including volts-per-hertz (V/Hz), FOC, DTC, direct self control (DSC), and DTC with space vector modulation (DTC-SVM). This comparison mentions advantages and disadvantages relative to steady-state measures, such as phase current peaks, current and torque harmonics, and switching frequency variation. Structural measures, such as the need for flux observers, and decoupling the torque and flux commands, are also presented. In [3], classical DTC and DTC-SVM but not IFOC are compared. The authors in [3] try to match the
switching scheme with the drive to have similar switching frequencies, but DTC is used with a switching table resulting in variable switching frequencies, and DTC-SVM is used with SVM resulting in a fixed switching frequency. The speed and torque dynamic responses, including settling time and overshoot, are compared. Tripathi et al. [7] propose a modified DTC which uses the stator flux to control the torque. No clear comparisons are made between DTC, DTC-SVM, and FOC, even though a vector diagram showing the dynamic operation of FOC and DTC is presented.

Cruz et al. [4] compare FOC, DTC and input/output linearization based on steady-state torque ripple, current peak, and switching frequency to name a few. They conclude that FOC and DTC are “good” in dynamic response, and that the parameter sensitivities are “low” and “medium” in DTC and FOC, respectively. Their conclusions can be seen in Table 1. It can be seen that no analysis has been done to give these vague performance descriptions numerical values. For instance, the dynamic responses for all of them are categorized as “good.” This is meaningless when trying to compare drives. The paper states that torque ripple is “high” for DTC, but “low” for FOC and input-output feedback linearization (IOL), which again shows the ambiguity of the results. Also, there is no need for FOC or IOL to have a fixed switching frequency, and DTC should be allowed to have a fixed switching frequency if desired. The point is that the motor drives should be completely separated from their switching actuators.
Table 1. Motor Drives Performance Comparison by Cruz et al. [4]

<table>
<thead>
<tr>
<th>Comparison criterion</th>
<th>Control Law</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>DTC</td>
</tr>
<tr>
<td>Dynamic response</td>
<td>GOOD</td>
</tr>
<tr>
<td>Torque ripple</td>
<td>HIGH</td>
</tr>
<tr>
<td>Average torque controller</td>
<td>NO</td>
</tr>
<tr>
<td>Current maintained within motor rating</td>
<td>NO</td>
</tr>
<tr>
<td>Complexity of architecture</td>
<td>LOW</td>
</tr>
<tr>
<td>Fixed switching frequency</td>
<td>NO</td>
</tr>
<tr>
<td>Parameter sensitivity</td>
<td>LOW</td>
</tr>
</tbody>
</table>

In [8], the authors compare DTC and IFOC directly using hysteresis control and a switching table for DTC and a current control using an anti-windup PI control. The authors compare speed, torque, flux, voltages and currents of the two drives and their corresponding switching techniques. They conclude that “… this simulation study reveals a slight advantage of [the] DTC scheme compared to [the] IFOC scheme regarding the dynamic flux control performance. The DTC might be preferred for high dynamic applications, but shows higher current and torque ripple” [pp. 4 of 8]. Again, these authors never numerically compare their results although simulation plots are given.

In [9] on pages 425-426, the authors compare rotor flux oriented control to direct torque control. They conclude that “… both techniques achieve a similar level of transient torque performance, however the DTC scheme is disadvantaged due to the possibility of loss of flux control at lower speeds/loads, higher torque/current ripple and the uncontrolled current transients. The vector control scheme however is disadvantaged by greater parameter sensitivity and the commissioning problems associated with setting up the current control loops. It can be concluded therefore that both techniques offer
similar levels of machine performance, each with particular advantages and disadvantages. The selection of the optimal drive controller will therefore depend on the particular application.” The authors do not say how to go about picking out what drives are suitable for various applications, and it is left up to the reader to decipher this. In Table 2, the performance summary is given for the two drive types, but again it is lacking in quantifiable results.

**Table 2. Motor Drives Performance Comparison by Telford et al. [9]**

<table>
<thead>
<tr>
<th></th>
<th>DTC</th>
<th>VC</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Dynamic response</strong></td>
<td>Excellent</td>
<td>Excellent</td>
</tr>
<tr>
<td><strong>Torque Ripple</strong></td>
<td>High</td>
<td>Low</td>
</tr>
<tr>
<td><strong>Average torque controlled</strong></td>
<td>No</td>
<td>Yes (Indirectly)</td>
</tr>
<tr>
<td><strong>Current maintained within m/c rating</strong></td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td><strong>Complexity of architecture</strong></td>
<td>Low</td>
<td>Medium</td>
</tr>
<tr>
<td><strong>Current distortion and harmonics</strong></td>
<td>High</td>
<td>Medium</td>
</tr>
<tr>
<td><strong>Fixed switching frequency</strong></td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td><strong>Flux control</strong></td>
<td>Poor at low load and speed</td>
<td>Excellent</td>
</tr>
<tr>
<td><strong>Parameter sensitivity</strong></td>
<td>Low</td>
<td>Medium</td>
</tr>
</tbody>
</table>

Wolbank et al. [6] compare low and zero-speed applications of DTC and sensorless FOC. They study steady-state stability and speed overshoot, where FOC shows slower dynamics but better steady-state tracking compared to DTC. As both FOC and DTC have drawbacks, an interesting combination of DTC and FOC is presented in [10]. The resulting direct torque and stator flux control method (DTFC) does not use voltage modulation, current regulation loops, coordinate transformations, or voltage decoupling. While Kazmierkowski simulates and experimentally validates three control techniques, DTC, IFOC and feedback-linearization, he only offers plots and neglects a numerical comparison. Casa de et al. [11] evaluate standard DTC and DFOC and present a unique scheme called discrete space vector modulation (DSVM), which is a
variation of the standard SVM. Performance criteria are steady-state current and torque ripples, and dynamic response due to a torque step.

Comparisons of other drives focus on steady-state response. Thomas [12] propose and experimentally validate geometric sliding mode/limit cycle control. Three different inverter modes, asynchronous, synchronous, and square wave, are analyzed. Industrial control objectives such as stator and rotor flux regulation, torque and speed/position control, minimal energy and harmonics criteria, and optimization of torque pulsations are evaluated. Sorchini and Krein [13] prove that DTC is independent of SVM, and decouple the motor drive from the inverter switching scheme. They discuss the singular perturbation method (SPM) and its application to the control of induction motors and prove that the switching strategy for SPM and standard DTC are equivalent. The SPM DTC controller was implemented with PWM, rather than SVM, but was not validated experimentally.

The best attempt found in the literature for comparing FOC and DTC, but not IFOC, is [5], where dynamic performance of both drives is compared and sensitivity analyses are done with respect to stator resistance for DTC and rotor time constant for FOC. Drawbacks in [5] include a “verbal” comparison of torque and flux dynamics, and parameter sensitivities. Le-Huy states, “It is difficult to clearly state on [sic] the superiority of DTC versus FOC because of the balance of the merits of the two schemes. Based on the simulation results, we can nevertheless say that the two control schemes provide, in their basic configuration, comparable performance regarding torque control performance and parameter sensitivity. We can note a slight advance of [the] DTC scheme compared to [the] FOC scheme regarding the dynamic flux performance and the
implementation complexity” [pp 1252 of 5]. Le-Huy echoes the remarks of Kazmierkowski, giving a slight advantage to DTC in dynamic performance for the flux response. Besides this, not much attention is paid to comparing the two drives in this paper.

Vasudevan and Arumugam [14] compare IFOC to DFOC along with classical DTC-SVM and direct torque neuro-fuzzy control using MATLAB/Simulink. Stator voltages and currents, angular velocity, torque, and flux responses to a change in torque or angular velocity, are compared. The effect of parameter variation, such as stator resistance variation due to temperature increases, is also discussed in relation to the DTC control method.

As for experimental hardware validation, many authors ignore it completely. No hardware validations are presented in [2], [4], [5], [7], [11], [15] and others. [15] presents an interesting approach targeting the operation of DFOC, DTC with PWM and DTC-SVM under a driving cycle of an electric vehicle, but no hardware validation is performed.

Now that a general literature review is complete, we will concentrate on the background for each individual motor drive. The following four sections go into more depth in order to set the stage for the analytical development in the next chapter. For each drive, a brief literature review is performed on current individual drive research to give a better setting; also, the theoretical underpinnings are discussed.
2.5 Direct Torque Control

Direct torque control (DTC) is an IM drive that is frequently used in IM control because of its balance between simplicity of design and its decent performance. It is a motor drive that uses little parameter information, and therefore is generally not considered to be a high performance drive. It essentially has two user inputs: torque and stator flux. The typical DTC IM drive uses these two inputs in hysteretic control as inputs to a look-up switching state table. In addition to the previous two inputs, standard DTC uses the angle of the stator flux to determine its “sector.” The typical table uses six flux sectors to distinguish where the flux angle lies. The output of the table is a vector that contains the information telling which gates of the inverter should be on at any point in time. It should be emphasized that this configuration is not necessarily optimal, but it is indeed common.

Sikorski et al. [16] compare linear DTC-SVM to nonlinear DTC methods, such as DTC-$\delta$, DTC-2x2, and DTFC-3A, using steady-state performance metrics. Excellent numerical hardware results were given that compared the variations in DTC. The average switching frequency was kept the same throughout these trials in an attempt to keep one variable constant. It was found that the ripple for the current and torque was found to be smallest in the standard DTC case, which was unexpected considering the higher level of complexity using the other methods.

2.6 Indirect Field-Oriented Control

Indirect field-oriented control, or IFOC, is the most common IM drive because of its use of moderate amounts of parameter information to give it respectable performance
Field-oriented control (FOC) was introduced by Blaschke in 1971 [17, 18]. FOC was created to imitate the control of a separately excited dc motor. In a similar fashion to the dc motor, the FOC drive keeps the rotor flux perpendicular to the stator flux to get the maximum output torque possible. The big advantage of FOC is that the flux and the torque can be decoupled by insuring that the other is in steady state. In this fashion, the dynamics can be independently controlled by the user. Because of this, the classical feedback control can be used to obtain desired motor performance. The basic attribute of IFOC is that it uses an estimate of the rotor flux in determining the next state of the inverter. In particular, it uses the angle of the rotor flux to determine where the flux is in vector-space. The angle is calculated by (21).

$$\rho = \tan^{-1}\left(\frac{\lambda_{qr}}{\lambda_{dr}}\right)$$  \hspace{1cm} (21)

This flux vector angle is then used in a matrix transformation that converts the stator current and rotor flux values into a new state space \(\{\omega_r, \psi, i_q, i_d, \rho\}\) [19] where

$$\begin{align*}
\omega_r &= \omega_r \\
\psi &= \sqrt{\lambda_{qr}^2 + \lambda_{dr}^2} \\
i_q &= \frac{\lambda_{dr}i_{qs} - \lambda_{qr}i_{ds}}{\psi} \\
i_d &= \frac{\lambda_{dr}i_{ds} + \lambda_{qr}i_{qs}}{\psi} \\
\rho &= \tan^{-1}\left(\frac{\lambda_{qr}}{\lambda_{dr}}\right)
\end{align*}$$  \hspace{1cm} (22)
This is equivalent to using the following matrix conversion for the same currents and flux values:

\[
\begin{pmatrix}
i_d \\
i_q
\end{pmatrix} = \Gamma_p \begin{pmatrix}
i_{ds} \\
i_{qs}
\end{pmatrix}
\]

\[
\begin{pmatrix}
\psi \\
0
\end{pmatrix} = \Gamma_p \begin{pmatrix}
\lambda_{dr} \\
\lambda_{qr}
\end{pmatrix}
\]

where

\[
\Gamma_p = \begin{pmatrix}
cos(\rho) & sin(\rho) \\
-sin(\rho) & cos(\rho)
\end{pmatrix}
\]

If we let the vector be equal to that in (24),

\[
\begin{pmatrix}
v_{qs} \\
v_{ds}
\end{pmatrix} = \psi \begin{pmatrix}
\lambda_{dr} & \lambda_{qr} \\
-\lambda_{qr} & \lambda_{dr}
\end{pmatrix}^{-1} \begin{pmatrix}
v_d \\
v_q
\end{pmatrix}
\]

where

\[
\begin{pmatrix}
v_d \\
v_q
\end{pmatrix} = \sigma \begin{pmatrix}
-n_p \omega r_i_q - \frac{Mr_i_q^2}{L_r \psi} + u_{flux} \\
n_p \omega r i_d + \frac{n_p \omega r M \psi}{\sigma L_r} + u_{speed}
\end{pmatrix}
\]

and where

\[
\sigma = \frac{L_r L_r - M^2}{L_r}
\]

the unwanted nonlinear terms cancel, and the closed loop dynamic system equations become similar to that of a dc motor, as seen in (25). The quadrature axis current represents the speed-producing element, while the direct axis current represents the torque-producing element. As revealed in (25), \(i_q\) and \(i_d\) are asymptotically decoupled in this reference frame. This will allow the user to independently control their steady-state values as well as their dynamic performance. The only downside of FOC is the nonlinear nature of \(\rho\) and the fact that it is already very difficult to estimate accurately. This is a common downside of many field-oriented controllers.
2.7 Feedback Linearization: Input-Output Decoupling Control

In [20], Krzeminski comes up with a type of nonlinear state feedback control which is completely input-output decoupled at all times even through transients. This differs from IFOC because IFOC is decoupled only when flux and speed are in steady state. FL-IODC achieves better performance than IFOC in theory due to accounting for the stator resistive drop and other terms that allow it to have complete decoupling. The drawback of this type of control is the additional parameter sensitivity that results from the addition of these extra terms. The inputs in this control scheme are $v_{qs}$ and $v_{ds}$, while the outputs are the rotor mechanical speed, $\omega_r$, and the flux magnitude squared, $\psi^2$. Krzeminski assumes in the paper that the load torque response is known. If this is the case, then the new state space is:

\[
\begin{align*}
\frac{d\omega_r}{dt} &= \frac{3n_p M \psi i_q}{2JL_r} - \frac{T_{\text{load}}}{J} \\
\frac{d\psi}{dt} &= -\frac{r_s \psi}{L_r} - \frac{r_r M i_d}{L_r} \\
\frac{di_q}{dt} &= -\left(\frac{M^2 r_s}{\sigma L_r^2} + \frac{r_s}{\sigma}\right) i_q + u_{\text{speed}} \\
\frac{di_d}{dt} &= -\left(\frac{M^2 r_s}{\sigma L_r^2} + \frac{r_s}{\sigma}\right) i_d + \frac{r_r M \psi}{L_r} + u_{\text{flux}} \\
\frac{d\rho}{dt} &= n_p \omega_r + \frac{r_r M i_q}{L_r \psi} 
\end{align*}
\]
\[ y_1 = \omega_r, \]
\[ y_2 = \frac{d\omega_r}{dt}, \]
\[ y_3 = \psi^2 = \lambda_{qr}^2 + \lambda_{dr}^2, \]
\[ y_4 = \frac{dy_3}{dt} = \frac{d\psi^2}{dt} = \frac{d(\lambda_{qr}^2 + \lambda_{dr}^2)}{dt}, \]
\[ y_5 = \tan^{-1}\left(\frac{\lambda_{qr}}{\lambda_{dr}}\right) = \rho \]  

