

## Implementation and simulation of sensorless control and field weakening for an induction machine

Using the the Statically Compensated Voltage Model

*Master of Science Thesis*

BJÖRKQVIST, JAKOB  
STOCKMAN, DAVID

Department of Energy and Environment  
Division of Electric Power Engineering  
CHALMERS UNIVERSITY OF TECHNOLOGY  
Göteborg, Sweden 2014



**Implementation and simulation of  
sensorless control and field  
weakening for an induction machine**  
**Using the the Statically Compensated Voltage Model**

BJÖRKQVIST, JAKOB  
STOCKMAN, DAVID

Department of Energy and Environment  
Division of Electric Power Engineering  
CHALMERS UNIVERSITY OF TECHNOLOGY  
Göteborg, Sweden 2014

Implementation and simulation of sensorless control and field weakening for an induction machine  
Using the the Statically Compensated Voltage Model  
BJÖRKQVIST, JAKOB  
STOCKMAN, DAVID

© BJÖRKQVIST, JAKOB  
STOCKMAN, DAVID, 2014.

Department of Energy and Environment  
Division of Electric Power Engineering  
Chalmers University of Technology  
SE-412 96 Göteborg  
Sweden  
Telephone +46 (0)31-772 1000

Cover:

Text concerning the cover illustration. In this case: Statically Compensated Voltage Model control scheme

Chalmers Bibliotek, Reproservice  
Göteborg, Sweden 2014

Implementation and simulation of sensorless control and field weakening for an induction machine  
Using the the Statically Compensated Voltage Model

BJÖRKQVIST, JAKOB

STOCKMAN, DAVID

Department of Energy and Environment

Division of Electric Power Engineering

Chalmers University of Technology

## Abstract

AROS electronics produce Permanent Magnet Synchronous Machines but the market for magnets can be very volatile in that the prices may fluctuate significantly. The Induction machine is then an attractive replacement. The robustness and simple construction makes it one of the most used electrical drives in the industry. However, it is often controlled with a speed sensor or an open loop configuration like the well known Volt/Hertz-control. A field oriented sensorless control would make the Induction machine even more attractive from an economical and maintenance point of view, but the problem is that the flux and the speeds need to be estimated. The largest drawback with sensorless control is that the machine will eventually turn unstable in the low speed region. In order to reach speeds above rated speed, field weakning is required. A field oriented sensorless control model with a flux estimator known as the Statically Compensated Voltage Model has been modelled and implemented together with field weakning in one of AROS's electronics digital signal processors. The control model was simulated in Matlab/SIMULINK to obtain information about the system robustness and its limitations. The implementation was done in a C-language environment on a 16-bit fixed point processor where tests showed that the system is operating well at nominal speed of 1400 RPM with a nominal torque of 7.5 Nm. The field weakening algorithm made it possible to reach twice the rated speed, 2800 RPM, with a load torque of 5.5 Nm. At about 3100 RPM the machine turns unstable because the maximum voltage that the converter can put out is reached and the current therefore becomes uncontrollable.

**Index Terms:** Induction machine, Field weakening, Simulink, SCVM, Sensorless, Vector control, Field oriented control



## **Acknowledgements**

First of all, thanks to AROS electronics for making it possible for us to do our thesis. We want to thank our supervisor at AROS, Mikael Alatalo for his support, help and encouragement. We also want to thank our examiner, Stefan Lundberg for giving us helpful directions. Finally, we want to thank the AROS employees Henrik Neugebauer and Andreas Pettersson for their help.

Jakob Björkqvist, David Stockman

Göteborg, Sweden, 2014



# Contents

|  |            |
|--|------------|
| <b>Abstract</b>  | <b>iii</b> |
| <b>Acknowledgements</b>  | <b>v</b>   |
| <b>Contents</b>  | <b>vii</b> |
| <b>1 Introduction</b>  | <b>1</b>   |
| 1.1 Aim . . . . .  | 1          |
| 1.2 Problem . . . . .  | 2          |
| 1.3 Scope . . . . .  | 2          |
| 1.4 Method . . . . .   | 2          |
| 1.5 Previous work . . . . .  | 3          |
| <b>2 Technical background</b>  | <b>5</b>   |
| 2.1 Review of the control methods . . . . .  | 5          |
| 2.2 Induction machine modeling . . . . .   | 6          |
| 2.2.1 Mechanical construction of a three phase induction machine . . . . .           | 6          |
| 2.2.2 Operational principles . . . . .   | 6          |
| 2.2.3 Space vectors . . . . .  | 7          |
| 2.2.4 The dynamic T-model for the IM in stationary coordinates . . . . .             | 8          |
| 2.2.5 The dynamic inverse gamma model for the IM in stationary coordinates . . . . . | 10         |
| 2.2.6 The dynamic inverse gamma model for the IM in rotating coordinates . . . . .   | 11         |
| 2.3 Flux estimators . . . . .  | 12         |
| 2.3.1 The current model (CM) . . . . .   | 12         |
| 2.3.2 The Statically Compensated Voltage Model (SCVM) in DFO . . . . .               | 12         |
| 2.3.3 IFO implementation of the SCVM . . . . .                                       | 14         |
| 2.3.4 Parameter sensitivity of the CM and SCVM . . . . .                             | 15         |
| 2.4 Current controller derivation . . . . .  | 15         |
| 2.4.1 Current reference calculation and current limiter . . . . .                    | 17         |
| 2.5 Field weakening . . . . .  | 17         |
| 2.6 Speed controller derivation . . . . .  | 18         |
| <b>3 MATLAB/Simulink implementation of the models</b>                                | <b>19</b>  |
| 3.1 Overview of the matlab model . . . . .   | 19         |
| 3.2 Machine Model . . . . .  | 21         |
| 3.3 The current controller block . . . . .   | 21         |

## Contents

|          |   |           |
|----------|---|-----------|
| 3.4      | The reference current calculation block . . . . .                   | 23        |
| 3.5      | The statically compensated voltage model block . . . . .            | 24        |
| 3.6      | The speed controller block . . . . .                                | 25        |
| 3.7      | The field weakening block . . . . .                                 | 27        |
| <b>4</b> | <b>Test setup and controller implementation</b>                     | <b>29</b> |
| 4.1      | Test bench . . . . .  | 29        |
| 4.1.1    | Friction measurement of the IM . . . . .                            | 30        |
| 4.1.2    | Startup procedure . . . . .   | 30        |
| 4.2      | Implementation of the controller structure . . . . .                | 30        |
| 4.2.1    | Fixed point DSP . . . . .   | 30        |
| 4.2.2    | Software implementation . . . . .                                   | 31        |
| <b>5</b> | <b>Results</b>  | <b>33</b> |
| 5.1      | Drive system using the current model flux observer . . . . .        | 33        |
| 5.2      | Drive system using the SCVM flux observer . . . . .                 | 37        |
| 5.2.1    | Nominal speed and torque . . . . .                                  | 37        |
| 5.2.2    | Twice the nominal speed with a 5.5 Nm load torque . . . . .         | 42        |
| 5.2.3    | Testing the FW-algorithm with several speed steps . . . . .         | 47        |
| 5.2.4    | Maximum speed with a load torque of 5.5 Nm . . . . .                | 49        |
| <b>6</b> | <b>Discussion and conclusions</b>                                   | <b>51</b> |
| 6.1      | Discussion . . . . .  | 51        |
| 6.2      | Conclusion . . . . .  | 52        |
| 6.3      | Future work . . . . .   | 52        |
|          | <b>References</b>   | <b>53</b> |
| <b>A</b> | <b>Appendix</b>   | <b>55</b> |
| A.1      | Matlab code . . . . .   | 55        |
| A.1.1    | Machine model . . . . .   | 55        |
| A.1.2    | simulation file with parameters . . . . .                           | 58        |
| A.1.3    | plot file . . . . .   | 59        |
| A.2      | IM ratings . . . . .  | 62        |
| A.3      | Induction machine parameter measurements and calculations . . . . . | 62        |
| A.4      | Overview of the Matlab block diagram . . . . .                      | 64        |

# Chapter 1

## Introduction

This thesis has been conducted in cooperation with AROS Electronics. AROS Electronics produce permanent magnet synchronous machines (PMSM), but also the controllers that are used for the PMSM. The magnets used in a PMSM can be made of a mixture of the rare-earth minerals neodymium, iron and boron (NdFeB). During the past years, the price of these magnets has fluctuated significantly, for instance during the summer of 2011, the prices were driven up as much as 30 times the original [8]. Although prices have decreased since then, the market may cause the magnet price to increase even further. AROS Electronics believe that it is possible that the Induction machine (IM) will become a cheaper alternative in the future and they might start producing them. Today there are several different methods of controlling the IM, where one such method is known as vector control or field oriented control (FOC) [2].

When using FOC as means of controlling an IM the flux in the machine needs to be estimated. For that, one can either use a current or voltage based flux estimator. The names refer to the electrical equations that are describing the operating principles of the IM. The current model flux estimator uses the electrical rotor equation as means of estimating the flux and the voltage model uses the stator equation. The current model is always stable and easy to implement compared to the voltage model. But it is lacking in performance at high rotor speeds and must always use a speed sensor for speed measurement [2].

In this work, the Statically Compensated Voltage Model (SCVM) will be used, which is a further modification of the voltage model. It may turn unstable at low speeds but performs well at nominal speeds and does not need a speed sensor in order to operate [3]. Removing the need of a speed sensor is of interest because this decreases the cost of the motor controller.

Certain applications may require a machine to operate above base speed and this can be achieved by implementing field weakening (FW). FW essentially means that the flux is weakened in the IM in order to decrease the induced back-EMF. The back-EMF is equal to the rotor flux times the electrical rotor speed and if it reaches the voltage level that the inverter can put out the speed may not increase any further, without reducing the flux [4].