(26)

From (26) and the original dynamic motor model, we get the following dynamic system, as seen in [21]:

\[ \frac{dy_1}{dt} = y_2, \]
\[ \frac{dy_2}{dt} = f_{21}(y_1, \ldots, y_5) + f_{22}(y_1, \ldots, y_5)v_{ds} + f_{23}(y_1, \ldots, y_5)v_{qs}, \]
\[ \frac{dy_3}{dt} = y_4, \]
\[ \frac{dy_4}{dt} = f_{41}(y_1, \ldots, y_5) + f_{42}(y_1, \ldots, y_5)v_{ds} + f_{43}(y_1, \ldots, y_5)v_{qs}, \]
\[ \frac{dy_5}{dt} = f_5 \]  

(27)

If one sets \( v_{qs} \) and \( v_{ds} \) to the vector

\[
\begin{pmatrix}
    v_{ds} \\
    v_{qs}
\end{pmatrix} =
\begin{pmatrix}
    f_{22}(y_1, \ldots, y_5) & f_{23}(y_1, \ldots, y_5) \\
    f_{42}(y_1, \ldots, y_5) & f_{43}(y_1, \ldots, y_5)
\end{pmatrix}^{-1}
\begin{pmatrix}
    -f_{21}(y_1, \ldots, y_5) + u_{speed} \\
    -f_{41}(y_1, \ldots, y_5) + u_{flux}
\end{pmatrix}
\]  

(28)

the system now looks like (29)
\[
\frac{dy_1}{dt} = y_2 \\
\frac{dy_2}{dt} = u_{\text{speed}} \\
\frac{dy_3}{dt} = y_4 \\
\frac{dy_4}{dt} = u_{\text{flux}} \\
\frac{dy_5}{dt} = n_p y_1 + \frac{2 r_r}{3 n_p y_3} (J y_2 + T_{\text{load}})
\] (29)

The following inputs in (30) can be set to completely decouple the inputs from the outputs when using constant design parameters \(k_1, k_2, k_3, \text{and } k_4\). Thus, if there is a transient in the flux magnitude squared or the speed of the rotor, the transient will not affect the other variable [17].

\[
\begin{align*}
    u_{\text{speed}} &= -k_1 \left( \omega - \omega_{\text{ref}} \right) - k_2 \left( \frac{T_c}{J} - \frac{T_{\text{load}}}{J} - \hat{\omega}_{\text{ref}}(t) \right) + \hat{\omega}_{\text{ref}}(t) \\
    u_{\text{flux}} &= k_3 \left( \psi^2 - \psi_{\text{ref}}^2 \right) - k_4 \left( \frac{2 R}{L_r} \left( M \left( \psi_d i_d + \psi_q i_q \right) - \left( \psi_d^2 + \psi_q^2 \right) \right) - \psi_{\text{ref}}^2 \right) + \psi_{\text{ref}}^2
\end{align*}
\] (30)

2.8 Vectorized Volts-per-Hertz

Vectorized volts-per-hertz is by far the simplest motor drive since it requires no parameter knowledge and is essentially an open-loop drive. Similar to the standard volts-per-hertz, it requires a desired operating frequency, \(f^*\), from the user to create a desired voltage on the IM. In addition to the frequency, it also requires a current, \(i_d^*\), to run the drive. This current essentially creates a desired voltage vector, \(v_d^*\), via the stator resistor, \(R_s\). From these two variables, \(i_d^*\) and \(f^*\), two reference voltage vectors, \(v_d^*\) and \(v_q^*\), are created that are used in driving the inverter output. With the knowledge of both \(v_d^*\)
and $v^*_q$, the whole voltage vector is created. Another way of looking at this is that $v^*_d$ and $v^*_q$ represent the voltage vector in rectangular coordinates, but it can also be thought of as a voltage vector in polar coordinates with a magnitude, $|V|$, and an angle, $\theta$.

2.9 Motor Control Continuum

The above four motor controllers—vectorized volts-per-hertz, DTC, IFOC and feedback-linearization—along with all other motor controllers are part of a motor controller complexity continuum. This idea, pioneered in [22], lets one quickly understand what types of motor controllers are more algorithmically complex than others. This helps influence the controller selection in various applications. Typically, the more complex an algorithm is, the more parameters are required. This also comes with an increase in performance, but also higher required computational power. Figure 2 shows the complexity continuum as described in [22].

![Figure 2. Motor Control Complexity Continuum](image)

As expected, scalar methods, such as constant volts-per-hertz control, are at the left of the scale. These methods require little parameter information, and the computational complexity is low. In the middle of the continuum, we find the FOC family. This family consists of a variety of field-oriented controls, including DFOC and IFOC, and requires a medium amount of parameter information and computation. To the right of this, there are observer-based nonlinear controllers, as shown by the “[6]” on the continuum. Finally, at the right end, we find feedback linearization, which allows for the highest
transient performance with the highest computational complexity and parameter knowledge. Using this complexity continuum, along with a little knowledge of drives, one should be able to pick a particular motor controller that will fit the specific application.
3. ANALYTICAL DEVELOPMENT

In this chapter, the four drive types will be analyzed by doing stability analysis, parameter sensitivity analyses, and several formal derivations.

3.1 Theoretical Basis for DTC

In this section, singular perturbation theory and sliding-mode control will be described. DTC will then be formally derived and global stability will be proved. A parameter sensitivity analysis will then be conducted.

3.1.1 Singular perturbation theory and sliding-mode control

Two methods are required to thoroughly understand a formal derivation of DTC from a mathematical standpoint: singular perturbation theory and sliding-mode control. These two methods will be explained to further understand a formal derivation of DTC. One tool used in the singular perturbation method is composite feedback control [23, 24]. This tool decomposes signals into “fast” and “slow” components, as shown in (31). By using this tool, the method is simplified so that the two different signal dynamics are controlled independently.

\[ u = u_{\text{slow}} + u_{\text{fast}} \]  

(31)

The above composite feedback control comes naturally after looking at a singularly perturbed system, which is a system that demonstrates more than one time-scale behavior. The standard singular perturbation system form looks like (32), where there are two functions, each of which can contain the variable in the first equation, the variable in the second equation, the input, and a small parameter, \( \varepsilon \). The first defines the function for the slow variable, while the second function defines the function for the fast
variable. The second equation is multiplied by $\varepsilon$, and when the limit is taken as $\varepsilon$ goes to zero, the second equation drops out, and in effect the result is a reduced order system. Conversely, if one put (32) in the fast time scale, where the fast time is defined by (33), then the resultant fast system would look like (34). This system is considered one in dynamic steady state [23], where the system remains “$\varepsilon$-close” to the expected system.

\begin{align*}
\frac{dx}{dt} &= f(x, z, \varepsilon, u) \\
\varepsilon \frac{dz}{dt} &= g(x, z, \varepsilon, u) \\
t_f &= \frac{t}{\varepsilon}
\end{align*}

\begin{align*}
\frac{dx}{dt_f} &= \varepsilon f(x, z, \varepsilon, u) \\
\frac{dz}{dt_f} &= g(x, z, \varepsilon, u)
\end{align*}

Sliding-mode control’s main element is a sliding surface, frequently called a “sliding manifold,” where the surface is defined by $s=0$. The goal of the controller is to stay close to this surface by switching actuator states, for example an inverter. The control is discontinuous and nonlinear. Because it is discontinuous it can reach a desired motor state in a finite amount of time, but also typically uses a hysteretic switching scheme for its realization. This is in stark contrast to a control mode in which the motor asymptotically reaches steady state, which theoretically takes an infinite amount of time. Sliding mode control guarantees that regardless of the initial condition, the states will “slide” along the sliding surface and arrive in steady state. The Cauchy–Lipschitz theorem guarantees the existence and uniqueness of this sliding manifold with a given initial condition. If the nonlinear system generally defined as (35) uses an input, $u$, of the
form in (36), it is considered to have a sliding mode controller. The functions \( h \) and \( g \) can be unknown, but both are functions of the variable being controlled, in this case, \( x \). The function \( \beta \) defines the uncertainty of the system, and is normally just a large constant. This function will be discussed more subsequently. It is important to note that the fast and slow components of the signal will be controlled via sliding mode controllers.

\[
\frac{dx}{dt} = h(x) + g(x)u 
\]  
(35)

\[
u = -\beta(x)\text{sgn}(s)
\]  
(36)

Converting (25) into one with only stator flux and current components yields (37) [13]. This is known as the standard singular perturbation form, where \( \sigma \) is the perturbation parameter.

\[
\frac{d\omega}{dt} = \frac{1}{J}(T_e - T_i)
\]

\[
\frac{d\lambda_{dc}}{dt} = -R_s i_{ds} + u_{ds}
\]

\[
\frac{d\lambda_{qs}}{dt} = -R_s i_{qs} + u_{qs}
\]

\[
\frac{\sigma di_{ds}}{dt} = \frac{R_r}{L_s L_r} \dot{\lambda}_{ds} + \frac{n_p}{L_s} \omega \lambda_{qs} - \sigma n_e \omega i_{qs} - \frac{\gamma}{L_s} i_{ds} + \frac{1}{L_s} u_{ds}
\]

\[
\frac{\sigma di_{qs}}{dt} = \frac{R_r}{L_s L_r} \dot{\lambda}_{qs} - \frac{n_p}{L_s} \omega \lambda_{qs} + \sigma n_e \omega i_{ds} - \frac{\gamma}{L_s} i_{qs} + \frac{1}{L_s} u_{qs}
\]

where \( \sigma = 1 - \frac{T_m^2}{L_s L_r} \) and \( \gamma = \frac{L_s R_r}{L_r} + R_s \)

To get the above differential equations to a more useful form, it is prudent to convert to the flux reference frame. To go about this, we must use the following transformation, shown below in (38) which transforms the input stator voltages. The model defined in (39) is the transformed singular perturbation form for an induction
motor. In this model, the speed, flux angle and the flux magnitude squared are the slow variables in the composite feedback control. The fast variables are the normalized torque, \( \tau \), and \( \eta \).

\[
\begin{pmatrix}
  u_\phi \\
  u_\tau
\end{pmatrix} = \begin{pmatrix}
  \cos \rho & \sin \rho \\
  -\sin \rho & \cos \rho
\end{pmatrix}
\begin{pmatrix}
  u_{ds} \\
  u_{qs}
\end{pmatrix}
\] (38)

\[
\frac{d \omega}{dt} = \frac{1}{J} \left( \frac{3}{2} n_p \tau - T_L \right)
\]

\[
\frac{d \phi}{dt} = -2 \left( R \eta - \sqrt{\phi} u_\phi \right)
\]

\[
\frac{d \rho}{dt} = -R_s \frac{\tau}{\phi} + \frac{1}{\sqrt{\phi}} u_\tau
\] (39)

\[
\begin{aligned}
\sigma \frac{d \tau}{dt} &= -\gamma \frac{R_s}{L_s} \tau - n_p \omega \phi + \frac{1}{L_s} \sqrt{\phi} u_\tau + \sigma \left( n_p \omega \eta + \frac{1}{\sqrt{\phi}} \left( -\eta u_\tau + \tau u_\phi \right) \right) \\
\sigma \frac{d \eta}{dt} &= -\gamma \frac{R_s}{L_s} \eta + \frac{R_s}{L_s L_r} \phi + \frac{1}{L_s} \sqrt{\phi} u_\phi - \sigma \left( n_p \omega \tau + R_s \left( \frac{\eta^2 + \tau^2}{\phi} \right) + \frac{1}{\sqrt{\phi}} \left( -\eta u_\phi - \tau u_\tau \right) \right)
\end{aligned}
\]

The last items required for the above model to operate correctly are the phase voltages. They are defined by the vector in (41). This is merely the inverse of Equation (38) when a two-phase to three-phase transformation, shown in (40), is applied. The inputs for Equation (41) are shown in (42).

\[
\begin{pmatrix}
  u_1 \\
  u_2 \\
  u_3
\end{pmatrix} = \begin{pmatrix}
  \cos \rho & -\sin \rho \\
  \cos \left( \rho - \frac{2}{3} \pi \right) & -\sin \left( \rho - \frac{2}{3} \pi \right) \\
  \cos \left( \rho + \frac{2}{3} \pi \right) & -\sin \left( \rho + \frac{2}{3} \pi \right)
\end{pmatrix}
\begin{pmatrix}
  u_\phi \\
  u_\tau
\end{pmatrix}
\] (40)
\[
\begin{pmatrix}
  u_a \\
  u_b \\
  u_c
\end{pmatrix} =
\begin{pmatrix}
  \cos \rho & -\sin \rho \\
  \cos \left( \rho - \frac{2}{3} \pi \right) & -\sin \left( \rho - \frac{2}{3} \pi \right) \\
  \cos \left( \rho + \frac{2}{3} \pi \right) & -\sin \left( \rho + \frac{2}{3} \pi \right)
\end{pmatrix}
\begin{pmatrix}
  u_\phi \\
  u_\tau
\end{pmatrix}
\] (41)

\[
u_\phi = -k_\phi \text{sgn}(e_\phi)
\]

\[
u_\tau = -k_\tau \text{sgn}(e_\tau)
\] (42)

where \( k_\phi > k_{\phi,\text{min}} \) and \( k_\tau > k_{\tau,\text{min}} \)

By setting \( \sigma = 0 \), the resulting model is (43) [13]. It should be noted that there are no fast components in this model. Here, the induction motor dynamic model is in quasi steady state. The equation that governs the fast dynamics is given in (44) [13].

\[
\frac{d\omega}{dt} = \frac{1}{J} \left( -\frac{3}{2} n_p \tau - T_L \right)
\]

\[
\frac{d\phi}{dt} = -2 \left( R_\phi \eta - \sqrt{\phi} u_{\phi,\text{slow}} \right)
\]

\[
\frac{d\rho}{dt} = -R_\rho \frac{\tau}{\phi} + \frac{1}{\sqrt{\phi}} u_{\tau,\text{slow}}
\] (43)

\[0 = -\frac{\gamma}{L_s} \tau - \frac{n_p}{L_s} \omega \phi + \frac{1}{L_s} \sqrt{\phi} u_{\tau,\text{slow}}\]

\[0 = -\frac{\gamma}{L_s} \eta + \frac{R_\tau}{L_s L_\tau} \phi + \frac{1}{L_s} \sqrt{\phi} u_{\phi,\text{slow}}\]

\[\eta = \frac{1}{\gamma} \left( \frac{R_\tau}{L_s} \phi + \sqrt{\phi} u_{\phi,\text{slow}} \right)\] (44)

Using (44) and the second equation of (43), the state equation for the flux magnitude squared is given in (45) [13]. This equation is now in the sliding mode control form, in which the sliding surface is defined by (46), and its time derivative given by (47) [13].

32
\[ \frac{d\phi}{dt} = -\frac{2R_s R_p}{\gamma L_s} \phi + 2\sqrt{\phi} \left( \frac{\gamma - R_s}{\gamma} \right) u_{\phi,\text{slow}} \]
\[ = h_\phi(\phi) + g_\phi(\phi) u_{\phi,\text{slow}} \]
\[ s_\phi = \phi - \phi_{\text{ref}} = e_\phi = 0 \]  \hspace{1cm} (46)

\[ \frac{ds_\phi}{dt} = \frac{d\phi}{dt} = h_\phi(\phi) + g_\phi(\phi) u_\phi \]  \hspace{1cm} (47)

3.1.2 Direct torque control stability analysis

A flux controller given by (49) is proved to be stable in [25] using the Lyapunov function \( V_\phi = .5 s_\phi^2 \) and is shown in (48). In (49), the error in the flux is given by \( e_\phi \). It should be noted that there must be a remnant flux in the motor to insure that with any initial condition, the flux will be stable and regulated. Typically, there exists a remnant flux already in the motor, but to be sure, a voltage pulse should be used to initialize the motor controller before operation.

\[ \frac{dV_\phi}{dt} = s_\phi \frac{ds_\phi}{dt} = s_\phi h_\phi(\phi) + g_\phi(\phi) s_\phi u_\phi \]
\[ \leq g_\phi(\phi) k_{\phi,\text{min}} |s_\phi| - g_\phi(\phi) (k_{\phi,\text{min}} + \delta) |s_\phi| \]
\[ = -\delta g_\phi(\phi) |s_\phi| \]
\[ \leq -2\delta \left( \frac{R_s + \gamma}{\gamma} \right) \sqrt{\phi_0} |s_\phi| \]

\[ u_\phi = - (k_{\phi,\text{min}} + \delta) \text{sgn}(s_\phi) = -(k_{\phi,\text{min}} + \delta) \text{sgn}(e_\phi) \]
where \( k_{\phi,\text{min}} = \frac{2R_s}{L_s} \sqrt{\phi_{\text{rated}}} \) and \( \delta > 0 \).
While the flux needs to be regulated at some constant value given by the user, the torque needs to have tracking capabilities. But unlike the slow flux variable, the torque variable will be considered to have a fast component and a slow component. The reduced induction motor model in the fast time scale is shown in (50) where $t_f = t / \sigma$ and $\sigma = 0$.

\[
\frac{d\omega}{dt_f} - \frac{d\phi}{dt_f} = \frac{d\rho}{dt_f} = 0
\]

\[
\frac{d\tau}{dt_f} = -\frac{\gamma}{L_s} \tau - \frac{n_p}{L_s} \omega \phi + \frac{1}{L_s} \sqrt{\phi} \left( u_{\tau,\text{slow}} + u_{\tau,\text{fast}} \right)
\]

\[
\frac{d\eta}{dt_f} = -\frac{\gamma}{L_s} \eta + \frac{R_s}{L_s L_i} \phi + \frac{1}{L_s} \sqrt{\phi} \left( u_{\tau,\text{slow}} + u_{\tau,\text{fast}} \right)
\]

The only equation that is needed from (50) is the second, where it is rewritten in (51) to be in the sliding mode control form while setting the slow torque components to zero.

\[
\frac{d\tau}{dt_f} = -\frac{\gamma}{L_s} \tau - \frac{n_p}{L_s} \omega \phi + \frac{1}{L_s} \sqrt{\phi} \left( u_{\tau,\text{fast}} \right)
\]

\[
= h_{\tau} (\tau, \omega, \phi) + g_{\tau} (\phi) u_{\tau,\text{fast}}
\]

The sliding surface is defined in (52) and its time derivative in (53). In (53), it is assumed that the time derivative with respect to the fast time torque component is zero, which says that the variation $\tau_{\text{ref}}$ is slow when compared to the fast time torque component.

\[
s_{\tau} = \tau - \tau_{\text{ref}} = e_{\tau} = 0
\]

\[
\frac{ds_{\tau}}{dt_f} = \frac{d\tau}{dt_f} = h_{\tau} (\tau, \omega, \phi) + g_{\tau} (\phi) u_{\tau,\text{fast}}
\]

The sliding mode torque controller is given by (54). In this case, the sliding mode variable, $s_{\tau}$, is defined to be equivalent to the torque error, $e_{\tau}$. The stability is again
guaranteed by doing a stability analysis using the Lyapunov function $V_\phi = .5s_\tau^2$ and insuring that a remnant flux exists in the induction motor before running.

$$u_{r,\text{fast}} = -(k_{r,\text{min}} + \delta)\text{sgn}(s_r) = -(k_{r,\text{min}} + \delta)\text{sgn}(e_r)$$  \hspace{1cm} (54)

The full controller is now given by (55), where it is recommended to use gain values greater than required in order to be robust. These inputs are now transformed to actual input voltages by using (41).

$$u_\phi = u_{\phi,\text{slow}} + u_{\phi,\text{fast}} = -(k_{\phi,\text{min}} + \delta)\text{sgn}(e_\phi)$$

$$u_r = u_{r,\text{slow}} + u_{r,\text{fast}} = -(k_{r,\text{min}} + \delta)\text{sgn}(e_r)$$ \hspace{1cm} (55)

where $\delta > 0$

The typical DTC controller uses six quantization steps, also known as sectors, but as stated before, this is not necessary. As an example, in the next chapter, 6, 12 and 256 sector DTC will be compared. Because of the quantization of the flux angle, there will be a quantization error that occurs, and the gain in (55) will need to be increased to insure stability. This equation for the quantization of the flux angle is given by (56) where $n$ is the number of sectors and $\rho$ is the flux angle [13]. It was shown in [13] that the minimum number of sectors that preserve the sign information using (40) is five, but this does not insure that a quality voltage waveform is output. The final controller that includes this quantization error gain is defined in (57). A 3φ inverter is used to create the desired voltages. This restricts the voltage possibilities, with values of $\pm 2/3 \ V_{dc}$, $\pm 1/3 \ V_{dc}$ and 0, where $V_{dc}$ is the dc bus voltage. It is also shown in [13] that a gain of three is needed for inverter stability. Therefore, for the torque and flux controllers to be stable in the sense of Lyapunov, the two conditions in (58) must be met [13].
\[
\rho_q = \frac{2\pi}{n} \text{round}\left(\frac{n\rho}{2\pi}\right) \tag{56}
\]

\[
u_\phi = -k_\phi k_\phi \text{sgn}(e_\phi) \tag{57}
\]

\[
u_z = -k_\phi k_z \text{sgn}(e_z) \tag{58}
\]

\[
k_\phi \rho < \frac{\sqrt{3}}{3},
\]

\[
k_\phi \rho < \frac{\sqrt{3}}{3}
\]

Assuming the above conditions are met, the vector in (59), \(w\), is used as the signals for the gates of the inverter [13]. In this configuration, if the component of the vector is positive, this indicates that the upper gate on that phase should be activated, while a negative signal indicates the lower gate on that phase should be activated. Since the resulting switching strategy for the six sector sliding mode controller is the same as a standard DTC switching table, DTC has been formally derived. The product of the six sector controllers is shown in Table 3 [13].

\[
w = \begin{pmatrix}
\cos \rho_q & -\sin \rho_q \\
\cos \left(\rho_q - \frac{2\pi}{3}\right) & -\sin \left(\rho_q - \frac{2\pi}{3}\right) \\
\cos \left(\rho_q + \frac{2\pi}{3}\right) & -\sin \left(\rho_q + \frac{2\pi}{3}\right)
\end{pmatrix}
\tag{59}
\]

### Table 3. Switching Strategy for the Quantized Sliding Mode Controller with n=6

<table>
<thead>
<tr>
<th>(\rho) (\in) ((-\pi/6, \pi/6))</th>
<th>(\rho) (\in) ((\pi/6, \pi/2))</th>
<th>(\rho) (\in) ((\pi/2, 2\pi/3))</th>
<th>(\rho) (\in) ((2\pi/3, \pi))</th>
<th>(\rho) (\in) ((-\pi/6, \pi/2))</th>
<th>(\rho) (\in) ((-\pi/2, -\pi/6))</th>
</tr>
</thead>
<tbody>
<tr>
<td>(e_\phi &gt; 0)</td>
<td>(e_\tau &gt; 0)</td>
<td>((-1,-1,1))</td>
<td>((1,-1,1))</td>
<td>((1,-1,-1))</td>
<td>((1,1,1))</td>
</tr>
<tr>
<td>(e_\phi &gt; 0)</td>
<td>(e_\tau &lt; 0)</td>
<td>((-1,-1,1))</td>
<td>((-1,-1,-1))</td>
<td>((-1,1,1))</td>
<td>((-1,1,1))</td>
</tr>
<tr>
<td>(e_\phi &lt; 0)</td>
<td>(e_\tau &gt; 0)</td>
<td>((1,-1,1))</td>
<td>((1,-1,-1))</td>
<td>((-1,-1,1))</td>
<td>((-1,-1,1))</td>
</tr>
<tr>
<td>(e_\phi &lt; 0)</td>
<td>(e_\tau &lt; 0)</td>
<td>((1,1,-1))</td>
<td>((-1,1,-1))</td>
<td>((1,-1,1))</td>
<td>((1,-1,1))</td>
</tr>
</tbody>
</table>
3.1.3 Direct torque control parameter sensitivity analysis

Due to the variations in motor parameters, the dynamic responses of the drives are expected to change accordingly. When DTC is used with a switching table, it is only dependent on the stator resistance, but when used with space vector pulse width modulation, or SVPWM, it needs an estimate of the electrical angle, \( \theta_e \), which is in turn sensitive to the rotor resistance, \( r_r \), and the rotor inductance, \( L_r \). The sensitivity related to the estimation of the electrical angle is expected to be small compared to other sensitivities, as \( \theta_e \) is used for transformations and reverse transformations from one frame to another, and impact may cancel out.

It is possible to build a Jacobian matrix, \( J \), in which the sensitivities of torque, speed, and other desired variables or outputs are estimated relative to any change in motor parameters. For DTC with a switching table, the expression for DTC sensitivity is given by (60).

\[
\begin{pmatrix}
\Delta T^e \\
\Delta \omega_{rm}
\end{pmatrix} = J_{DTC-ST} [\Delta r_r]
\]

where

\[
J_{DTC-ST} = \begin{pmatrix}
\frac{\partial T^e}{\partial r_s} \\
\frac{\partial \omega_{rm}}{\partial r_s}
\end{pmatrix}
\]  

(60)

For DTC with SVPWM, the Jacobian would be that defined by (61).
\[
\begin{pmatrix}
\Delta T^e \\
\Delta \omega_{rm}
\end{pmatrix} = J_{DTC-SVPWM} \begin{pmatrix}
\Delta r_s \\
\Delta L_r \\
\Delta r_r
\end{pmatrix}
\]

where \( J_{DTC-SVPWM} = \begin{pmatrix}
\frac{\partial T^e}{\partial r_s} & \frac{\partial T^e}{\partial L_r} & \frac{\partial T^e}{\partial r_r} \\
\frac{\partial \omega_{rm}}{\partial r_s} & \frac{\partial \omega_{rm}}{\partial L_r} & \frac{\partial \omega_{rm}}{\partial r_r}
\end{pmatrix} \)

(61)

3.2 Indirect Field-Oriented Control

In this section, global asymptotic stability will be proved for indirect field-oriented control. A parameter sensitivity analysis will then be performed for IFOC.

3.2.1 Indirect field-oriented control stability analysis

Since IFOC relies on the use of an estimated stator flux, the observer must be proved to be stable in the sense of Lyapunov. It would be prudent to see if a solution exists that makes the control scheme globally asymptotically stable (GAS) using Lyapunov’s second method. With this method, it is required—just as in the DTC case—that the Lyapunov function (LF) is positive definite and its derivative is negative definite. In this case, the candidate LF given in (62) is composed from four positive semi-definite matrices in (63) [26]. This candidate LF is only valid if \( R_r \) is identically \( \hat{R}_r \).