### 1.1 Aim

The aim of this work is to simulate the voltage model with field weakening in a Matlab/SIMULINK environment, and also to implement the control algorithms in a Digital signal processor (DSP). The

goal of the implementation is to reach three times the nominal speed, with an arbitrary load torque.

## **1.2 Problem**

In order to achieve the aim, the problem was divided into the following sub problems:

- Simulating the model of the converter, IM, current controller and speed controller
- Implementing the CM
- Simulating and testing the drive system
- Implementing the FW
- Simulating and testing the drive system with FW
- Implementing the SCVM
- Simulating the SCVM without/with FW
- Implementing the control system into the test bench
- Comparing the simulations with the measured data from the implemented system

The current model was implemented at first because it makes the implementation of the SCVM so much easier.

## **1.3 Scope**

The SCVM will be investigated thoroughly at nominal speeds and speeds above. It is however of interest to determine the operating point where the machine turns unstable in the low speed region. There will be no theoretical stability analysis of the SCVM, since this is an implementation study. Furthermore the resistance and inductance of the machine are subject to change during operation. This will affect the drive system performance, but no investigations will be made in order to determine the impact of these parameter errors.

## **1.4 Method**

Simulations in MatLab/SIMULINK were made in order to give an idea about how bandwidths and controller parameters affected the system. When the simulations showed that the system was stable it was implemented in a fixed-point DSP. Since implementation in a fixed-point processor is not straightforward, the SCVM was implemented in steps. This was done by first implementing the current model which uses a speed sensor. The SCVM was then executed along the current model and parameters such as estimated speed, estimated angle and flux could be extracted. When it was seen that all the estimates were calculated correctly, the SCVM could then be used to run the IM. Finally to verify that the implemented system was behaving as it should, measurements were performed and compared with the simulations.

## **1.5 Previous work**

An earlier master thesis at AROS by H. Carlsson and J. Bergerlind [1] has evaluated the SCVM at low speeds where known problems such as instability occurs. It has also investigated the influence of parameter errors. The motor used in this work is exactly the same as the motor that the earlier master thesis evaluated, except that this motor has one more pole pair and its parameters differ slightly.

*Chapter 1. Introduction*

## Chapter 2

# Technical background

The aim of this chapter is to supply the reader with knowledge about the IM, vector control and flux estimators. The reader is assumed to have at least some rudimentary knowledge about electric drives, control theory and vector control. Thus this chapter only briefly explains these principles. A more complete walkthrough of vector control and the dynamic IM model can be found in [2].

### 2.1 Review of the control methods

The basic working principles of the induction machine were developed during the 19th century. New discoveries in physics such as electromagnetism and later, the invention of the rotating magnetical field were giving rise to a numerous of new electromechanical equipments, one of them being the induction machine. Due to its simple construction, robustness and cheap manufacturing cost, the induction machine is one of the most widely-used electrical machines. Through the years since its invention it has been mostly used in fixed speed applications such as driving fans, pumps, compressors and more. During the recent decades, modern power electronics has made it more popular in variable-speed applications [5].

Today there are several different methods for controlling the speed of the machine, where one of the most common control methods is the ' $V/f$ '-control, or volt/hertz control. This method uses an open loop control and sets the stator voltage and frequency after a desired speed reference. The ' $V/f$ '-control works well for applications where quick torque response and precise control of the speed is not important [2]. However, some applications require precision speed holding and accurate torque control which can be accomplished with field oriented control (FOC) [2]. During the 1960s, a lot of research were done in the induction machine control area. It was mainly due to the desire of replacing the DC-machine in applications where quick torque response was needed [2].

The result of the research resulted in the development of FOC. This control method requires knowledge about the stator- or rotor flux angle. In this work, the rotor flux angle is chosen. Measurement of the flux angle tends to be difficult and expensive and therefore most modern FOC drives relies on flux estimation [2]. One common flux estimator known as the Current Model (CM) uses the speed feedback from a tachometer mounted on the rotor shaft. However, using a tachometer comes with a few drawbacks [6]:

- Economic - The tachometer increases the cost of the drive system.

- Maintenance - A failure in the tachometer would cause the CM to fail, thus a sensorless drive would be more reliable in that respect.
- Environment - The tachometer is sensitive to the surrounding environment. It could for example not be used in chemical plants [6].

A removal of the tachometer would erase these drawbacks. This work will focus on speed-sensorless control using a flux estimator called the Statically Compensated Voltage Model (SCVM) which is a further development from the classic Voltage Model (VM). Figure 2.1 Shows a FOC drive system using the SCVM. The control structure used is a cascaded system with a fast inner current controller and an outer slower speed controller. In the MATLAB/Simulink chapter, the blocks in Figure 2.1 are described.

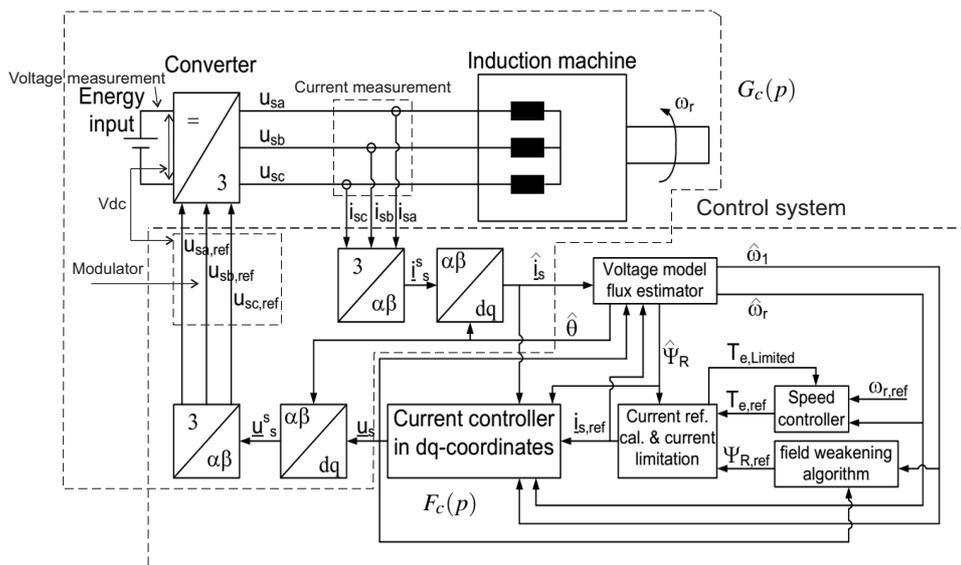


Figure 2.1: Overview of the FOC drive system using the SCVM. Courtesy of Stefan Lundberg

## 2.2 Induction machine modeling

### 2.2.1 Mechanical construction of a three phase induction machine

An induction machine consists of a stator and a rotor. The stator windings consists of coils that are located in slots and these coils make up three identical windings that are distributed around the stator. The three windings are shifted 120 degrees in respect to one another, and so they create a balanced three phase system if a three phase ac supply is connected to the windings. The rotor is normally constructed in the shape of a squirrel cage and so its rotor bars are short circuited in both ends [5].

### 2.2.2 Operational principles

A rotating magnetic field is created when an alternating three phase voltage source is connected to the stator windings. Its rotational speed,  $\omega_1$ , commonly known as the synchronous speed or stator speed, depends on the frequency of the stator voltage  $f_1$ . When the rotating magnetic field cuts the

## 2.2. Induction machine modeling

rotor bars, a voltage is induced in the rotor windings. Due to the short circuited windings, the induced rotor voltage will drive a current in the rotor.

The machine produces torque when the induced currents in the rotor bars interact with the rotating magnetic field. This torque is a result of the so called Lorentz Force, and depends on the relative motion between the rotating magnetic field and the current carrying rotor bars. During motor operation, the electrical rotor speed,  $\omega_r$ , will always lag the electrical stator speed. The relative motion between these two speeds defines the slip:

$$s = \frac{\omega_1 - \omega_r}{\omega_1} \quad (2.1)$$

The slip is often expressed as a normalised quantity where a slip of 0 means that  $\omega_1 = \omega_r$  and a slip of 1 corresponds to a stationary rotor. A larger slip will cause the flux to cut the rotor bars more frequently, inducing a higher voltage and thereby a higher torque is created. A slip of 0 would induce zero voltage in the rotor and thus no torque is created. Furthermore, the angular slip frequency is defined as

$$\omega_2 = s\omega_1 = \omega_1 - \omega_r \quad (2.2)$$

The actual mechanical rotor speed is written as

$$\Omega_r = \frac{\omega_r}{n_p} \quad (2.3)$$

where  $n_p$  is the number of pole pairs.

### 2.2.3 Space vectors

It is often adequate to use phasor diagrams and the equivalent circuit, in order to analyse the IM during steady state operation. However, in a variable-speed drive the IM cannot be described by these methods when the frequency, phase or amplitude of the stator voltage is changed. Therefore space vectors must be used. The purpose of vector control, regarding the IM is to mathematically transform it into a separately magnetized DC machine, since it is much easier to implement a control system for a DC machine.