All of the \( a \) and \( b \) values below are positive constants, made up of combinations of resistance and gain values.

\[
V(w) = \frac{1}{2} w^T P_w
\]

(62)
\[
P = (z_1 + z_2)P_1 + \left(\frac{K_I + R_r^2K_L}{R_r}\right)P_2 + P_3 + P_4
\]

where \(P_1 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}, P_2 = \begin{pmatrix} 1 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & \hat{R}_r \end{pmatrix},
\]

\[
P_3 = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & K_p \\ 0 & 0 & K_p & K_p^2 \end{pmatrix}, P_4 = \begin{pmatrix} K_p^2 & 0 & K_p & 0 \\ 0 & 0 & 0 & 0 \\ K_p & 0 & h^2 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix},
\]

\(K_p\) is the proportional constant
\(h\) is a positive constant
\(z_1 = \frac{b_{13}^2}{R_r a_3}\) and
\(z_2 = \frac{b_{14}^2}{R_r a_4}\)

The derivative of this candidate LF is shown in (64). Assuming that all \(a\) and \(b\) constants are positive, the derivative of the candidate LF is shown in [26] to be always negative, or negative definite. This along with the fact that the candidate LF is positive definite makes it an LF.

\[
\dot{V}(w) = -a_1 w_1^2 - a_3 w_3^2 - a_4 w_4^2 - 2b_{13} w_1 w_3 + 2b_{14} w_1 w_4
\]

\[
= \left(\frac{h_{13}^2}{a_3} - \frac{h_{14}^2}{a_4}\right)w_1^2 - \left(\frac{h_{13}^2}{a_3} - \frac{h_{14}^2}{a_4}\right)w_2^2
\]

(64)

For the other case where \(R_r\) is not identically equal to \(\hat{R}_r\), the proposed candidate LF from [26] is shown in (65). The same constituent P matrices in the previous case are again used in this candidate LF. The derivative of this candidate LF is shown in (66), where the constants \(\varepsilon_1\) and \(\varepsilon_2\) are off-diagonal coefficients for the constant symmetric
matrix $Q(\varepsilon_1, \varepsilon_2)$. Since $Q(\varepsilon_1, \varepsilon_2)$ is shown to be positive definite in [26], the derivative is negative definite, and the candidate LF is indeed an LF as long as the estimated rotor resistance is correct within 100%. By insuring this condition, all of the signals in the system will remain bounded [26].

$$V(w) = w^T \left( \frac{1}{2} P_1 + 1 P_2 + P_3 + P_4 \right) w$$

$$\dot{V}(w) = -w^T Q(\varepsilon_1, \varepsilon_2) w$$

where $\varepsilon_1 = \frac{R_r \beta^2 - \hat{R}_r \hat{v}_2}{2 R_r \beta^2}$ and $\varepsilon_2 = \frac{\hat{R}_r \hat{v}_1}{2 R_r \beta^2}$

### 3.2.2 Indirect field-oriented control parameter sensitivity

When IFOC is used with current control or with SVPWM, it is dependent on the rotor leakage inductance, $L_{lr}$, magnetizing inductance, $L_m$, and the rotor resistance, $r_r$. Again, it is possible to build a Jacobian matrix, $J_{IFO C}$, in which the sensitivities of torque, speed, and other desired variables or outputs are estimated relative to change in motor parameters. For IFOC, the Jacobian matrix is expected to be (67).

$$\frac{\Delta T^e}{\Delta \omega_{rm}} = J_{IFO C} \begin{bmatrix} \Delta L_{lr} \\ \Delta L_m \\ \Delta r_r \end{bmatrix}$$

where $J_{IFO C} = \begin{bmatrix} \frac{\partial T^e}{\partial L_{lr}} & \frac{\partial T^e}{\partial L_m} & \frac{\partial T^e}{\partial r_r} \\ \frac{\partial \omega_{rm}}{\partial L_{lr}} & \frac{\partial \omega_{rm}}{\partial L_m} & \frac{\partial \omega_{rm}}{\partial r_r} \end{bmatrix}$

It is important here to consider torque and speed ripple under switching control for both IFOC and DTC. While the sensitivity analyses would result in steady-state variations $\Delta T^e$ and $\Delta \omega_{rm}$, dynamic variations can also result from switching. For
example, for a given stator current under hysteretic switching, the formulations of the above Jacobian matrices are not trivial. If the stator current $i_s$ is given by $i_s = I_s + \Delta i_s$, where $\Delta i_s$ is the width of the hysteresis band and $I_s$ is the desired stator current, then the expected $T^e$ and $\omega_{rm}$ are given by (68) and (69), respectively. They are both broken into two terms: the offset, or average component, and the hysteresis band component.

$$T^e = T_{(offset)}^e + \Delta T_{(hys)}^e$$  \hspace{1cm} (68) \\
$$\omega_{rm} = \omega_{rm(\text{offset})} + \Delta \omega_{rm(hys)}$$ \hspace{1cm} (69) \\

Denoting the time average of a variable $x$ as $\langle x \rangle$, the resulting averages would be those in (3.70) and (3.71).

$$\langle T^e \rangle = \langle T_{(offset)}^e \rangle + \langle \Delta T_{(hys)}^e \rangle$$  \hspace{1cm} (70) \\
$$\langle \omega_{rm} \rangle = \langle \omega_{rm(\text{offset})} \rangle + \langle \Delta \omega_{rm(hys)} \rangle$$ \hspace{1cm} (71) \\

An offset will not occur if $\langle \Delta T_{(hys)}^e \rangle$ and $\langle \Delta \omega_{rm(hys)} \rangle$ are zero, but zero-average ripple is not guaranteed in general and must rely on integral gain in the loop controls.

3.3 Feedback Linearization: Input-Output Decoupling Control

In this section, global stability will be proved for feedback linearization input-output decoupled control. A parameter sensitivity analysis will also be carried out for this motor controller.

3.3.1 Feedback linearization: input-output decoupling control stability analysis

As in the other motor control schemes, it is still required that the Lyapunov function (LF) be positive definite and its derivative negative definite. In this case, the
candidate LF is given in (72) from [27]. This particular LF is positive definite by observation.

\[ V(e, \hat{T}_L) = \frac{1}{2} e^2 + \frac{1}{2\gamma} \hat{T}_L^2 \]

where \( e = (\omega - \omega_{\text{ref}}) - (\omega_{\text{M}} - \omega_{\text{M,ref}}) \), \( \gamma > 0 \in \mathbb{R} \)

and \( \hat{T}_L \) is the observed load torque

The derivative of the above candidate LF is given by (3.73).

\[ \dot{V}(e, \hat{T}_L) = -ke^2 + \hat{T}_L \left( -\frac{e}{J} - \frac{1}{\gamma} \frac{d}{dt} \hat{T}_L \right) \]

where \( k > 0 \in \mathbb{R} \)

Then from the torque adaptation law given in (74) from [27], the last two terms cancel out and (75) is the resulting derivative. Since \( k \) is positive, the derivative is always negative and therefore negative definite. Since the LF is positive definite and its derivative is negative definite, this means that the candidate LF is indeed an LF. This shows that the error between the speed error and the model’s speed error goes to zero as time goes to infinity. The same goes for the observed load torque. This shows that the controller is stable since the load torque is always a bounded real number.

\[ \frac{d}{dt} \hat{T}_L = -\frac{e}{J} \]

\[ \dot{V}(e, \hat{T}_L) = -ke^2 \]

3.3.2 Feedback linearization: input-output decoupling control parameter sensitivity analysis

Of all of the control schemes, FL-IOL is the most sensitive to parameter change. FL-IOL is sensitive to errors in rotor and stator resistances, magnetizing inductance, and
rotor and stator self inductances. This is demonstrated in the Jacobian shown in (76).

The reason for this is that in order to totally decouple the input and output, all parameters must be known accurately; if not, the scheme does not work correctly.

\[
\begin{pmatrix}
\Delta T^e \\
\Delta \omega_{rm}
\end{pmatrix} = J_{FL-IOL} \begin{pmatrix}
\Delta r_r \\
\Delta r_s \\
\Delta L_m \\
\Delta L_r \\
\Delta L_s
\end{pmatrix}
\]  

(76)

where \( J_{FL-IOL} = \begin{pmatrix}
\frac{\partial T^e}{\partial r_r} & \frac{\partial T^e}{\partial r_s} & \frac{\partial T^e}{\partial L_m} & \frac{\partial T^e}{\partial L_r} & \frac{\partial T^e}{\partial L_s} \\
\frac{\partial \omega_{rm}}{\partial r_r} & \frac{\partial \omega_{rm}}{\partial r_s} & \frac{\partial \omega_{rm}}{\partial L_m} & \frac{\partial \omega_{rm}}{\partial L_r} & \frac{\partial \omega_{rm}}{\partial L_s}
\end{pmatrix}\)

3.4 Vectorized Volts-Per-Hertz

In this section, a parameter sensitivity analysis will be conducted for vectorized volts-per-hertz control. Since vectorized volts-per-hertz control is inherently an open loop control drive, there are no stability issues associated with it.

3.4.1 Vectorized volts-per-hertz control parameter sensitivity analysis

Assuming that the direct current, \( i_d^* \), and the electrical rotor speed, \( \omega_e^* \), are commanded and not the direct voltage, \( v_d^* \), and the quadrature voltage, \( v_q^* \), then the vectorized volts-per-hertz drive is sensitive to the stator resistance and the stator leakage inductance, as shown in (77). If commanded voltages are used, then the drive is not sensitive to any parameters.
\[
\begin{pmatrix}
\Delta T^e \\
\Delta \omega_{rm}
\end{pmatrix} = J_{\text{Vectorized } V} \begin{pmatrix}
\Delta r_s \\
\Delta L_{ls}
\end{pmatrix}
\]

where
\[
J_{\text{Vectorized } V} = \begin{pmatrix}
\frac{\partial T^e}{\partial r_s} & \frac{\partial T^e}{\partial L_{ls}} \\
\frac{\partial \omega_{rm}}{\partial r_s} & \frac{\partial \omega_{rm}}{\partial L_{ls}}
\end{pmatrix}
\] (77)

3.5 Load Modeling

There are many different types of loads to consider when working with an induction machine. For instance, there are friction, bearings, and windage for internal motor loads, and devices such as pumps, fans, compressors, conveyors, hoists, rolling mills, coil systems, take-up systems, and traction applications like a subway train or an electric vehicle for external loads. All of these loads can be represented by the model in (78).

\[
T_{\text{load}} = f(\omega, \theta, t) = \sum T^e - J \frac{d\omega}{dt}
\]

\[
= k_0 + k_1\omega + k_2\omega^2 + k_3\omega^3 + k_4\int \omega dt - J \frac{d\omega}{dt}
\] (78)

Hoists are represented by constant forces, while friction and bearings look like linear functions of rotational speed. A load that is represented by a second order function of the rotational speed is windage, and one that looks like a constant power load is a take-up system. For the purpose of this work, only a second order polynomial with known parameters is used, such as a pump or fan. The load model used in the simulations that follow is exhibited in Equation (79).

\[
T_{\text{load}} = \left(1.82e^{-4}\right)\omega^2 + \left(1.82e^{-2}\right)\omega
\] (79)
Friction merits some special attention since it is very important to almost all of the above loads. Although friction is typically modeled by a linear function, this is a first order approximation. Depending on the amount of friction, this model might be very inaccurate. At no speed, there is a high amount of torque required to get the object in motion. Once it is in motion, the object accelerates quickly, looking much like an exponential function for a small segment of the torque function until it levels off to a linear non-zero sloped function. Figure 3 shows the typical friction force phenomenon, where $F_{brk}$ is the breakaway friction, $F_c$ is the Coulomb friction, $F_s$ is the static friction and $F_v$ is the viscous friction [28].

![Figure 3. Friction Force Phenomenon [28]](image)

3.6 Flux Observer Design

There are numerous examples of observers for flux estimation in the literature. One particular paper, [29], stands out when looking for a survey of observers for
induction machine control. Verghese and Sanders discuss existing methods for flux observers and come up with some innovative insights including an observer that uses predictive flux error to improve the observer. The resulting observer looks like that in (80). Although this observer is shown in [29] to converge very quickly, it requires more computation than a simpler but less accurate observer (81). The observer in (81) was used in all of the simulations that follow in Chapter 4 because of its simplicity and prevalence. It comes directly from the machine equations in (13).

\[
\dot{\lambda}_r = \left[ \begin{pmatrix} -1/T_r & I + \omega J \\ \frac{1}{T_r} & 0 \end{pmatrix} \right] \hat{\lambda}_r + \left( \frac{1}{T_r} \right) M_i + K (\hat{v}_s - v_s) \quad (80)
\]

where \( K \) is a \( 2 \times 2 \) matrix of observer gains.

\[
\frac{d\hat{\lambda}_{qs}}{dt} = v_{qs} - R_s i_{qs} - \omega \hat{\lambda}_{ds} \\
\frac{d\hat{\lambda}_{ds}}{dt} = v_{ds} - R_s i_{ds} - \omega \hat{\lambda}_{qs} \quad (81)
\]
4. COMPUTER SIMULATION

In this chapter, system modeling and design process are discussed. A detailed examination of how each of the motor drive simulations was performed is presented.

4.1 Simulation Modeling

A simulation should be completed with parameters that are as accurate as desired in order to better predict what will happen in any real-life system. The reason for simulating is that it is relatively cheap compared to the equivalent design in hardware. The costs behind simulation are the software package, in this case MATLAB-Simulink, and the man-hours needed to build, test, and debug a design. Typically this is far cheaper than going straight to fabricating a design, physically testing, and then physically debugging. In many cases, hardware debugging might not even be feasible due to major errors in the design that could be found in software.

In the actual simulations themselves, as in much of engineering, tradeoffs must be made. In this case, the tradeoffs are between model accuracy, and simulation time. If the application requires a detailed analysis of the system, the model should contain as many parameters as possible, parameters should be known very accurately, time-steps should be short, and full, not approximate models, should be used. The other side of this is a quick simulation, in which some parameters are not well known or neglected, time-steps are longer, and approximate models are used. The application, and not the designer, should determine the type of simulation to be done. For instance, in many hybrid vehicle motor drive applications, the input from the user is a torque function, which need not be extremely accurate. This is in stark contrast to an application that requires extreme precision, such as robotics, or a precise industrial application that needs
to track speed or position exactly. In the former, while a very precise simulation is alright to do, an imprecise simulation is frequently used to speed up the design process. In the latter, an extremely accurate simulation must be completed in order to insure that a drive is able to achieve “high performance” status.

There is also a common design process that is used in practice. A flow chart similar to Figure 4 is typically employed when creating a new process or design [30]. This is known as a “design by iterative process” [30]. It consists of a loop between desired performance parameters and the physical attributes via an engineer’s design and analysis. Usually there are given performance parameters, from which theory is used to create a design. This design is then created. It is then tested and the results analyzed. From the analysis, the performance parameters are changed and/or a new design is conceived. Sometimes this process is switched around wherein there are given or desired physical attributes and resulting performance parameters are found, but this is more uncommon.