A 3-phase induction machine is constructed such that each phase is shifted 120 degrees or  $\frac{2\pi}{3}$  radians in respect to one another. Thus the stator voltages in each phase can be described as

$$\begin{aligned} V_a &= \hat{U} \cos(\omega t) \\ V_b &= \hat{U} \cos\left(\omega t - \frac{2\pi}{3}\right) \\ V_c &= \hat{U} \cos\left(\omega t - \frac{4\pi}{3}\right) \end{aligned} \quad (2.4)$$

For an arbitrary chosen time,  $t_0$ , the sum of  $V_a$ ,  $V_b$  and  $V_c$  will be zero.

$$V_a(t_0) + V_b(t_0) + V_c(t_0) = 0 \forall t \quad (2.5)$$

This results in that one of the phases can be expressed by means of the other two,

$$V_a(t) = -V_b(t) - V_c(t) \quad (2.6)$$

The system can now be described as a 2-phase system in the complex plane as such

$$\mathbf{v}^s(t) = v_\alpha + jv_\beta = \frac{2}{3}K[v_a(t) + e^{j\frac{2\pi}{3}}v_b(t) + e^{j\frac{4\pi}{3}}v_c(t)] \quad (2.7)$$

where  $K$  is a scaling factor that can be chosen depending on the application. Hereafter  $K = 1$  will be used, this is called amplitude invariant transformation [2]. The complex stator voltage vector,  $\mathbf{v}^s(t)$ , is called a space vector and is rotating with the angular frequency,  $\omega_1$ . Furthermore, the superscript "s" shows that  $\mathbf{v}^s(t)$  is referred to the stator reference frame. The 3-phase to 2-phase transformation matrix can be expressed as

$$\begin{bmatrix} v_\alpha \\ v_\beta \end{bmatrix} = K \underbrace{\begin{bmatrix} \frac{2}{3} & -\frac{1}{3} & -\frac{1}{3} \\ 0 & \frac{1}{\sqrt{3}} & -\frac{1}{\sqrt{3}} \end{bmatrix}}_{T_{32}} \begin{bmatrix} v_a(t) \\ v_b(t) \\ v_c(t) \end{bmatrix} \quad (2.8)$$

This transformation is known as the Clarke transformation. To transform back to 3-phase the inverse matrix,  $T_{32}^{-1} = T_{23}$ , is used according to

$$\begin{bmatrix} v_a(t) \\ v_b(t) \\ v_c(t) \end{bmatrix} = \frac{1}{K} \underbrace{\begin{bmatrix} 1 & 0 \\ -\frac{1}{2} & \frac{\sqrt{3}}{2} \\ -\frac{1}{2} & -\frac{\sqrt{3}}{2} \end{bmatrix}}_{T_{23}} \begin{bmatrix} v_\alpha \\ v_\beta \end{bmatrix}. \quad (2.9)$$

In order to be able to use the DC-quantities in vector control, the reference frame needs to rotate with the same angular frequency as  $\omega_1$ . This will give DC-quantities in steady state and ease both analysis and control algorithm implementation. To rotate the reference frame, the space vector needs to be multiplied with opposite rotation of  $\omega_1$  according to

$$\mathbf{v} = \mathbf{v}^s e^{-j(\omega_1 t)} = K\hat{V} e^{j(\omega_1 t)} e^{-j(\omega_1 t)} = K\hat{V} = v_d + jv_q \quad (2.10)$$

This is called a Park transformation or dq-transformation. Since the purpose of vector control is to control variable speed drives,  $\omega_1$  will not be constant. In the FOC drive system the dq-coordinate system is aligned with the field in the machine. In this work it is aligned with the rotor flux vector and the angle of the rotor flux space vector,  $\theta_1$  is used as transformation angle. The transformation between  $\alpha\beta$ - and  $dq$ -coordinates are then given by

$$\mathbf{v} = \mathbf{v}^s e^{-j\theta_1} \quad (2.11)$$

and the transformation between dq and  $\alpha\beta$  are given by

$$\mathbf{v}^s = \mathbf{v} e^{\theta_1} \quad (2.12)$$

## 2.2.4 The dynamic T-model for the IM in stationary coordinates

By assuming that the sum of all instantaneous voltages and currents are zero, space vectors can be used to model the IM. The representation in figure 2.2 is referred to as the T-model and includes the stator, rotor and magnetizing impedances. Here the core losses are neglected, leaving only the magnetizing inductance to represent the core.

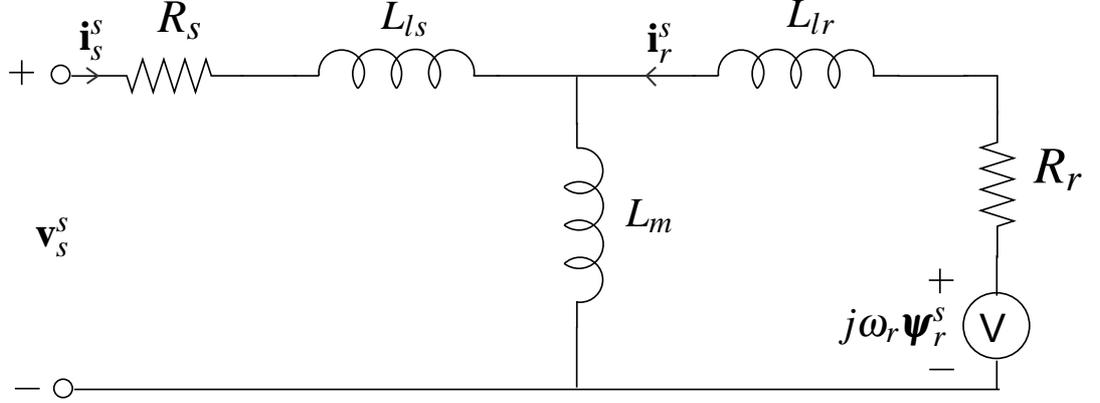


Figure 2.2: The dynamic T-model of the induction machine

The dynamic electrical and mechanical behaviour of the IM in the stationary  $\alpha\beta$  coordinate system is described in [7]. Starting with the electrical equations for the stator and rotor

$$\mathbf{v}_s^s = R_s \mathbf{i}_s^s + \frac{d\boldsymbol{\psi}_s^s}{dt} \quad (2.13)$$

$$\mathbf{v}_r^s = R_r \mathbf{i}_r^s + \frac{d\boldsymbol{\psi}_r^s}{dt} - j\omega_r \boldsymbol{\psi}_r^s \quad (2.14)$$

where  $R_s$  is the winding resistance of the stator and  $R_r$  is the winding resistance of the rotor. The stator- and rotor flux linkage are described as

$$\boldsymbol{\psi}_s^s = L_s \mathbf{i}_s^s + L_m \mathbf{i}_r^s \quad (2.15)$$

$$\boldsymbol{\psi}_r^s = L_r \mathbf{i}_r^s + L_m \mathbf{i}_s^s \quad (2.16)$$

The self inductance of the stator and rotor are composed of the leakage and mutual inductance as

$$L_s = L_r = L_{ls} + L_m \quad (2.17)$$

since it is assumed that the leakage is equal in the stator and rotor, i.e.  $L_{ls} = L_{lr}$ . The dynamic mechanical equations for the rotor speed and rotor position are described as

$$\frac{J}{n_p} \frac{d\omega_r}{dt} = T_e - T_L \quad (2.18)$$

$$\frac{d\theta}{dt} = \omega_r \quad (2.19)$$

where  $J$  is the moment of inertia,  $n_p$  is the number of pole pairs,  $\omega_r$  is the electrical speed and  $\theta$  is the rotor position. The torque produced by the IM can be expressed as

$$T_e = \frac{3n_p}{2} \text{Im} \left\{ \boldsymbol{\psi}_s^{s*} \mathbf{i}_s^s \right\} = \frac{3n_p}{2} L_m (i_{r\alpha} i_{s\beta} - i_{r\beta} i_{s\alpha}) \quad (2.20)$$

The load torque  $T_L$  is assumed to have a linear friction dependency on the speed such as

$$T_L = b\Omega_r + T_{L,extra} = b \frac{\omega_r}{n_p} + T_{L,extra} \quad (2.21)$$

where  $b$  is the viscous damping coefficient,  $\Omega_r$  is the mechanical speed and  $T_{L,extra}$  is the extra load torque.

### 2.2.5 The dynamic inverse gamma model for the IM in stationary coordinates

The disadvantage with the T-model is that it is overparametrized, leading to more complicated control implementation. Another circuit representation is the inverse- $\Gamma$  model, where the rotor leakage inductance is transferred to the stator side, forming a total leakage inductance [7], see figure 2.3. The transformation from the T-model to the inverse- $\Gamma$ -model is given by the transformation coefficient  $\gamma = \frac{L_m}{L_r}$  and the equations

$$\psi_R = \psi_r \gamma \quad (2.22)$$

$$L_\sigma = L_s - \gamma \quad (2.23)$$

$$L_M = \gamma L_m \quad (2.24)$$

$$R_R = R_r \gamma^2 \quad (2.25)$$

where  $L_\sigma$ ,  $L_M$ ,  $R_R$ . and  $\psi_R$  are the new variables for the inverse- $\Gamma$  model.

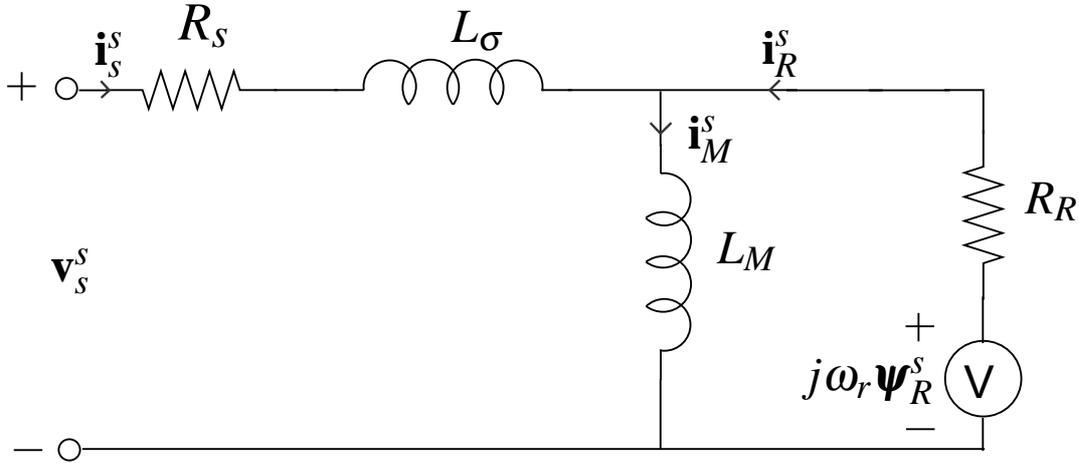


Figure 2.3: Dynamic inverse- $\Gamma$ -model

Note that the rotor flux vector has the same angle in the T-model and the inverse- $\Gamma$  model. The transformation coefficient  $\gamma$  is chosen so it only changes the length of the vector. The dynamic governing equations for the inverse- $\Gamma$ -model in the stationary  $\alpha\beta$  coordinate system are

$$\mathbf{v}_s^s - R_s \mathbf{i}_s^s - L_\sigma \frac{d\mathbf{i}_s^s}{dt} - L_M \frac{d\mathbf{i}_M^s}{dt} = 0 \quad (2.26)$$

$$j\omega_r \psi_R^s - R_R \mathbf{i}_R^s - L_M \frac{d\mathbf{i}_M^s}{dt} = 0 \quad (2.27)$$

where

$$\psi_s^s = L_\sigma \mathbf{i}_s^s + L_M \mathbf{i}_M^s \quad (2.28)$$

$$\boldsymbol{\Psi}_R^s = L_M \mathbf{i}_M^s = L_M (\mathbf{i}_R^s + \mathbf{i}_s^s) \quad (2.29)$$

and  $\mathbf{i}_M^s$  is the magnetizing current. Expressing the rotor current as

$$\mathbf{i}_R^s = \mathbf{i}_M^s - \mathbf{i}_s^s = \frac{\boldsymbol{\Psi}_R^s}{L_M} - \mathbf{i}_s^s \quad (2.30)$$

the change in the rotor flux linkage can be described by using (2.26) as

$$\underbrace{\frac{d\boldsymbol{\Psi}_R^s}{dt}}_{\mathbf{E}_f^s} = \mathbf{v}_s^s - R_s \mathbf{i}_s^s - L_\sigma \frac{d\mathbf{i}_s^s}{dt} \quad (2.31)$$

or by using (2.27) as

$$\underbrace{\frac{d\boldsymbol{\Psi}_R^s}{dt}}_{\mathbf{E}_f^s} = R_R \mathbf{i}_s^s - \left( \frac{R_R}{L_M} - j\omega_r \right) \boldsymbol{\Psi}_R^s \quad (2.32)$$

Equation (2.31) introduces a new symbol for the stator-flux derivative, the flux EMF. It is important to separate the back EMF ( $\mathbf{E}_s^s$ ) and the flux EMF ( $\mathbf{E}_f^s$ ), because they are not equal. Combing (2.31) with (2.32) gives

$$L_\sigma \frac{d\mathbf{i}_s^s}{dt} = \mathbf{v}_s^s - (R_s + R_R) \mathbf{i}_s^s + \underbrace{\left( \frac{R_R}{L_M} - j\omega_r \right) \boldsymbol{\Psi}_R^s}_{\mathbf{E}_s^s} \quad (2.33)$$

which introduces the back EMF. Equation (2.32) and (2.33) describes the dynamic behavior of the electrical system of the IM. The mechanical system can still be described with the same equations that are used for the T-model with two exceptions. The subscript for the rotor current should be changed from r to R and the subscript for the magnetizing inductance should be changed from m to M.

### 2.2.6 The dynamic inverse gamma model for the IM in rotating coordinates

As mentioned before the rotating coordinate system is aligned with the rotor flux vector and the transformation angle is chosen so that the rotor flux becomes real valued,  $\boldsymbol{\Psi}_R = \psi_d + j\psi_q = \Psi_R$ . This is called perfect field orientation and is essential for good performance of the vector controlled system.

The dynamic governing equations for the inverse- $\Gamma$ -model in the rotating  $dq$ -coordinate system can be derived from (2.32) and (2.33) by using (2.12) and this results in

$$\frac{d\Psi_R}{dt} = R_R \mathbf{i}_s - \left( \frac{R_R}{L_M} + j(\omega_l - \omega_r) \right) \Psi_R \quad (2.34)$$

and

$$\mathbf{v}_s = L_\sigma \frac{d\mathbf{i}_s}{dt} + (R_s + R_R + j\omega_l L_\sigma) \mathbf{i}_s - \underbrace{\left( \frac{R_R}{L_M} - j\omega_r \right) \Psi_R}_{\mathbf{E}_s} \quad (2.35)$$

Controlling the IM would not be so complicated if the flux angle was easily measured. Unfortunately, perfect field orientation is not realistic without measuring the flux angle. Since the control method

that will be used is based on sensorless control, this introduces a problem, namely that the flux angle must be estimated. This also means that when the flux angle is being estimated it is not possible to achieve perfect field orientation. However if the accuracy is good enough the field orientation will hardly suffer.

## 2.3 Flux estimators

### 2.3.1 The current model (CM)

The current model is quite straightforward and can be readily derived from the rotor equation in  $dq$ -coordinates. Splitting (2.34) into its real and imaginary parts and assuming perfect field orientation, yields

$$\frac{d\psi_R}{dt} = R_R i_d - \frac{R_R}{L_M} \psi_R \quad (2.36)$$

$$R_R i_q - (\omega_1 - \omega_r) \psi_R = R_R i_q - \omega_2 \psi_R = 0 \quad (2.37)$$

where  $i_d$  is the flux producing current component and  $i_q$  is the torque producing current component. Rewriting (2.37), a relation between the  $i_q$  current and the angular slip speed can be expressed as

$$\omega_2 = \frac{R_R i_q}{\psi_R} \quad (2.38)$$

which can be used to calculate the angular speed of the rotor flux vector, if the rotor speed is known as

$$\omega_1 = \omega_r + \omega_2 = \omega_r + \frac{R_R i_q}{\psi_R} \quad (2.39)$$

The CM flux observer is obtained by integrating (2.36) and (2.39) as

$$\hat{\psi}_R = \int (\hat{R}_R i_d - \frac{\hat{R}_R}{\hat{L}_M} \hat{\psi}_R) dt \quad (2.40)$$

$$\hat{\theta}_1 = \int (\omega_r + \frac{\hat{R}_R i_q}{\hat{\psi}_R}) dt \quad (2.41)$$

where  $\hat{\psi}_R$  is the estimated rotor flux linkage magnitude and  $\hat{\theta}_1$  is the estimated rotor flux vector angle. The current model flux observer then requires measurement of the rotor speed,  $\omega_r$ , in order to estimate the angle. The hat on the parameters are used to indicate that the measured parameters of the IM are used in the estimator. But the real parameters of the IM will differ from the measured parameters. The resistances will increase due to heating and the inductances may decrease due to magnetic saturation. Equation (2.39) is used to estimate the frequency. The current models greatest advantage is that it is the only flux estimator that gives stable operation at low speeds [2].

### 2.3.2 The Statically Compensated Voltage Model (SCVM) in DFO

There are two different ways of implementing the flux estimator, Direct Field Orientation (DFO) or Indirect Field Orientation (IFO). DFO directly estimates the rotor flux space vector directly in the stator-reference frame ( $\alpha\beta$ -coordinates) and does not need any computation of trigonometric

functions. This was a benefit two decades ago when implementation in analog electronics were used. However, today's digital implementation on DSPs can easily handle trigonometric functions and, as will be shown later, IFO adds an extra degree of freedom to the SCVM [2]. The SCVM in DFO is derived from the traditional voltage model which is based on the relation between the rotor flux and the flux-EMF. This relation is described by (2.31) and it can be integrated in order to express the rotor flux as

$$\hat{\psi}_R^s = \int \mathbf{E}_f^s dt \quad (2.42)$$

Inserting the expression of the rotor flux EMF from (2.31) into (2.42), the rotor flux can be expressed as

$$\hat{\psi}_R^s = \int (\mathbf{v}_s^s - \hat{R}_s \mathbf{i}_s^s) dt - \hat{L}_\sigma \mathbf{i}_s^s \quad (2.43)$$

Note that  $\hat{R}_s$  and  $\hat{L}_\sigma$  now are estimates and that they are the critical parameters for the SCVM. The voltage model uses open-loop integration and is therefore marginally stable [3]. To gain stability, modifications need to be made.

### Lowpass filter

The stability of the VM can be improved by using a first order lowpass filter instead of the direct integration according to

$$\hat{\psi}_R^s = \frac{\hat{\mathbf{E}}_f^s}{p + \alpha_v} \quad (2.44)$$

Generally, the Laplace operator is denoted  $s$  but to avoid confusion the slip it will from now on be denoted  $p$ . Assuming perfect parameters where  $\psi_R^s$  is the true rotor flux and inserting  $\hat{\mathbf{E}}_f^s = p\psi_R^s$  into (2.44) gives

$$\hat{\psi}_R^s = \frac{p}{p + \alpha_v} \psi_R^s \quad (2.45)$$

Analyzing this equation during steady-state,  $p = j\omega_1$  yields

$$\hat{\psi}_R^s = \frac{j\omega_1}{j\omega_1 + \alpha_v} \psi_R^s \quad (2.46)$$

This introduces a large error when  $|\omega_1| < \alpha_v$ . However, this error could be reduced if the bandwidth,  $\alpha_v$ , is selected to be proportional to the stator speed [6],  $\omega_1$  according to

$$\alpha_v = \lambda |\omega_1| \quad (2.47)$$

The flux estimator can now be written as

$$\hat{\psi}_R^s = \frac{1}{1 + j\lambda \text{sign}(\omega_1)} \psi_R^s \quad (2.48)$$

While the largest error has been removed, a smaller static error is obtained for all stator frequencies. However, if  $\lambda$  is chosen to be arbitrarily small, the error is limited and  $\psi_R \sim \hat{\psi}_R$  could be assumed. Though this reduces the static error, the system is still considered marginally stable and further modifications are needed. It needs to be mentioned that choosing  $\lambda$  too small will give a poorly damped system.