![Figure 4. Design and Analysis Flow Chart](image)

From the design and analysis process previously described a system, theory, or machine is created. This result can either be in the physical or simulation form. As discussed earlier, the simulation is used as a way to inexpensively and quickly run
through this iterative process. In the next few sections, the four main motor drives’ block diagrams will be presented, along with comparative results and analysis.

4.2 Simulation of Direct Torque Control

![Figure 5. DTC Block Diagram](image)

The block diagram for a hysteretic DTC motor drive is shown in Figure 5. The typical use of DTC in an industrial setting, where a motor is connected to the electric grid via an inverter and rectifier pair. The ac/dc block in Figure 5 stands for the rectifier, while the dc/ac block represents the inverter. Between the two is a dc link which can vary from a few volts to well into the kV range. The induction motor in the block diagram is represented by the labeled circle. As described earlier, the inputs from the user in this motor drive are the electrical torque, $T^*_e$, and the stator flux, $\lambda^*_s$, which are given by equations (12) and (82), respectively.
They are compared against the calculated torque and stator flux, respectively. The difference, or error, is sent through the hysteresis block for each signal. The output from these blocks is a -1, 0, or 1, where -1 represents a negative error, 0 no error, and 1 a positive error. In practice, it is very unlikely that there will be no error, so the 0 output is neglected. The output for both the torque and the flux signals is sent into the switching table, which decides what gate signals should be set to the inverter by exploiting a simple look-up table. The other input to the look-up table is the stator flux angle, $\rho_s$, given by Equation (83).

$$\rho_s = \tan^{-1}\left(\frac{\lambda_{qs}}{\lambda_{ds}}\right)$$  \hspace{1cm} (83)

The other important signals are the voltage and current measurements taken from the motor. Combined, the stator current and voltages are transformed into the stator qd0 reference frame, and used to create the stator flux magnitude estimate. The stator flux is then used along with the transformed currents to come up with the torque estimate. These two estimates, $\hat{T}_s$ and $\hat{\lambda}_s$, are then compared against the commanded value given by the user.

The block diagram for a DTC motor drive using the SVPWM switching technique is shown in Figure 6. One can see that this block diagram is slightly different from that in Figure 3; the drive that uses SVPWM is distinct in the fact that it uses the quadrature and direct voltages created by two PID loops in addition to a stator flux position estimator to determine the correct pulses to be applied to the inverter.
4.2.1 Direct torque control parameter sensitivity results

Now that the block diagrams for the simulation of DTC have been discussed, some simulation parameter sensitivity results can be shown. From the sensitivity analysis in Chapter 3, the only parameter that shows any sensitivity to changes in normal DTC that uses a hysteretic switching table is stator resistance. To run the parameter sensitivity analysis, the resistance was increased and decreased by 25% to see just how sensitive the motor controller is to the stator resistance. These two trial runs are then compared against the case when the stator resistance is unaltered. When looking at the results found via simulation in Figure 7, it is shown that while the torque response of the motor does change with large variations in stator resistance, it is barely noticeable. This was a little unexpected since it was anticipated that the increase in stator temperature would increase the resistance and thus cause a bigger variation in torque performance.
After examining the parameter sensitivity in standard DTC, it is desired to see the performance of the motor controller with a completely different switching scheme, such as SVPWM. This would allow one to see if the parameter sensitivity was potentially caused by the switching scheme that was chosen or the drive itself. The torque command that is used is 5 N-m. The motor parameters that were used in the simulation are listed in Appendix A. Figure 8 shows the DTC-SVPWM results using a decrease of 25% in the stator resistance and the unaltered case. A totally different response results when the stator resistance is just 10% greater than thought to be; it becomes unstable and can be seen in Figure 9. Its use in the flux estimator seems to affect the stability of the system as a whole. From the Jacobian in Chapter 3, it was shown that DTC-SVPWM had sensitivity to the rotor resistance.
Figure 8. Sensitivity to $r_s$, Negative Deviation from Nominal [31]

Figure 9. Sensitivity to $r_s$, Negative Deviation from Nominal [31]

Figure 10 shows that when the rotor resistance is increased and decreased by 25%, there is barely any change in the torque response. This leads one to think that the partial with respect to the rotor resistance is very small, and can be considered to be insensitive to that parameter.
4.2.2 DTC performance with changing sector count

Now that the parameter sensitivity analysis has been performed above, another concept was tested via simulation: changing the number of sectors that the flux position could be divided into. The typical configuration of DTC uses six sectors to determine the flux position, while it is proven in [13] that the minimum number of sectors is five. The six flux sector DTC is typically used because it makes the waveforms all be offset from one another by 120° and also makes the waveforms a balanced three phase set. Because of this fact, any multiple of six gives a good quality waveform.

For acquiring results, the standard hysteretic DTC topology was used to compare 6, 12 and 256 sector DTC. The only variable that changed in this experiment was the number of stator flux sectors. The results of this experiment are found in Figures 11 and 12. When examining the results, it is apparent that there is a significant performance increase in the ripple when transitioning from 6 to 12 sectors. By visual inspection, it
appears that there is not much of an improvement in the ripple of the 256 sector trial when contrasted to the 12 sector trial. The maximum ripple of the 6 sector DTC is 0.52 N-m when 5 N-m is the commanded torque. This compares to a maximum ripple of 0.28 N-m using 12 sectors, and 0.24 N-m with 256 sectors. The resultant speed plot is that of Figure 9. In this figure, there is not much of a difference between all of the trials because they all have similar average torques. By reducing the ripple, the stresses seen on the motor over time will decrease. This can potentially lead to a longer motor lifetime and higher reliability for the overall solution. Looking at these results, it is therefore natural to conclude that a higher sector count is useful in the reduction of torque ripple in a standard DTC motor drive.

Figure 11. Torque Performance Between 6, 12, and 256 Sector DTC
It is useful to now compare the dynamics of DTC using the 6, 12, and 256 stator flux sectors. Shown in Figure 13 is the start-up performance comparing all three sector counts when the initial flux is 0.1 V-s. Looking at this figure, it is hard to determine which sector count is “best.” This is because for different levels of torque demanded, there are different optimum sector counts. For instance, if 1.5 N-m of torque was commanded, then DTC with 6 sectors would arrive first in 1.97 ms and 12 and 256 sector arriving 0.22 ms later. If 3 N-m was commanded as shown in Figure 10, then 12 sector DTC is the best performing, with 256 very close in performance and 6 sector lagging behind. When using 0.001 V-s as the initial flux, the results are slightly different. There is a period of time where each sector count is superior to the others, as shown in Figure 14. For higher torque values—greater than 1.13 N-m—256 sector DTC is best performing.
Figure 13. Start-Up Torque Performance Between 6, 12, and 256 Sector DTC, With Initial Flux of 0.1 V-s

Figure 14. Start-Up Torque Performance Between 6, 12, and 256 Sector DTC, With Initial Flux of 0.001 V-s
There is also a very large difference between performance levels within the same sector count for DTC drives using hysteretic control. The gains used in Equation (57) alter the performance of the DTC drive greatly for 12 sector and 256 sector DTC, but not for 6 sector DTC. Figure 15 shows the performance of 12 sector DTC with two different gain conditions: “worst condition”: \( k_\phi = -100 \) and \( k_\tau = -300 \), and “best condition”: \( k_\phi = -300 \) and \( k_\tau = -300 \). Figure 16 shows that there are different performances using different gains for 256 sector DTC as well. In this case, the gain conditions are as follows: “worst condition”: \( k_\phi = -510 \) and \( k_\tau = -300 \), and “best condition”: \( k_\phi = -300 \) and \( k_\tau = -300 \). The 256 sector case is slightly different since the trial called “worst” condition is actually better for a range of torque values from 0 to 0.65 N-m. Therefore, it is very important to have the correct gains in place when using a higher number of flux sectors than six.

Figure 15. Start-Up Torque Performance for 12 Sector DTC- Different Gains
4.2.3 DTC performance analysis

In this section, an in-depth performance analysis will be performed and discussed. One of the goals of this research is to push the envelope and see what kind of performance can be achieved. Another goal is to see what would happen if there was an error in the physical measurement of signals such as voltages and currents. Table 4 shows the performance of a standard DTC drive using hysteretic control when there is an error in the direct or quadrature voltage when the initial flux is set to be only 0.001 V-s. The time that it takes for the torque and the flux to reach steady state is recorded in Table 4. The commanded torque and flux in this section are set to be 4 N-m and 0.52 V-s, respectively. The nominal steady-state times to compare against without any error are 4.12 ms for torque and 4.50 ms for flux. This table shows that the torque response cannot
be increased, but the flux response can be. There is a tradeoff between having a better flux response and a worse torque response for a positive error in $V_{ds}$. The actual force that moves the motor is the torque, so it is natural to optimize around not letting the torque response’s performance decline. A decrease in $V_{ds}$ will not cause any change in torque performance, but the time that it takes the flux to reach steady-state becomes longer so this is an undesirable condition. For an increase or decrease in $V_{qs}$, the torque response does not change. It is possible to have the flux arrive in steady state quicker—4.25 ms versus 4.5 ms—with a positive error of 25%. The opposite is true for a decrease in $V_{qs}$—the time it takes to get to steady-state for the flux increases to 4.67 ms.

**Table 4. DTC Performance, Initial $\phi_s=0.001$ V-s, Incorrect Voltage Measurements**

<table>
<thead>
<tr>
<th>Change in Gains (increase/decrease)</th>
<th>Times to reach steady state (ms)</th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Torque</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Increase</td>
<td>Decrease</td>
<td>Increase</td>
<td>Decrease</td>
<td></td>
</tr>
<tr>
<td>$V_{ds}$</td>
<td>5%</td>
<td>4.12</td>
<td>4.12</td>
<td>4.20</td>
<td>4.85</td>
</tr>
<tr>
<td></td>
<td>10%</td>
<td>4.17</td>
<td>4.12</td>
<td>4.00</td>
<td>5.10</td>
</tr>
<tr>
<td></td>
<td>15%</td>
<td>4.20</td>
<td>4.12</td>
<td>3.70</td>
<td>5.80</td>
</tr>
<tr>
<td></td>
<td>25%</td>
<td>4.24</td>
<td>4.12</td>
<td>3.50</td>
<td>unstable</td>
</tr>
<tr>
<td>$V_{qs}$</td>
<td>5%</td>
<td>4.12</td>
<td>4.12</td>
<td>4.43</td>
<td>4.50</td>
</tr>
<tr>
<td></td>
<td>10%</td>
<td>4.12</td>
<td>4.12</td>
<td>4.42</td>
<td>4.50</td>
</tr>
<tr>
<td></td>
<td>15%</td>
<td>4.12</td>
<td>4.12</td>
<td>4.40</td>
<td>4.60</td>
</tr>
<tr>
<td></td>
<td>25%</td>
<td>4.12</td>
<td>4.12</td>
<td>4.25</td>
<td>4.67</td>
</tr>
</tbody>
</table>

Similar to the data displayed in Table 4, the data displayed in Table 5 shows the performance of DTC, but now with a larger initial flux value of 0.1 V-s. The nominal time to reach steady-state without any error in voltage measurements is 2.64 ms for the torque, and 5 ms for the flux. Because of the initial flux of 0.1 V-s, the torque could reach steady-state much quicker than in the case when the flux was initialized at 0.001 V-s. Table 4 shows that varying the direct and quadrature voltages does nothing to affect
the torque performance except in the case of increasing $V_{qs}$. In this case, there is a degradation of performance as the error in quadrature voltage increases. This also causes the flux to perform better; for instance, if there is an increase in quadrature voltage of 50%, then the flux reaches steady state in 2.37 ms, compared to 5 ms in the nominal case. This is a decrease of 52.6% from the projected value. What actually moves the motor is torque, so this increase in flux performance actually hinders the motor controller’s response.

**Table 5. DTC Performance, Initial $\phi_i=0.1$ V-s, Incorrect Voltage Measurements**

<table>
<thead>
<tr>
<th></th>
<th>Flux (ms)</th>
<th>Torque (ms)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Increasing $V_{ds}$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>50%</td>
<td>5.00</td>
<td>2.64</td>
</tr>
<tr>
<td>100%</td>
<td>4.95</td>
<td>2.64</td>
</tr>
<tr>
<td>150%</td>
<td>4.90</td>
<td>2.64</td>
</tr>
<tr>
<td>200%</td>
<td>4.80</td>
<td>2.64</td>
</tr>
<tr>
<td>Decreasing $V_{ds}$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>25%</td>
<td>4.90</td>
<td>2.64</td>
</tr>
<tr>
<td>50%</td>
<td>4.80</td>
<td>2.64</td>
</tr>
<tr>
<td>75%</td>
<td>4.55</td>
<td>2.64</td>
</tr>
<tr>
<td>Increasing $V_{qs}$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>50%</td>
<td>2.37</td>
<td>2.95</td>
</tr>
<tr>
<td>100%</td>
<td>1.75</td>
<td>3.45</td>
</tr>
<tr>
<td>150%</td>
<td>1.29</td>
<td>3.87</td>
</tr>
<tr>
<td>200%</td>
<td>1.07</td>
<td>4.00</td>
</tr>
<tr>
<td>Decreasing $V_{qs}$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>25%</td>
<td>unstable</td>
<td>2.64</td>
</tr>
<tr>
<td>50%</td>
<td>unstable</td>
<td>2.64</td>
</tr>
<tr>
<td>75%</td>
<td>unstable</td>
<td>2.64</td>
</tr>
</tbody>
</table>

While Tables 4 and 5 showed the motor drive performance for errors in voltages, Table 6 shows the time that it takes to reach steady-state when there is an error in the current measurements. It can be seen that if there is an error in current measurement, there is no change in the torque performance. It can also be seen that if either the quadrature or direct current has measurement errors that decrease its value, the time it takes to reach steady-state goes down. This can potentially be useful when actively
trying to decrease this time—intentionally adding a gain less than 1 to the current measurements to increase the flux dynamic performance. Again the opposite is true—a positive error will cause a decrease in dynamic flux performance.