### Modification: Lowpass filter with compensation gain

To obtain both a small error and a stable system, (2.44) can be modified to be perfectly compensated during steady state operation. The compensation is performed by multiplying the estimator in (2.44) by the inverse of the steady state error introduced by the lowpass filter, described in (2.48) [2]. The flux estimator can now be expressed as

$$\hat{\psi}_R^s = \frac{1 - j\lambda \text{sign}\omega_1}{p + \lambda |\omega_1|} \hat{\mathbf{E}}_f^s \quad (2.49)$$

Where  $\alpha_v$  in (2.44) is changed to  $\lambda |\omega_1|$ . Transforming (2.49) back to the time domain yields

$$\frac{d\hat{\psi}_R^s}{dt} = (1 - j\lambda \text{sign}\omega_1) (\mathbf{v}_s^s - R_s \mathbf{i}_s^s - L_\sigma \frac{d\mathbf{i}_s^s}{dt}) - \lambda |\omega_1| \hat{\psi}_R^s \quad (2.50)$$

This is the Statically Compensated Voltage Model in DFO. Under the assumption that  $\hat{R}_s = R_s$  and  $\hat{L}_\sigma = L_\sigma$ , the estimated flux in steady state will be equal to the real flux.

### 2.3.3 IFO implementation of the SCVM

The SCVM in IFO is obtained by transforming the DFO into the rotating dq-coordinate system. Transforming (2.50) to IFO by using (2.11) gives

$$\frac{d\hat{\psi}_R^s}{dt} + j\hat{\omega}_1 \hat{\psi}_R^s = (1 - j\lambda \text{sign}\omega_1) \underbrace{(\mathbf{v}_s^s - R_s \mathbf{i}_s^s - j\hat{\omega}_1 \hat{L}_\sigma \mathbf{i}_s^s)}_{\hat{e}_d + j\hat{e}_q} - L_\sigma \frac{d\mathbf{i}_s^s}{dt} - \lambda |\omega_1| \hat{\psi}_R^s \quad (2.51)$$

assuming that the current controller is much faster than the flux estimator the stator current derivative can be neglected [2] and furthermore two new voltages are defined as

$$\hat{e}_d = \hat{v}_d - \hat{R}_s \hat{i}_d + \hat{\omega}_1 \hat{L}_\sigma \hat{i}_q \quad (2.52)$$

$$\hat{e}_q = \hat{v}_q - \hat{R}_s \hat{i}_q - \hat{\omega}_1 \hat{L}_\sigma \hat{i}_d \quad (2.53)$$

splitting the real and imaginary parts of (2.51) and assuming perfect field orientation, gives

$$\hat{\psi}_R = \frac{\hat{e}_d + \lambda \text{sign}(\omega_1) \hat{e}_q}{p + \lambda |\omega_1|} \quad (2.54)$$

$$\hat{\omega}_1 = \frac{\hat{e}_q - \lambda \text{sign}(\omega_1) \hat{e}_d}{\hat{\psi}_R} \quad (2.55)$$

As mentioned earlier, the IFO implementation opens up for an extra degree of freedom, without adding additional errors to the estimation. Introducing an extra coefficient,  $\mu$ , in (2.56) according to [6], the flux modulus equation can be rewritten as

$$\hat{\psi}_R = \frac{\mu \hat{e}_d + \lambda \text{sign}(\omega_1) \hat{e}_q}{p + \lambda |\omega_1|} \quad (2.56)$$

Stability analysis of the SCVM has been conducted by both L.Harnefors in [3] and by R.Ottersten in [6]. By choosing  $\mu = -1$  and  $\lambda = \sqrt{2}$  a well damped system is created, with pole placements according to

$$p = - \left( \frac{\lambda}{2} \pm j \sqrt{\mu + \frac{3\lambda^2}{4}} \right) |\omega_r| \quad (2.57)$$

With the recommended parameter selections the poles will then be placed at  $p = -|\omega_r|e^{\pm j\pi/4}$ . The electrical rotor speed can be estimated by combining (2.55) with (2.2) and results in

$$\hat{\omega}_r = \hat{\omega}_1 - \hat{\omega}_2 = \frac{\hat{e}_q - \lambda \text{sign}(\hat{\omega}_1) \hat{e}_d - \hat{R}_R \hat{i}_{sq,ref}}{\hat{\psi}_R} \quad (2.58)$$

For easier implementation in a block diagram (2.56) can be rewritten as

$$\hat{\psi}_R = \frac{1}{p} (\mu \hat{e}_d + \lambda \text{sign}(\hat{\omega}_1) \hat{e}_q - \lambda |\hat{\omega}_1| \hat{\psi}_R) \quad (2.59)$$

### 2.3.4 Parameter sensitivity of the CM and SCVM

To clarify, all parameters with hats are considered as estimates. If it is assumed that  $\theta_1 = \hat{\theta}_1$ , the rotor flux will be perfectly aligned with the  $d$ -axis. If the accuracy of the estimated parameters would be poor, the  $i_d$ -current would "spill over" from the  $d$ -direction into the  $q$ -direction and vice versa. Inspecting (2.39) and (2.76) one notices that the CM is sensitive to the estimations of the rotor resistance and the magnetizing inductance, i.e.  $\hat{R}_R$  and  $\hat{L}_M$ . The result of having estimation errors in these parameters is that the field orientation becomes poor. The rotor resistance of the machine,  $R_R$ , changes when the rotor gets hot and the magnetizing inductance of the machine,  $L_M$ , is affected by magnetic saturation.

In [2] it has been shown that the error angle for the SCVM can be expressed as

$$\tilde{\theta}_1 = \arcsin \left( \frac{\tilde{R}_s i_d}{\omega_1 \psi_R} - \frac{\tilde{L}_\sigma i_q}{\psi_R} \right) \quad (2.60)$$

where this relation is valid for steady state. Furthermore

$$\tilde{\theta}_1 = \hat{\theta}_1 - \theta_1 \quad (2.61)$$

and the same relation is true for  $\tilde{R}_s$  and  $\tilde{L}_\sigma$ . From (2.60) it can be concluded that the SCVM is not stable at  $\omega_1 = 0$  and that it is sensitive to the estimates of  $\hat{R}_s$  and  $\hat{L}_\sigma$ . The stator resistance of the machine,  $R_s$ , can increase by as much as 60 percent and the leakage inductance of the machine,  $L_\sigma$ , varies with at least 15 percent [6].

## 2.4 Current controller derivation

The system to be controlled by the current controller, can be derived by laplace transforming (2.35). The term  $\frac{R_R}{L_M}$  is small compared to  $\omega_r$  so it can be neglected. Furthermore, if perfect field orientation and linear power electronics are assumed, the system can be expressed as

$$\mathbf{i}_s = G_c(p)(\mathbf{v}_s - j\omega_r \boldsymbol{\Psi}_R) \quad (2.62)$$

where  $G_c(p)$  is the transfer function for the stator equation of the IM

$$G_c(p) = \frac{1}{pL_\sigma + R_R + R_s + j\omega_1 L_\sigma} \quad (2.63)$$

The back emf term  $j\omega_r \Psi_R$  and the cross coupling term  $j\omega_1 L_\sigma$  can be removed. Also, an active damping term  $R_a$  can be added. All of this can be achieved by setting the voltage to the machine equal to

$$\mathbf{v}_s = \mathbf{v}_s^* + (j\omega_1 L_\sigma - R_a)\mathbf{i}_s + j\omega_r \Psi_R \quad (2.64)$$

where  $\mathbf{v}_s^*$  is the voltage reference that will be realized by the PI-controller and  $\mathbf{v}_s$  is the voltage that will be "seen" by the IM. The transfer function from  $\mathbf{v}_s^*$  to  $\mathbf{i}_s$  can now be expressed as

$$G_c^*(p) = \frac{\mathbf{i}_s}{\mathbf{v}_s^*} = \frac{1}{pL_\sigma + R_R + R_s + R_a} \quad (2.65)$$

The transfer function is now of order one, which means that a PI-controller, that also is of order one, can be used to eliminate the steady state error. The closed loop system is selected to be [2]

$$G_{cl} = \frac{\alpha_c}{p + \alpha_c}. \quad (2.66)$$

where  $\alpha_c$  is the bandwidth in rad/s. The bandwidth of a first order system is related to the rise time  $t_r$  according to [2]

$$\alpha_c = \frac{\ln 9}{t_r} \quad (2.67)$$

The actual closed loop system is on the form

$$G_{cl} = \frac{F_c(p)G_c(p)}{1 + F_c(p)G_c(p)} \quad (2.68)$$

where  $F_c(p)$  is the PI-controller

$$F_c(p) = k_{pc} + \frac{k_{ic}}{p} \quad (2.69)$$

Combining (2.68) with (2.66) and solving for  $F_c(p)$  results in

$$F_c(p) = k_{pc} + \frac{k_{ic}}{p} = \alpha_c \hat{L}_\sigma + \frac{\alpha_c (\hat{R}_R + \hat{R}_s + R_a)}{p}. \quad (2.70)$$

Where the active damping resistance is selected as

$$R_a = \alpha_c \hat{L}_\sigma - \hat{R}_s - \hat{R}_R \quad (2.71)$$

Active damping,  $R_a$  is added in order to make the system less sensitive to disturbances and parameter errors. The cross coupling term is removed, because otherwise a step in the d-current would affect the q-current and vice versa. The back emf term is removed because this decreases the current control error [2].

A voltage limiter needs to be added in order to consider the rated voltage of the machine and the converter. The voltage limiter limits the length of the voltage vector to rated voltage, if the current controller is asking for a voltage that is too large. The limited voltage reference is selected according to

$$\mathbf{v}_{s,lim} = \begin{cases} \mathbf{v}_s & \text{if } |\mathbf{v}_s| \leq v_{s,max} \\ |v_{s,max}| < \mathbf{v}_s & \text{if } |\mathbf{v}_s| > v_{s,max} \end{cases} \quad (2.72)$$

The addition of a voltage limiter may cause integrator windup. If the reference voltage gets higher than the voltage limit, there will be an error between the limited voltage and the actual voltage. This causes the current to increase slower, which leads to that the integrator integrates too much. When the current reaches the reference the accumulated error will be too large, causing an overshoot. To negate this, a back calculation algorithm can be used. The unlimited voltage reference to the machine can be expressed as

$$\mathbf{v}_s = k_{pc}\mathbf{e} + k_{ic}\mathbf{I} + (j\omega_l L_\sigma - R_a)\mathbf{i}_s + j\omega_r \boldsymbol{\Psi}_R \quad (2.73)$$

where  $\mathbf{e} = \mathbf{i}_{s,ref} - \mathbf{i}_s$  is the current error and  $\mathbf{I}$  is the integrator state variable. A new error is now introduced, so that the current controller now puts out a limited voltage as

$$\mathbf{v}_{s,lim} = k_{pc}\bar{\mathbf{e}} + k_{ic}\mathbf{I} + (j\omega_l L_\sigma - R_a)\mathbf{i}_s + j\omega_r \boldsymbol{\Psi}_R \quad (2.74)$$

where  $\mathbf{v}_{s,lim}$  is the limited voltage and  $\bar{\mathbf{e}}$  is the new error that needs to be fed to the integrator in order to avoid an overshoot. Subtracting the unlimited voltage from the limited and solving for the new error results in

$$\bar{\mathbf{e}} = \mathbf{e} + \frac{\bar{\mathbf{v}}_{s,lim} - \mathbf{v}_s}{k_{pc}} \quad (2.75)$$

If this error signal is fed to the integrator, the windup of the integrator can be avoided.

### 2.4.1 Current reference calculation and current limiter

The current references can be derived from the torque and rotor equations. Assuming steady state, perfect field orientation and separating the real part of (2.34), gives

$$i_{sd,ref} = \frac{\Psi_{R,ref}}{\hat{L}_M} \quad (2.76)$$

where  $\Psi_{R,ref}$  is the desired flux reference. The flux level is said to be indirectly controlled by  $i_{sd,ref}$ , through the magnetizing inductance. Assuming perfect field orientation for (2.20) gives

$$i_{sq,ref} = \frac{2T_{e,ref}}{3n_p \hat{\Psi}_R} \quad (2.77)$$

A current limiter needs to be added in order to consider the rated current of the IM. The current limiter is working on the absolute value of the current vector and can be derived by realizing that  $I_{s,rated}^2 = |\mathbf{i}_s|^2 = i_{sd}^2 + i_{sq}^2$ . The limiter should only limit the  $i_q$  current and this puts out a limited torque reference  $T_{e,lim}$ . The flux producing component  $i_d$  will either be creating rated flux or lower than that when the machine enters the field weakening region. This means that if  $i_d$  decreases,  $i_q$  can increase until the current vector has again reached its limit,  $I_{s,rated}$ .

## 2.5 Field weakening

The magnetic field needs to be weakened in order to reach higher speeds than rated speed, if the stator voltage should be limited to rated voltage. This can be realized by inspecting (2.35). In order to

## Chapter 2. Technical background

increase the rotor speed,  $\omega_r$ , the stator voltage magnitude,  $|\mathbf{v}_s|$ , needs to increase. As a consequence of increasing the rotor speed, the stator speed,  $\omega_1$ , increases. If the flux magnitude,  $|\Psi_R|$ , is constant it means that at a certain  $\omega_1$  the voltage,  $|\mathbf{v}_s| = v_{s,lim}$ . This is caused by the back emf term,  $j\omega_r\Psi_R$ . If the voltage has reached its limit, the only way to increase the speed is then to decrease the flux magnitude. The machine is then said to be operating in the field weakening region. As the magnetic flux decreases, so does the torque. Thus when an IM is operating above nominal speed will produce less torque, as can be seen from (2.20).

The used fieldweakening algorithm can be found in [4] as

$$\Psi_{R,ref} = \int k(v_{base}^2 - v_{d,ref}^2 - v_{q,ref}^2) dt \Big|_{\Psi_{R,min}}^{\Psi_{R,rated}} \quad (2.78)$$

Where  $v_{d,ref}$  and  $v_{q,ref}$  are the ideal references from the current controller. The algorithm weakens the field when the voltage reference from the current controller approaches the maximum voltage,  $v_{base}$ , that the power electronics can put out. This margin is needed in order to be able to control the current. It is important to limit the algorithm so that rated flux is achieved at nominal speed and speeds below. It is also recommended to have a lower limit so that the machine cannot be demagnetized [4]. By choosing  $k$  as

$$k = \frac{\alpha_f \hat{L}_M}{2\omega_f \hat{L}_\sigma v_{s,rated}} \quad (2.79)$$

a constant bandwidth,  $\alpha_f$  of the field weakening algorithm is obtained [4]. The frequency  $\omega_f$  should be chosen according to

$$\omega_f = \begin{cases} \omega_{1,rated} & \text{if } |\omega_1| \leq \omega_{1,rated} \\ |\omega_1| & \text{if } |\omega_1| > \omega_{1,rated} \end{cases} \quad (2.80)$$

where  $\omega_{1,rated}$  is the rated synchronous speed of the machine.

## 2.6 Speed controller derivation

The entire system in figure 2.1 is cascade controlled, where the speed controller is part of the outer and slower loop with respect to the current controller. The speed controller is designed according to the same principles as the current controller. It also contains active damping and a limit to the torque reference that comes from the limit in current, that is set inside the current reference calculation block. It turns out as

$$F_\omega(s) = k_{p\omega} + \frac{k_{i\omega}}{s} = \alpha_\omega \hat{J} + \frac{\alpha_\omega(\hat{b} + B_a)}{s}. \quad (2.81)$$

where  $\hat{J}$  is the inertia of the total mechanical system in  $kgm^2$ ,  $\hat{b}$  is the viscous damping coefficient of the total mechanical system in  $kgm^2/s$  and the active damping is selected as  $B_a = \alpha_\omega \hat{J} - \hat{b}$ , to give better load disturbance rejection [2].

## Chapter 3

# MATLAB/Simulink implementation of the models

Before implementing the control system in a microcontroller it is convenient to do simulations in order to test the performance of the control strategy. It is also useful to know how sensitive the system will be to parameter errors, especially in  $\hat{R}_s$ , and  $\hat{L}_\sigma$ . This is however not in the scope of the report because the previous master thesis at AROS electronics evaluated the effects of parameter errors [1].

### 3.1 Overview of the matlab model

The induction machine is controlled by a cascaded speed and current controller as shown in Figure 3.1. First, a speed reference is given to the speed controller that puts out a torque reference to the current reference calculation block. This block recalculates the torque reference into a current reference and passes it to the current controller. The current controller then puts out the required voltage to the machine in order to reach the current reference. In order to simplify the model, it is assumed that the voltage is created by an ideal power electronic circuit. Another part of the model is the field weakening algorithm that reduces the magnetic field in the machine, allowing it to reach speeds above rated speed. This algorithm is set up to have constant bandwidth, equal to the speed controller bandwidth. Finally, the statically compensated voltage model is used to estimate  $\hat{\omega}_1$ ,  $\hat{\omega}_r$ ,  $\hat{\psi}_R$  and  $\hat{\theta}_1$ . When using a cascade controlled system, the inner loops must be faster than the outer loops in order for it to work. The most inner loop is the converter that should realize the voltage references, then comes the current controller, SCVM, speed controller and field weakening algorithm. A complete overview of the matlab model can be found in Appendix A.4. In the following chapters, the blocks in Figure 3.1 are described.