Table 6. DTC Performance, Initial $\phi_0 = 0.001$ V-s, Incorrect Current Measurements

<table>
<thead>
<tr>
<th>Change in Gains (increase/decrease)</th>
<th>Times to reach steady state (ms)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Torque</td>
</tr>
<tr>
<td></td>
<td>Increase</td>
</tr>
<tr>
<td>Ids</td>
<td>5%</td>
</tr>
<tr>
<td></td>
<td>10%</td>
</tr>
<tr>
<td></td>
<td>15%</td>
</tr>
<tr>
<td></td>
<td>25%</td>
</tr>
<tr>
<td>Iqs</td>
<td>5%</td>
</tr>
<tr>
<td></td>
<td>10%</td>
</tr>
<tr>
<td></td>
<td>15%</td>
</tr>
<tr>
<td></td>
<td>25%</td>
</tr>
</tbody>
</table>

The DTC performance is shown for different errant values of measured current in Table 7. The difference between Table 6 and Table 7 is again that an initial flux value was given to be 0.1 V-s instead of 0.001 V-s. Just as in the previous case, the torque response does not change with current. The flux performance increases with increasing direct current error, and also with decreasing quadrature current. Increasing the error in the quadrature current too much causes the flux to not come to steady state. It should be noted, just as in Table 5, that the torque reaches steady-state in 2.64 ms, versus 4.12 ms in Tables 4 and 6. Some other more unrealistic trials were taken, and are therefore not going to be discussed, but are in Appendix A.
Table 7. DTC Performance, Initial $\phi_s = 0.1$ V-s, Error in Current Measurements

<table>
<thead>
<tr>
<th></th>
<th>Flux (ms)</th>
<th>Torque (ms)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Increasing $I_d$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>50%</td>
<td>4.85</td>
<td>2.64</td>
</tr>
<tr>
<td>100%</td>
<td>4.60</td>
<td>2.64</td>
</tr>
<tr>
<td>150%</td>
<td>4.20</td>
<td>2.64</td>
</tr>
<tr>
<td>200%</td>
<td>3.80</td>
<td>2.64</td>
</tr>
<tr>
<td>Decreasing $I_d$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>25%</td>
<td>5.00</td>
<td>2.64</td>
</tr>
<tr>
<td>50%</td>
<td>5.00</td>
<td>2.64</td>
</tr>
<tr>
<td>75%</td>
<td>5.00</td>
<td>2.64</td>
</tr>
<tr>
<td>Increasing $I_q$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>50%</td>
<td>unstable</td>
<td>2.64</td>
</tr>
<tr>
<td>100%</td>
<td>unstable</td>
<td>2.64</td>
</tr>
<tr>
<td>150%</td>
<td>unstable</td>
<td>2.64</td>
</tr>
<tr>
<td>200%</td>
<td>unstable</td>
<td>2.64</td>
</tr>
<tr>
<td>Decreasing $I_q$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>25%</td>
<td>4.00</td>
<td>2.64</td>
</tr>
<tr>
<td>50%</td>
<td>3.50</td>
<td>2.64</td>
</tr>
<tr>
<td>75%</td>
<td>3.20</td>
<td>2.64</td>
</tr>
</tbody>
</table>

4.3 Simulation of Indirect Field-Oriented Control

The block diagram for an IFOC motor drive with current hysteresis as the switching scheme is shown in Figure 17. This combination is by far the most common higher-performance drive used in industry. The commanded signals are the torque, $T^*$, and direct axis rotor flux, $\lambda_{dr}^*$, which differs from the DTC motor controller that uses the absolute value of the total stator flux. The torque and rotor flux commands are converted into the quadrature and direct stator current variables and then compared to the measured induction motor currents that are fed back. The induction motor in the block diagram is represented by the circle with the label “IM”.

63
The second switching scheme that is analyzed with IFOC is SVPWM. The block diagram for this topology can be found in Figure 18. The difference between this topology and the previous current hysteretic IFOC topology is that the quadrature and direct voltages are used in the switching scheme instead of the currents. To get these voltages, the equations labeled “1” and “2” in Figure 18 are shown in (84) and (85) [31].

\[
v_{qs}^* = \sigma \left( -\frac{P}{2} \omega_l i_{ds} \frac{L_m}{\tau_r} \left( i_{qs}^r \lambda_{dr}^r - i_{ds}^r \lambda_{qr}^r \right) \right) - \frac{L_m P \omega_l \lambda_{dr}^r}{2L_r} \tag{84}
\]

\[
v_{ds}^* = \sigma \left( -\frac{P}{2} \omega_l i_{qs} \frac{L_m}{\tau_r} \left( i_{ds}^r \lambda_{dr}^r - i_{qs}^r \lambda_{qr}^r \right) \right) - \frac{L_m P \omega_l \lambda_{dr}^r}{2L_r} \tag{85}
\]
Figure 18. Block Diagram of IFOC-SVPWM [31]

Now that the standard topologies have been discussed, the parameter sensitivity results will be revealed. Standard IFOC is sensitive to the rotor leakage inductance, $L_{lr}$, magnetizing inductance, $L_m$, and rotor resistance, $r_r$, as discussed in Chapter 3. Some of the results compiled here have been previously published in [31]. By visual inspection of Figure 19, one can see that by increasing or decreasing the rotor leakage inductance by 25% from the nominal value, there is minimal change in the torque response. Figure 20 shows a totally different story: when the magnetizing inductance is decreased by 25% from its nominal value, the steady state average torque output increases to 6.93 N-m, or a 38.6% increase over the desired torque command of 5 N-m. When the magnetizing inductance is increased by 25%, the steady state decreases to 4.37 N-m, which is a 12.6% decrease from the desired torque. The last parameter that has any sensitivity to change is the rotor resistance. Figure 21 shows the change in performance of standard
IFOC when the rotor resistance is increased and decreased 25% from the nominal value. Increasing the rotor resistance by 25% will affect the rotor time constant and therefore increase the torque response so that there is an overshoot, while a decrease of 25% leads to a torque response that is overdamped and therefore an undershoot occurs. There are also steady-state torque errors for both cases, which is a very undesirable effect. In conclusion, the only two parameters that seem to have a high sensitivity for standard IFOC are the magnetizing inductance and the rotor resistance.

Figure 19. IFOC, Sensitivity to $L_r$ [31]
Figure 20. IFOC, Sensitivity to $L_m$ [31]

Figure 21. IFOC, Sensitivity to $r_r$ [31]
Now that standard IFOC has been examined, it is useful to look at the parameter sensitivity of IFOC but with a different switching scheme: SVPWM. As shown in Chapter 3, IFOC-SVPWM is sensitive to the change in rotor leakage inductance, $L_{lr}$, rotor self inductance, $L_r$, magnetizing inductance, $L_m$, and rotor resistance, $r_r$. The sensitivity from a change in the rotor leakage inductance is shown in Figure 22. The sensitivity from a 25% decrease or increase in the rotor leakage inductance is very low, since the altered parameter performance is almost identical to the nominal performance. As for the sensitivity for the magnetizing inductance, when the error is 25% greater than the nominal value, the torque gets to steady-state quicker, but there is a steady-state error where it is 0.23 N-m lower than commanded (Figure 23). The case where the error is 25% lower than the nominal value is almost identical to the nominal case, as can be seen in Figure 23.

Figure 22. IFOC-SVPWM, Sensitivity to $L_{lr}$ [31]
Another parameter that was analyzed in Chapter 3 for a sensitivity analysis was the rotor inductance. Figure 24 shows that the torque performance is very insensitive to any changes in the rotor inductance—the altered cases are virtually identical to the unaltered case. The last parameter that was tested in the IFOC-SVPWM parameter sensitivity analysis is the rotor resistance. Figure 25 shows the sensitivity results for this parameter. There seems to be a very low sensitivity to any change in the rotor resistance, which was slightly unexpected. Therefore, the only parameter for IFOC-SVPWM that seems to affect performance is the magnetizing inductance.

Figure 23. IFOC-SVPWM, Sensitivity to $L_m$ [31]
4.4 Simulation of Vectorized Volts-per-Hertz

The next motor controller to examine is vectorized volts-per-hertz. The block diagram can be seen in Figure 26. The unique aspect of this controller is the fact that
there is no inherent feedback—it is an open loop controller. The block diagram shows
the motor speed being fed back, but this is not required. As discussed previously, this
has positive and negative implications, the positive being that it is a simple controller
and has low parameter sensitivity, and the negative that it is that it is not typically known
as a “high performance” drive because of its simple model. A commanded current is
then converted into a commanded voltage and an inverter switching scheme is then
chosen to give the gate drive pulses.

Figure 26. Block Diagram of Vectorized Volts-per-Hertz

Now that the block diagram for the simulation has been discussed, the parameter
sensitivity results will be revealed. As discussed in Chapter 3, the vectorized volts-per-
hertz drive is sensitive to changes in stator resistance and stator leakage inductance.
Figure 27 shows the speed results for the altered and unaltered stator resistance, while
Figure 28 shows the corresponding torque results. Looking at these figures, one can see
that the drive is slightly sensitive to changes in the stator resistance. A positive error of 25% in the stator resistance causes an increase of 0.67 N-m, or 12.4%, in the peak torque. A negative error of 25% causes a 0.44 N-m decrease, or 8.2%, in peak torque.

Figure 27. Vectorized V/Hz, Speed Parameter Sensitivity to $R_s$

Figure 28. Vectorized V/Hz, Torque Parameter Sensitivity to $R_s$
This is not the case for the stator leakage inductance; Figure 29 shows the speed changes resulting from the change in stator leakage inductance. The speed’s matching torque response is shown in Figure 30. With a 25% positive error in the stator leakage inductance, the speed is up to 4 rad/s above the predicted value. A similar phenomenon occurs with a 25% negative error, where the speed can be 5 rad/s slower than forecasted. When exploring the sensitivities for vectorized volts-per-hertz as a whole, the drive seems to be sensitive to parameter changes in the stator leakage inductance and the stator resistance.

Figure 29. Vectorized V/Hz, Speed Parameter Sensitivity to L_{ls}
4.5 Simulation of Feedback Linearization

Totally unlike vectorized volts-per-hertz control, feedback linearization uses the feedback of many variables for its control algorithm. The variables that are fed back are: stator currents, stator voltages, and the speed, as shown in Figure 31. From these, the rotor flux is estimated. With the rotor flux and the stator currents, the states $\dot{\Psi}_r$ and $\dot{\omega}_r$ along with their integrals $\Psi_r$ and $\omega_r$ are sent to flux and speed controllers. From the flux and speed controllers, the desired voltages are sent to the switching scheme of choice. As before, the switching scheme creates the gate pulses that will control the induction motor.
From Chapter 3, the parameters that when change influence the feedback linearization motor drive are the rotor leakage inductance, magnetizing inductance, rotor self inductance, and the rotor resistance. The parameter sensitivity results for the leakage rotor inductance are shown in Figures 32 and 33. Figure 32 shows the torque response for changes in the parameter, while Figure 33 shows the resulting speed plot. One can see that the rotor leakage inductance has a low sensitivity to deviations.
Figure 32. FB-Linearization, Torque Parameter Sensitivity to $L_{lr}$

Figure 33. FB-Linearization, Speed Parameter Sensitivity to $L_{lr}$

Figure 34 shows the results for when the magnetizing inductance is altered by +/- 25%, and Figure 35 is the resulting speed plot. Having a positive error of 25% causes a 12% increase of peak torque in addition to a slight increase in time required to get to steady-state. The opposite is the case for when the magnetizing inductance is lower than
predicted by 25%; the peak torque is lowered by 11% and the steady-state time is reduced by 0.04 seconds, or 6.1%, from nominal conditions.

Figure 34. FB-Linearization, Torque Parameter Sensitivity to $L_m$

Figure 35. FB-Linearization, Speed Parameter Sensitivity to $L_m$
Altering the rotor self-inductance by 25% in a positive fashion causes the torque peak to be reduced by 9.1% and enter into steady-state 8.8% quicker than the nominal condition, as shown in Figure 36. The corresponding speed plot is given in Figure 37 that shows similar performance numbers. Again, the opposite is true—a 25% decrease in expected rotor inductance results in a poorer performance from this motor controller. This causes the peak torque to increase by 14.5%, and the time to steady-state to increase by 21.4%. Therefore, if one was estimating the rotor self-inductance, they would want to err on the positive side—overestimate the rotor self-inductance to achieve better performance.

![Figure 36. FB-Linearization, Torque Parameter Sensitivity to Lr](image-url)
The last parameter that needs to be examined for feedback linearization is the rotor resistance. This parameter is very sensitive to changes, as shown in Figure 38 for torque response, and Figure 39 for speed response. Altering the rotor resistance causes exactly the same response as changing the magnetizing inductance, because both parameters are contained in the feedback linearization flux observer and are in the same signal chain. This means that a 1% error in the same direction for either parameter’s value causes the exact same erroneous response. Therefore, a positive error in the knowledge of the rotor resistance causes an undesired effect, while a negative error yields an increase in performance, as in the parameter sensitivity of the magnetizing inductance case above.
In summary, magnetizing inductance, rotor self-inductance, and rotor resistance have high sensitivities, while rotor leakage inductance shows a low sensitivity to change. To increase performance of a feedback-linearization drive, one would want to make sure...
that if the parameter knowledge for the magnetizing inductance and rotor resistance was incorrect, it should be incorrect in an overestimated fashion. The opposite is true for the rotor self-inductance, which should be underestimated to achieve higher performance.

4.6 Comparison of Motor Drives

Now that all of the drives’ results have been individually analyzed, mainly using the parameter sensitivity tests, this chapter will conclude with tests that compare all of the drives to one another. The tests that were chosen are: speed step, torque step, position command, and Bode plots of the motor drive. The speed step is useful when trying to change from one speed to another. One potential use of this would be cruise-control on the highway in an HEV or EV. The next test is a torque step, which would be useful for determining which drives have a high torque response. Sticking with the automotive theme, one application could be the sheer acceleration of a HEV or EV from standstill, or when coming onto the highway. The third test is a position command, which tests the position control of a motor drive at very low speed. This is useful for slow applications like a factory floor conveyor belt, or trying to track an exact position. The last test, a Bode plot of each of the motor drives, shows the torque response of a drive to a varying torque input command. By going through these four tests, one can arrive at a possible conclusion as to which drive is better in a range of applications.

The first test that was run to compare the motor drives is the torque step test using an initial flux of 0.1 V-s. The results of this test can be seen in Figure 40. The feedback linearization input-output decoupled control and standard six-sector DTC drives have an almost instantaneous response—in the millisecond range—while current hysteretic IFOC and vectorized volts-per-hertz have a slower torque response. The plot
in Figure 41 shows the start-up and step response for the corresponding speed response of the drives. Using standard motor drive and switching topologies, DTC and FB-linearization clearly are the top performers.

![Figure 40. Torque Response of All Four Motor Drives to a Step Torque Command of 4 N-m to 3 N-m](image_url)
Figure 41. Speed Response of All Four Motor Drives to a Step Torque Command of 4 N-m to 3 N-m

In traditional current-hysteretic IFOC, the drive seems to be limited by the current PI controller. In [31], the authors discuss alternative switching topologies—such as SVPWM or a switching table—for IFOC. The filtered torque step response results of SVPWM DTC versus SVPWM IFOC are shown in Figure 42. The speed step response is shown in Figure 43. The dynamic performances of DTC and IFOC are both affected significantly by the choice of switching scheme. These PID gains were tuned for best performance for each of the drives, and gain values can be found in Appendix A.

IFOC-SVPWM performs much better than IFOC with hysteretic control; for instance, the torque overshoot is much less with SVPWM—less than 0.5%, compared to 14%—since the system is almost critically damped with the chosen gains under SVPWM. DTC-SVPM has a deterioration of performance—26% torque overshoot
compared to 0% with the standard six sector switching table. The worst case IFOC-SVPWM speed overshoot is 40.6%, while DTC-SVPWM has only a 9.4% speed overshoot. The torque settling time of IFOC-SVPWM in this case has noticeably reduced to about 0.3 s, and the DTC-SVPWM torque settling time has become a good deal longer—about 0.3 s compared to 15 ms in the case with the switching table. The speed settling times are also about 0.3 s for both DTC and SVPWM. Based on this more direct comparison using the same switching scheme, there is no clear distinction as to which method to choose for a better torque step performance. It is shown that when the drive and switching scheme are totally decoupled, the performances can be very dissimilar; IFOC-SVPWM performs much better than the classic IFOC using current hysteresis.

From these results, it looks as if IFOC performs slightly better than DTC using the same switching technique, in this case SVPWM. This does not mean it is a better drive overall, since there may be a different switching scheme that yields a different result. For instance, IFOC may turn out to be inferior to DTC when using a switching table as the preferred switching scheme. Therefore, the results are still inconclusive as to which of the four drives has the best torque response because of the many switching schemes that can be applied to each drive. The numerous permutations between motor drives and switching schemes do not allow for a timely and thorough analysis.
Figure 42. Torque Response of DTC-SVPWM and IFOC-SVPWM to a Driving Cycle [31]

Figure 43. Speed Response of DTC-SVPWM and IFOC-SVPWM to a Driving Cycle [31]

The next test, as mentioned earlier, is the speed step test. The results of this assessment are shown in Figures 44 and 45 for the speed and torque responses, respectively. The PID gains greatly affect the response of each drive, so they were
Figure 44. Speed Response of All Four Motor Drives to a Speed Command of 100 rad/s to 80 rad/s

Figure 45. Torque Response of All Four Motor Drives to a Speed Command of 100 rad/s to 80 rad/s
optimized for each drive individually. Looking at the speed drop from 100 rad/s to 80 rad/s, one can see that IFOC, DTC and FB linearization all performed similarly, while the outlier was vectorized volts-per-hertz control with a much inferior speed response. Steady-state in this work is defined as +/- 1% of the nominal value. IFOC, DTC and vectorized volts-per-hertz all had undershoots, while FB linearization had a slight overshoot. DTC seems to come out ahead in this trial when one looks at the speed response alone; if one also looks at DTC’s torque response, it might be a little unrealistic. The torque rating for this motor is around 12 N-m at the rated speed of 1740 RPM. The maximum torque is set by the maximum $\mathcal{J} \times \mathcal{B}$. Since $\mathcal{B}$ is a fixed value—usually around 1.5 Wb—this true torque maximum is set by the maximum current density $\mathcal{J}$. The current density can vary greatly, but can sustain peaks for a few milliseconds before the copper wire gets too hot and naturally lowers the current density. Therefore, the actual torque can be a few multiples higher than the “rated” value. When looking at Figure 45, DTC has a torque peak of 33.5 N-m, which may or may not be unrealistic depending mostly on the inverter used. If a high current is possible with the selected inverter, then DTC might be able to achieve this; otherwise, this might be achievable only in simulation. With that in mind, it looks as if DTC has the best speed response, with IFOC and FB linearization following closely behind.

Table 8 sums up dynamic speed performance. This shows the steady-state error that occurs after 2 s for all four drives. The speed command that was given was 100 rad/s. This shows that FB linearization, current hysteretic IFOC, and 6-sector DTC all performed similarly, while vectorized volts-per-hertz was slightly inferior. Also, the quickest that any motor controller could get the IM to 100 rad/s was 0.1 s using DTC. It
is also evident from Table 8 that IFOC had a very similar response with 0.102 s required to get to steady-state speed.

**Table 8. Maximum Steady State and Dynamic Performance**

<table>
<thead>
<tr>
<th></th>
<th>Vectorized V/Hz</th>
<th>DTC (6 sector)</th>
<th>IFOC</th>
<th>FB-Linearization</th>
</tr>
</thead>
<tbody>
<tr>
<td>S.S. Error after 2 s (rad/s)</td>
<td>0.41</td>
<td>0.29</td>
<td>0.28</td>
<td>0.20</td>
</tr>
<tr>
<td>S.S. Error %</td>
<td>0.41%</td>
<td>0.29%</td>
<td>0.28%</td>
<td>0.20%</td>
</tr>
<tr>
<td>Time to reach S.S. speed (s)</td>
<td>0.545</td>
<td>0.100</td>
<td>0.102</td>
<td>0.270</td>
</tr>
</tbody>
</table>

The third test that was run for all of the motor drives was a position control test. The command was given to move from the position of 0 rad to 1 rad, which is $1/2\pi$ of a rotation. This was done so that the speed of the motor had no impact on the results of the position control. The results of the test are shown in Figure 46. From this figure, it seems as if IFOC and DTC have superior performance if time to steady-state is the objective. If the objective is to get to steady-state without overshooting the commanded position, then it looks like FB linearization has the best performance since both DTC and IFOC have overshoots. The gains were again optimized for each motor controller to achieve the best results possible.
The last test that was done to compare all of the motor drives was the creation of Bode magnitude and phase plots. This test is useful in determining the output torque response to a torque input or even a possible disturbance. These plots were produced by first starting with a steady-state torque input of 2 N-m and then adding a sinusoidal ripple component with exactly one frequency at a time. A number of points were taken, and are displayed in Figure 47 for the magnitude, and Figure 48 for the phase. From Figures 47 and 48, it is relatively obvious that FB linearization has the best response. It can track the torque inputs most accurately. DTC has the next best input-output response. For the last two drives, IFOC and vectorized volts-per-hertz, it is unclear which has the better overall performance; vectorized volts-per-hertz tracks the input better at frequencies up to 25 Hz, but is unable to track at higher frequencies.
Figure 47. Bode Magnitude Plot of All Four Drives

Figure 48. Bode Phase Plot of All Four Drives
5. ALGORITHM IMPLEMENTATION

In this chapter, the hardware implementation of the above motor drives and controllers is explained in detail. This particular implementation uses a power inverter, dynamometer, induction motor and a computer. Some time will now be spent in describing the details of the hardware system. A brief hardware result will also be given, along with matching simulation results to show the ability of the system to be further realized. Figure 49 shows the complete setup of the above components as the motor drive system.

![Figure 49. Complete Setup of Motor Drive System](image)

5.1 Modular Inverter

The modular inverter was used as our power source coupled with the control scheme. The power inverter consists of three main modules, and is therefore called a “modular” inverter. It was designed in-house by eight UIUC graduate students along with a research engineer, from 2003 to 2006. Figure 50 shows the modules: control stage (1), power stage (2), and front end (3). These three stages are assembled in a black metal
housing that has $3\varphi$ ac voltage as an input, and $3\varphi$ ac voltage as an output. The modular inverter can also be used in the configuration where there is no front end module which then requires a dc source to be fed directly into the power stage.

5.1.1 Front end

The front end, shown in Figure 51 allows for the input $3\varphi$ ac voltage to be converted into a variable high voltage bus, e.g., 400 V. It consists two parts: a passive rectifier and an active dc-dc converter. The dc-dc converter is usually a boost but is capable of buck operation as well. Since a boost converter is unstable at lighter loads, over-voltage protection must be used in order to keep the output voltage from going unstable and increasing too high. In addition to overvoltage protection, it has inrush current protection and an emergency shut-off contactor. This board also comes with signal conditioning used in dc bus voltage calibration. While this board was available, it
was preferred to use a 240 V dc bus that was already available in the lab. This allowed for all attention to be paid to the control and power stages.

Figure 51. Modular Inverter Front End

5.1.2 Power stage

The power stage module, seen in Figure 52, is made up of five main components: gate dead-time circuitry, phase voltage, current and dc bus signal measurements, fault logic, gate drivers, and power electronics components. The gate drivers are powered by isolated power supplies that amplify the signal sent from the control board. They are used in order to provide enough current to insure that the gate turns on properly. There are three power stage configurations: the high-voltage high-current (400 V, 40 A_{RMS}) IGBT configuration, and the low-voltage high-current (100 V, 40 A_{RMS}) and high-voltage low-current (400 V, 10 A_{RMS}) MOSFET configurations. The IGBT that is used is the BSM100GB60DLC, the low-voltage high-current MOSFET is the IRFPS3815 and the high-voltage low-current MOSFET is the IRFPS40N50L.
The power stage has four phase legs, each of which is independently controlled by the control stage. This power inverter is more flexible since it has a fourth phase leg that allows for advanced control algorithms that might want to use this leg as a neutral phase or another unusual switching scheme. The gates are driven by an HCPL-316J, a 2 A optocoupler gate driver. The dead time between switching events is set by a DIP switch to allow for more or less dead time as the application needs. The actual gate signals are sent from the control board along with phase enables and a master enable. These control board signals are used by the power board CPLD along with overvoltage, overcurrent and gate fault signals to determine if each of the phase gates should be given a switching signal. If one is allowed to pass, the signal is routed to the correct gates via the CPLD. This can also be useful in rerouting signals when a problem arises or an advanced control scheme is used. The overvoltage, overcurrent and gate fault signals are determined independently on the power board. The overvoltage and overcurrent signals
are sent to the CPLD to impede the gate signals there, while the gate fault signal directly turns off the gate by using the enable pin on the HCPL-316J.

Each of the phase’s voltages and currents is scaled and measured along with the dc bus voltage on the power board. These signals are then conditioned and sent back to the control board as an analog signal for processing. To insure correct operation, these signals must be calibrated before first using the modular inverter. If these signals are not calibrated, erroneous voltages and currents will be read by the DSP, and faulty gate signals will be issued. The current is measured by the LAH 100-P Hall effect current transducer. The voltages are measured by using a simple voltage divider with high impedance resistors to allow for minimal power loss.

5.1.3 Control stage

The control module, pictured in Figure 53, is the brains of the modular inverter. This stage consists of three main units: the control board, daughterboard, and ezDSP TMS320F2812 DSP. It provides the enable, fault, reset and gate signals to the power board and front end. It receives the voltage, current, encoder, and peripheral signals from the front end and power board. The control board CPLD is used to route the signals to the correct pins on the daughterboard. It also provides conditioning for outgoing and incoming signals. The master enable and phase enable functionality is provided by a DIP switch on the control board that allows the user to turn off phases if needed.
The signals from the control board are routed to the daughterboard. The daughterboard allows for the signals to be conditioned yet again with a fourth order Sallen-Key-implemented low pass filter. Eight LEDs are available to the user for use in viewing digital signals coming from the daughterboard.

The last unit in the control module is the DSP. The ezDSP TMS320F2812 DSP does all the signal processing. It contains the program that is downloaded to it from the computer. The clock speed of the DSP is 150 MHz, and it is capable of 32-bit operations. The onboard available flash memory is 2.048 Mb. The TMS320F2812 DSP was created specifically for motor control operation, and therefore Park’s and Clark’s transformations are conveniently built in. Another convenient feature is that it has sixteen 12-bit ADC pins that allow for a high degree of precision while taking many possible measurements.
5.2 Dynamometer

The dynamometer setup includes a dynamometer and its controller. The dynamometer is the Magtrol Hysteresis Brake Dynamometer (HD) model HD-715-7N, shown in Figure 54, and the dynamometer controller is the Magtrol DSP6001, pictured in Figure 55. A hysteresis brake type dynamometer is only an absorptive dynamometer—it cannot provide power to move the motor, and it is therefore considered a type of brake. It is different from the typical disc brake in that it does not use mechanical friction losses to slow the rotation—it uses the eddy-current losses to dissipate the rotational energy. Also mounted on the dynamometer is the encoder that has 2048 lines of positional resolution to allow for very precise speed measurements which are required for many of the motor drives.
The dynamometer controller tells the dynamometer how much oppositional torque to apply to the shaft. It gets its commands from the user on the computer via the real-time data exchange (RTDX) interface. It has some visual outputs such as power, speed, and torque that allow the user to view real-time data. Scaled torque and speed measurements can also be read every 2 ms from the dynamometer controller and fed into an oscilloscope as an analog signal. This signal can then be read, and after scaling, can be used as the true torque and speed signals.

5.3 Induction Motor

A 3φ 1.5 hp Dayton induction motor was set up to be used in validation testing. The maximum power rating of the motor that can be used has hardware limitations. The limitations of the modular inverter are set to be around 10 kW, but theoretically with the IGBTs used, the maximum power output is closer to 16 kW. This would give a maximum power rating of about 13-21 hp. While the inverter can handle this much power, the maximum rating of the hysteresis dynamometer is 3 kW continuous power,
and 3.4 kW peak power, or 4 hp and 4.55 hp respectively. This is in effect limiting the torque of the load, and therefore this rating should be used so that the motor can be slowed to zero speed by the hysteresis braking system at all torques and speeds in the torque-speed curve.