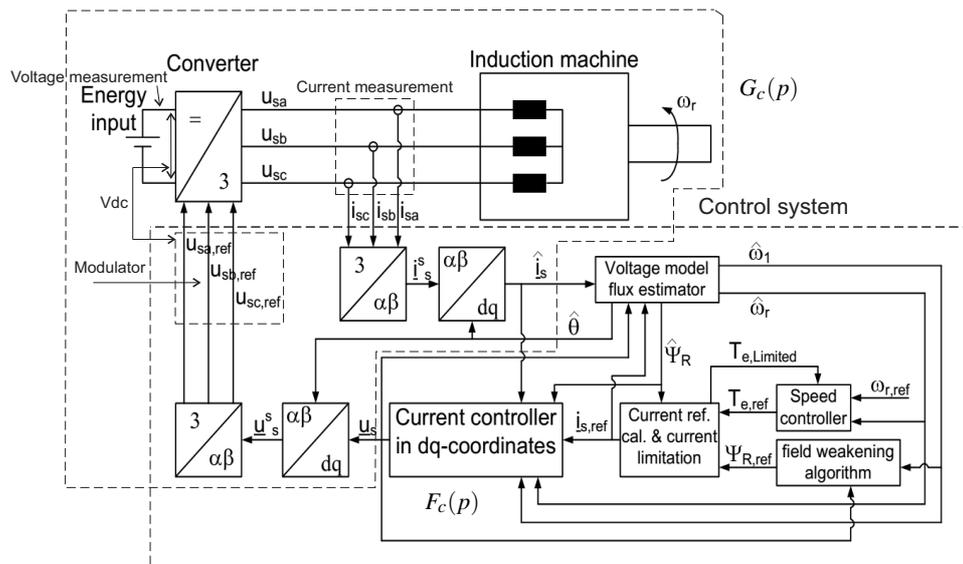


Fig. 3.1 overview of the system, courtesy of Stefan Lundberg

### 3.2 Machine Model

In order to implement the machine model in the MATLAB/Simulink environment it needs to be written on state space form. Combining equations (2.13) - (2.16), splitting the real and imaginary parts and assuming a short circuited rotor,  $\mathbf{v}_r = 0$ , yield

$$\underbrace{\begin{bmatrix} v_{s\alpha} \\ v_{s\beta} \\ 0 \\ 0 \end{bmatrix}}_{\mathbf{u}} = \underbrace{\begin{bmatrix} R_s & 0 & 0 & 0 \\ 0 & R_s & 0 & 0 \\ 0 & \omega_r L_m & R_r & \omega_r L_r \\ -\omega_r L_m & 0 & -\omega_r L_r & R_r \end{bmatrix}}_{\mathbf{R}} \underbrace{\begin{bmatrix} i_{s\alpha} \\ i_{s\beta} \\ i_{r\alpha} \\ i_{r\beta} \end{bmatrix}}_{\mathbf{x}} + \underbrace{\begin{bmatrix} L_s & 0 & L_m & 0 \\ 0 & L_s & 0 & L_m \\ L_m & 0 & L_r & 0 \\ 0 & L_m & 0 & L_r \end{bmatrix}}_{\mathbf{L}} \underbrace{\begin{bmatrix} \frac{di_{s\alpha}}{dt} \\ \frac{di_{s\beta}}{dt} \\ \frac{di_{r\alpha}}{dt} \\ \frac{di_{r\beta}}{dt} \end{bmatrix}}_{\dot{\mathbf{x}}} \quad (3.1)$$

The equations are then rearranged and written on state space form, where the states can be obtained as

$$\dot{\mathbf{x}} = -\mathbf{L}^{-1}\mathbf{R}\mathbf{x} + \mathbf{L}^{-1}\mathbf{u} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u} \quad (3.2)$$

and (2.18) - (2.21) results in

$$\dot{\omega}_r = -\frac{b}{J}\omega_r + \frac{n_p}{J}T_e - \frac{n_p}{J}T_{L,extra} \quad (3.3)$$

$$\dot{\theta}_r = \omega_r \quad (3.4)$$

This gives that there are six states for the model. The four currents in (3.1), the electrical speed of the rotor and its angle. The rotor flux in  $\alpha\beta$  can be expressed from (2.16) by splitting it into its real and imaginary parts and the magnitude and angle of the rotor flux can now be calculated. Now these equations can be implemented in a S-function block in MATLAB/Simulink. The inputs to the model are the stator voltage in  $\alpha\beta$  and the extra load torque  $T_{L,extra}$ , that is the part of the load torque that is not described by the viscous damping. The outputs from the model are the stator currents, rotor currents, electrical speed, rotor position, electrodynamical torque, rotor flux magnitude and the rotor flux angle. The implemented S-function can be found in Appendix A.1.1.

### 3.3 The current controller block

The block diagram of the current controller is constructed with help from (2.70) and it is shown in figure 3.2. Since active damping and the decoupling of the  $i_q$  and  $i_d$  currents were not implemented in the DSP, the gain blocks  $R_a$  and  $j\hat{L}_\sigma$  were disconnected. The lim block should limit the voltage to what the power electronics can put out. The maximum voltage that can be generated by the power electronics is depending on the DC-link voltage of the converter and the modulation used. Space vector modulation is used in the DSP, which gives that the maximum length of the voltage vector is limited to [2]

$$|v_s^s|_{max} = \frac{2V_{dc}}{3} \quad (3.5)$$

where  $V_{dc}$  is the DC-link voltage of the converter and  $|v_s^s|_{max}$  is the maximum length of the voltage vector that can be generated. The converter has a DC-link voltage of 540 V and the maximum

duty cycle is 95 percent. But in order to prevent overmodulation a factor of 0.55 is used instead of  $\frac{2}{3}$ . This means that the maximum value of the phase voltage that can be generated is  $|v_s^*|_{max} = 282 V_{peak}$ . This gives a RMS-value of about 200 V. The machine that was evaluated in [4] had very similar parameters and ratings to the one that was evaluated in this work. The bandwidth for the current controller was for that reason, put to  $\alpha_c = 1500 rad/s$ , which is the bandwidth used in [4]. The measured motor parameters in table A.4 in Appendix A.3 were used, together with (2.63) and resulted in  $K_{pc} = 27$  and  $K_{ic} = 6400$ .

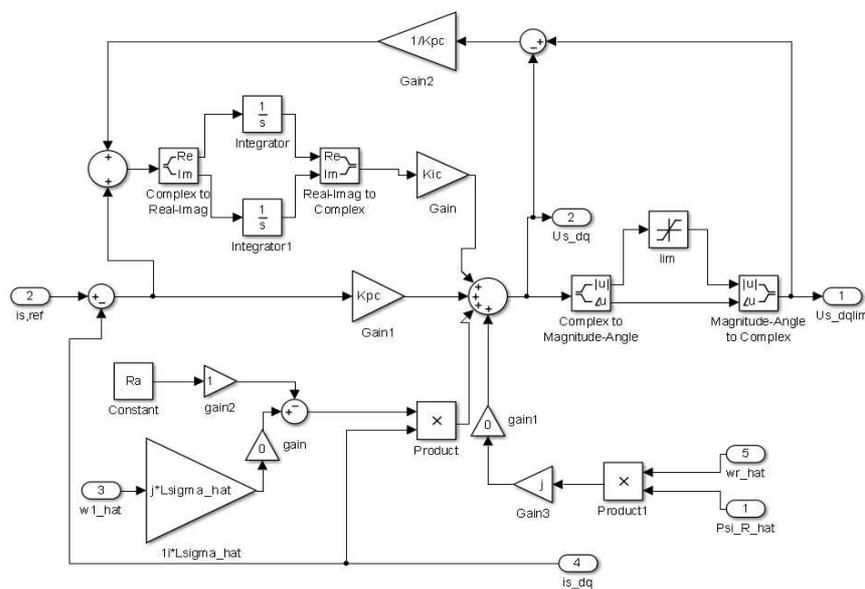


Figure 3.2: Simulink implementation of the current controller block

### 3.4 The reference current calculation block

The current reference calculation block is derived from (2.76) - (2.77). The current reference block has a current limiter that is set to 9 A which is a little higher than the rated peak current of 6.7 A of the machine, (the ratings of the machine that was used can be found in Appendix A.2). The limiter is chosen like this because during the implementation it was observed that the system performed better during large speed steps with a slightly higher permissible current. The saturation block for  $i_d$  is put to limit the current between 0 and  $I_{srated}$ . The current reference block can be seen in Figure 3.3.

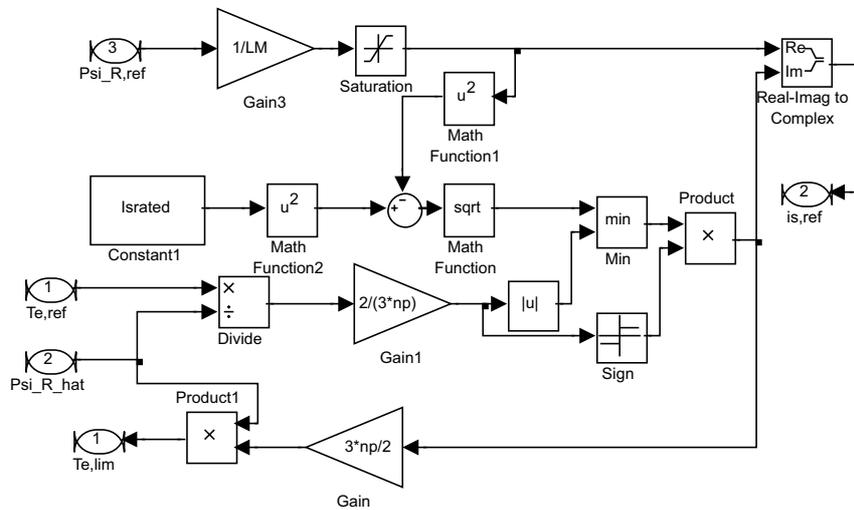


Figure 3.3: Simulink implementation of the reference current calculation block, with current limiter.

The current limiter works on the absolute value of the current, which means that it will limit both negative and positive currents. In order to change the current limit, a new peak value can be given to the variable called  $I_{srated}$  in the matlab code. The code can be found in Appendix A.1.2.