5.4 Role of the Computer

The computer plays an extremely important role in the implementation of this project; it is where all of the software development and debugging occurs. All of the future motor controllers will be created in MATLAB-Simulink using the Real-Time Workshop (RTW) toolbox. From this high-level programming language, assembly code will be built that will then be downloaded onto the ezDSP using Texas Instrument’s Code Composer Studio (CCS) version 3.1. After downloading the model file from Simulink onto the DSP, the user would then need to run the m-file associated with the model file. This would then enable the program on the DSP, and allow for user input with RTDX. To get the motor to run correctly, the user would then need to enter inputs into the appropriate text boxes in the figure file that pertain to each individual motor controller program.

5.5 Hardware Results

In this section, brief hardware results from [31] will be shown for IFOC with current hysteretic control using the hardware setup described in Sections 5.1-5.4. The IFOC drive will then be compared to the simulated results from [31]. Although all of the motor drives were not created in hardware, results using IFOC show that if more time was allotted, the other three motor drives with various switching schemes could have
been created in hardware and compared experimentally with presumably the same results.

To simulate an electric or hybrid-electric vehicle driving cycle, a stepping torque profile was simulated using IFOC and DTC in Simulink. The simulated IFOC is from [1] while DTC is from [13]. The motor that was used in simulation is a 1.5 hp induction motor that matches the experimental setup. The torque load that was used both in simulation and experimentally is given in (86). The simulation was run for 8 s with torque commands of 5, 1, 4 and 2 N-m, changing every 2 s. Fixed stator and rotor flux commands of 0.52 V-s and 0.5 V-s are used considering that there is a 4% leakage inductance. The simulation results for IFOC with current-hysteretic control [1] and DTC using a switching table [13] are shown in Figures 56 and 57.

\[ T_L = 9.1 \cdot 10^{-4} \omega_r^2 \]  

(86)

Experimental results of IFOC shown in Figure 58 show speed overshoots higher than the simulated driving cycle, shown in Figure 56. The hardware system is underdamped compared to the simulated system, with an average overshoot around 30% versus 6.75% in simulations. The torque performance is almost identical to that shown in simulations; it has a very quick response, much like the simulations when a flux is present in the motor. With these results, it is evident that the hardware system performs extremely similar to the simulated system. The results would likely improve further if the torque PID loops were tuned more.
Figure 56. Speed Responses of IFOC and DTC under a Drive Cycle [31]

Figure 57. Torque Responses of IFOC and DTC under a Drive Cycle [31]
Figure 58. Torque and Speed Responses of IFOC under a Drive Cycle: $T_e$ (Upper Trace) 2 N·m/div, Speed (Lower Trace) 300 rpm/div. [31]
6. CONCLUSIONS AND FUTURE WORK

The work analyzes and compares different motor controllers using a second-order motor load, typical of many loads like a fan or industrial pump. All of the analysis has been done using this motor load and cannot be directly extrapolated to different applications, such as constant power loads, higher order loads, or loads that have inverse speed relationships. It was found and exhibited in Chapter 4 that each drive is advantageous in its own way. It was found that DTC and FB linearization have superior torque step performance. IFOC has a comparable torque response when SVPWM is used for the switching scheme in place of the standard current hysteretic switching. The second test, a speed step response, showed that FB linearization, DTC and IFOC have similar performances, while vectorized volts-per-hertz performs poorly. DTC and IFOC reacted very quickly in the position test, while FB linearization had the best position response without a torque overshoot. The last test, a Bode plot of the motor drive systems, showed that FB linearization has the best tracking ability out of all of the drives, with DTC in a close second place. Overall, it seems that FB linearization, given the right conditions and gains, performs the most admirably, while vectorized volts-per-hertz is by far the worst of the four major motor controllers mostly because of its inherent disadvantage in the lack of feedback signals.

It was found that the current literature assumes that certain motor controllers are associated with certain switching schemes, but this is largely an arbitrary connection. This thesis discussed how the motor controller and the switching scheme should be thought of as completely decoupled. This work also has concentrated on comparing the main types of induction motor controllers in standard topologies, with a minor emphasis
on comparing drives using a common switching scheme. In the future, an investigation into which switching scheme is optimal with each motor controller could be carried out. Drives could then be more fairly compared to determine the best motor drive system. Pole placement techniques should be considered for determining optimum performance of motor controllers in place of PID loops. Also, using an enhanced flux observer might improve convergence times. In the future “inner loop” control, or control using currents, voltages and fluxes, versus “outer loop” control, or one that uses torques and speeds, should be analyzed to help in the design of improved motor controllers.

A further, more comprehensive analysis of these results by hardware verification should be completed. DTC hardware verification is in progress, but not completely finished as of this writing. Vectorized volts-per-hertz has been successfully implemented in hardware along with IFOC. IFOC hardware results have been presented in this body of work to show the achievability of the other two motor drives in addition to bringing credibility to the other drives’ simulation results. The lone motor controller not yet attempted here is feedback linearization; this controller should be explored in hardware in the near future to compare against other common motor controllers.
REFERENCES


APPENDIX A: ADDITIONAL DATA

Table 9 shows the induction motor parameters used in simulation. Table 10 shows the PID gains that were used in simulation comparing IFOC and DTC using SVPWM. Table 11 gives more results for the performance of a DTC drive when measurements are inaccurate.

Table 9. Induction Motor Parameter Data

<table>
<thead>
<tr>
<th>Induction Motor Data</th>
<th>Symbols</th>
<th>Description</th>
<th>Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rs</td>
<td>Stator Resistance</td>
<td>1.5293 Ω</td>
<td></td>
</tr>
<tr>
<td>Rr</td>
<td>Rotor Resistance</td>
<td>0.7309 Ω</td>
<td></td>
</tr>
<tr>
<td>Ls</td>
<td>Stator Inductance</td>
<td>0.20135 H</td>
<td></td>
</tr>
<tr>
<td>Lr</td>
<td>Rotor Inductance</td>
<td>0.20315 H</td>
<td></td>
</tr>
<tr>
<td>Lls</td>
<td>Stator Leakage Inductance</td>
<td>0.00356 H</td>
<td></td>
</tr>
<tr>
<td>Llr</td>
<td>Rotor Leakage Inductance</td>
<td>0.00535 H</td>
<td></td>
</tr>
<tr>
<td>Lm</td>
<td>Magnetizing Inductance</td>
<td>0.19779 H</td>
<td></td>
</tr>
<tr>
<td>P</td>
<td>Number of Poles</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>Pm</td>
<td>Power of Induction Motor</td>
<td>3 hp</td>
<td></td>
</tr>
</tbody>
</table>

Table 10. PID Gains for IFOC-SVPWM and DTC-SVPWM

<table>
<thead>
<tr>
<th></th>
<th>ki</th>
<th>kp</th>
<th>kd</th>
</tr>
</thead>
<tbody>
<tr>
<td>IFOC-SVPWM- Speed</td>
<td>10</td>
<td>1000</td>
<td>0.05</td>
</tr>
<tr>
<td>IFOC-SVPWM- Torque</td>
<td>3000</td>
<td>1000</td>
<td>60</td>
</tr>
<tr>
<td>DTC-SVPWM- Vq</td>
<td>1000</td>
<td>20</td>
<td>1</td>
</tr>
<tr>
<td>DTC-SVPWM- Vd</td>
<td>50</td>
<td>1000</td>
<td>0</td>
</tr>
</tbody>
</table>
Table 11. DTC Performance, Change of Multiple Simultaneous Measurements

<table>
<thead>
<tr>
<th>Change in Gains (increase/decrease)</th>
<th>Torque Increase</th>
<th>Torque Decrease</th>
<th>Flux Increase</th>
<th>Flux Decrease</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vds &amp; Ids</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5%</td>
<td>4.12</td>
<td>4.12</td>
<td>4.25</td>
<td>4.75</td>
</tr>
<tr>
<td>10%</td>
<td>4.15</td>
<td>4.12</td>
<td>4.00</td>
<td>5.10</td>
</tr>
<tr>
<td>15%</td>
<td>4.20</td>
<td>4.12</td>
<td>3.90</td>
<td>5.35</td>
</tr>
<tr>
<td>25%</td>
<td>4.23</td>
<td>4.12</td>
<td>3.65</td>
<td>6.05</td>
</tr>
<tr>
<td>Vqs &amp; Iqs</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5%</td>
<td>4.12</td>
<td>4.12</td>
<td>4.45</td>
<td>4.50</td>
</tr>
<tr>
<td>10%</td>
<td>4.12</td>
<td>4.12</td>
<td>4.42</td>
<td>4.55</td>
</tr>
<tr>
<td>15%</td>
<td>4.12</td>
<td>4.12</td>
<td>4.40</td>
<td>4.60</td>
</tr>
<tr>
<td>25%</td>
<td>4.12</td>
<td>4.12</td>
<td>4.35</td>
<td>4.65</td>
</tr>
<tr>
<td>Vds &amp; Vqs</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5%</td>
<td>4.12</td>
<td>4.12</td>
<td>4.20</td>
<td>4.80</td>
</tr>
<tr>
<td>10%</td>
<td>4.18</td>
<td>4.12</td>
<td>4.00</td>
<td>5.15</td>
</tr>
<tr>
<td>15%</td>
<td>4.22</td>
<td>4.12</td>
<td>3.82</td>
<td>5.65</td>
</tr>
<tr>
<td>25%</td>
<td>4.24</td>
<td>4.12</td>
<td>3.55</td>
<td>7.00</td>
</tr>
<tr>
<td>Vqs &amp; Iqs</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5%</td>
<td>4.12</td>
<td>4.12</td>
<td>4.50</td>
<td>4.48</td>
</tr>
<tr>
<td>10%</td>
<td>4.12</td>
<td>4.12</td>
<td>4.55</td>
<td>4.45</td>
</tr>
<tr>
<td>15%</td>
<td>4.12</td>
<td>4.12</td>
<td>4.55</td>
<td>4.40</td>
</tr>
<tr>
<td>25%</td>
<td>4.12</td>
<td>4.12</td>
<td>4.55</td>
<td>4.32</td>
</tr>
<tr>
<td>Vds &amp; Vqs</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5%</td>
<td>4.12</td>
<td>4.12</td>
<td>4.15</td>
<td>4.88</td>
</tr>
<tr>
<td>10%</td>
<td>4.15</td>
<td>4.12</td>
<td>3.93</td>
<td>5.85</td>
</tr>
<tr>
<td>15%</td>
<td>4.20</td>
<td>4.12</td>
<td>3.85</td>
<td>unstable</td>
</tr>
<tr>
<td>25%</td>
<td>4.22</td>
<td>4.12</td>
<td>3.40</td>
<td>unstable</td>
</tr>
<tr>
<td>Iqs &amp; Ids</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5%</td>
<td>4.12</td>
<td>4.12</td>
<td>4.60</td>
<td>4.40</td>
</tr>
<tr>
<td>10%</td>
<td>4.12</td>
<td>4.12</td>
<td>4.62</td>
<td>4.32</td>
</tr>
<tr>
<td>15%</td>
<td>4.12</td>
<td>4.12</td>
<td>4.65</td>
<td>4.25</td>
</tr>
<tr>
<td>25%</td>
<td>4.12</td>
<td>4.12</td>
<td>5.05</td>
<td>4.12</td>
</tr>
<tr>
<td>Vds, Vqs, Ids &amp; Iqs</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5%</td>
<td>4.12</td>
<td>4.12</td>
<td>4.21</td>
<td>4.80</td>
</tr>
<tr>
<td>10%</td>
<td>4.15</td>
<td>4.12</td>
<td>4.05</td>
<td>5.15</td>
</tr>
<tr>
<td>15%</td>
<td>4.20</td>
<td>4.12</td>
<td>3.85</td>
<td>5.50</td>
</tr>
<tr>
<td>25%</td>
<td>4.23</td>
<td>4.12</td>
<td>3.55</td>
<td>6.40</td>
</tr>
</tbody>
</table>
APPENDIX B: SIMULATION MODELS, CONSTANTS, AND VARIABLES

Figures 59-68 show the models used for simulation: the induction motor model, hysteresis, a DTC flux calculator, the DTC switching algorithm, the full DTC model, the IFOC current hysteretic model, the FB linearization PID current controller and flux observer, the whole FB linearization model, and the vectorized volts-per-hertz model.

Following is the code for initializing all of these simulations.

```matlab
%Machine parameters
%Smaller 2.2kW (3 HP) machine used in DTC paper
nphase = 3; %number of phases
P = 4; %number of poles
Pm = 3; %HP, rated output power
%V = 380; %VRMS, rated voltage
%Vb = V/sqrt(3); %rated line-neutral RMS voltage
rpm = 1740; %RPM, rated speed
wb = rpm*2*pi/60; %rated radian frequency
Tb = 12; %Nm, rated torque
Ib = 8.6; %ARMS, rated phase current
rs = 1.5293; %Ohms, stator resistance
Lls = 0.0036; %H, stator leakage inductance
LM = .1978; %H, magnetizing inductance
Llr = 0.0053; %H, referred rotor leakage inductance
rr = 0.7309; %Ohms, referred rotor resistance
Lss = Lls+LM; %Ohms, Stator Self-Inductance
Lrr = Llr+LM; %Ohms, Rotor Self-Inductance
sigma=1-LM^2/(Lrr*Lss);
J = 0.01; %kg*m^2, rotor inertia
Bm=.0001;
vqr=0;
vdr=0;
v0r=0;
lamSref = .52; %Wb, taken from no load steady state
dellamS = 0.01; %stator flux hysteresis band half width
delTe = .01; %Nm, torque hysteresis band half width
fsw=10000; %fixed switching frequency
gamma=Lls*rr/Llr+rs;
k_q=5; %quantization gain
Kt=.000182; %2nd Order Load Constant
V0 = 300; %V, input dc voltage
Ts=5e-6; %Sampling Frequency
Linv=inv([Ls 0 0 Lm 0 0; 0 Ls 0 0 Lm 0; 0 0 Lls 0 0 0; Lm 0 0 Lr 0 0; 0 Lm 0 0 Lr 0; 0 0 Lrm 0 Lrr]);
```
Figure 59. Induction Motor Model
Figure 60. DTC Simulation Model
Figure 61. DTC Switching Algorithm
Figure 62. Hysteresis of a Signal

Figure 63. DTC Flux Calculator by Sector
Figure 64. IFOC Current Hysteretic Model
Figure 65. FB Linearization Current Hysteretic Model
Figure 66. FB Linearization Position PID Current Controller

Figure 67. FB Linearization Flux Observer
Figure 68. Vectorized V/Hz Control Model