### 3.5 The statically compensated voltage model block

The SCVM block diagram is constructed from (2.55) - (2.59) and is shown in Figure 3.4. There are three things that are different, compared to the derived model in (2.55) - (2.59). First of all, the switch block needs to be there in order for the simulation to work with  $\mu = -1$ . Because the term  $\mu \hat{e}_d$  in (2.59) is very negative in the beginning when  $\omega_1 = 0$ , the estimated flux will become very negative fast and the simulation will hang up. A negative flux does not even correspond to anything physical, it is just a result of the added  $\mu$  term. The switch block makes the  $\mu \hat{e}_d$  term appear after  $|\omega_r|$  has increased to 1 rad/s. Until  $|\omega_r|$  reaches this speed, it is only the  $\hat{e}_d$  term that is used. This solves the problem. Secondly, in order to break the algebraic loop that is created,  $\omega_r$  can be lowpass filtered with a bandwidth equal to the current controller, that is 1500 rad/s [3]. During the implementation it was noted that having a bandwidth of 5000 rad/s worked better because it made the estimated speed to oscillate less. Thirdly a small term needs to be added to the estimated flux magnitude output, otherwise there will be a division by zero before the machine is magnetized in the beginning of the simulation. In this work  $10^{-5}$  is added to prevent this. The selection of the values  $\mu = -1$  and  $\lambda = \sqrt{2}$  is explained in the theory chapter of the SCVM.

### 3.6. The speed controller block

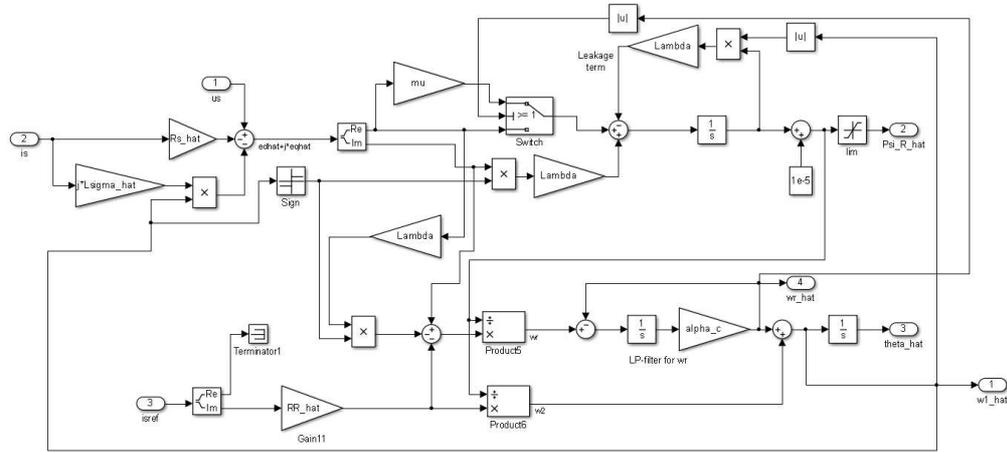


Figure 3.4: Simulink implementation of the statically compensated voltage model block.

### 3.6 The speed controller block

The speed controller is constructed with help from (2.81) and is shown in Figure 3.5. The DSP implementation of active damping was however not successful so the gain block  $B_a$  is put to zero. The gain parameters of the controller were calculated from (2.81). The total inertia of the system was obtained from data sheets and can be found in Appendix A.2 as  $J = 0.004 \text{ kgm}^2$ . The friction coefficient of the IM was measured to  $b = 0.003$ . The details of the friction measurement can be found in the implementation chapter. In [4] a bandwidth of 30 rad/s was recommended and this gave that  $K_{p\omega} = 0.175$  and  $K_{i\omega} = 1.5$ .

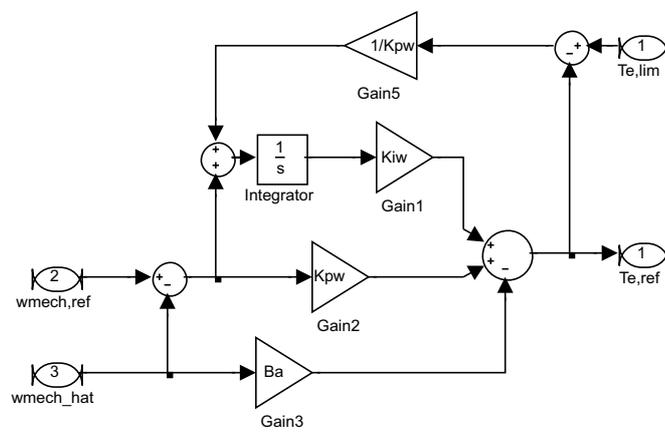


Figure 3.5: Simulink implementation of the speed controller block diagram with active damping and back calculation to prevent integrator windup

### 3.7 The field weakening block

The field weakening block is constructed from (2.78) - (2.80) and the implementation for simulink is shown in figure 3.6. The lower limit of the saturation blocks is put to  $\omega_{1, rated}$  and the higher limit is put to infinity. The limited integrator has its upper saturation limit put slightly above rated flux, 0.72 Wb, and the lower limit is put to 0.4 Wb. These limits prevent the machine from getting either over or undermagnetized and can be found in [4].

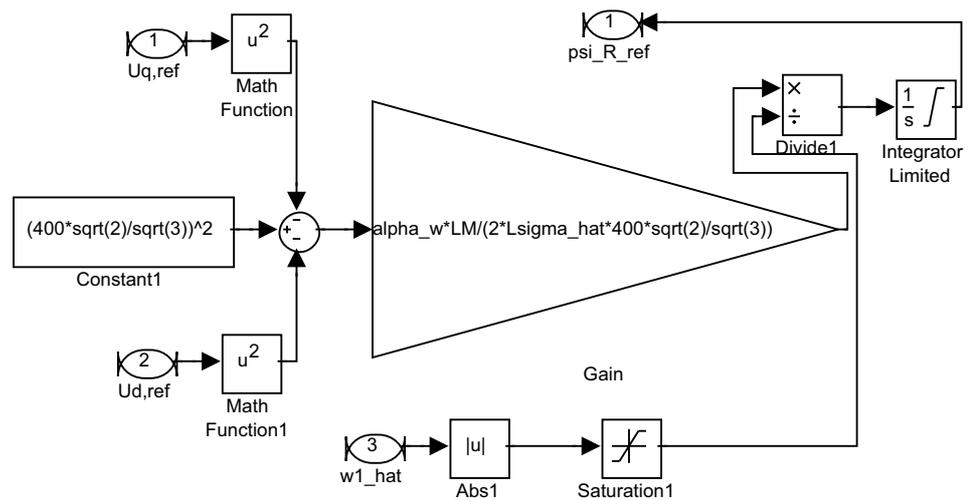


Figure 3.6: field weakening block

$v_{base}$  is selected as the maximum peak voltage that the power electronics can put out. In [4] it is recommended that the bandwidth  $\alpha_f$  should be equal to  $\alpha_\omega$ . In Figure 3.6, the speed controller bandwidth is called  $\alpha_w$ .

*Chapter 3. MATLAB/Simulink implementation of the models*

## Chapter 4

# Test setup and controller implementation

### 4.1 Test bench

The motor bench that the motor was used in was rated for continuous speeds of 3000 RPM. The induction machine (1) was mounted in the motor bench according to figure 4.2. On one side of the machine, an encoder (2) was mounted and on the other side, the rotor shaft was connected to an axial coupling (3). Between the two axial couplings, a torque sensor (5) was installed which is connected to a Panasonic MSMA402A1G AC servo motor (4).

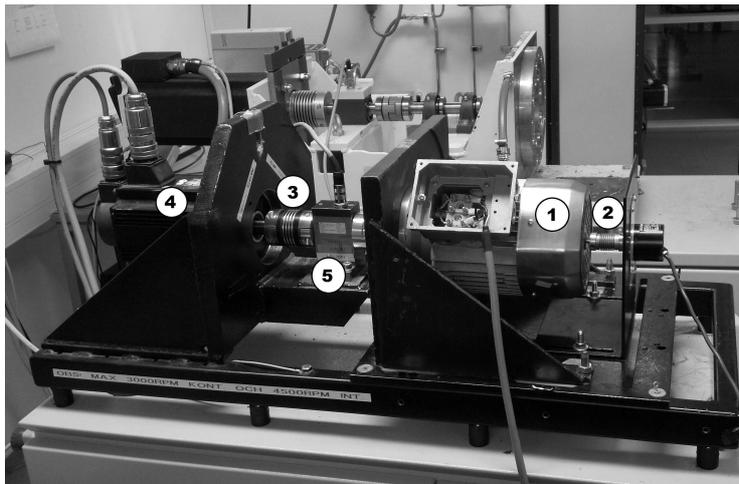


Figure 4.1: Photo of the experimental setup. 1. Induction machine 2. Encoder 3. One of the axial couplings 4. The AC servo machine 5. Torque sensor

## Chapter 4. Test setup and controller implementation

The induction machine used in the test bench is from the Bonfiglioli Group and the type is a BN80C. It is connected in delta. A no-load and locked-rotor test were performed to acquire the circuit parameters and these can be found in appendix A.3.

The servo is operated through a control panel that can either be put in speed control or torque control mode. Since it was of interest to see which loads the IM could handle, torque mode was used. The maximum torque production for the servo motor is 37.9 Nm and the maximum speed is 4500 RPM.

The converter is of standard design and uses a diode bridge rectifier, which is a 36MT120 from International Rectifier rated 35A continuously and 1200 V peak voltage. The DC link voltage is operating at 540 V and the inverter on the control board is limited to a duty cycle of 95 percent. Furthermore, the max output is limited in the DSP to 55 percent of the DC-link voltage. This value is set to prevent overmodulation from the converter. Using the duty cycle percentage and the maximum voltage output limit, the maximum voltage supplying the IM is  $540 * 0.95 * 0.55 = 283.65$  V peak. This gives a RMS value of about 200 V. Only the phase currents in the inverter and DC-link voltage is measured. Moreover, the switching frequency is 4 kHz.

### 4.1.1 Friction measurement of the IM

The friction of the IM was measured in a simple and straightforward way. The servo was put in speed control mode and it was then used to drive the motor at various operating points, namely rated speed, half rated speed and a third of the rated speed. The torque was then read from the torque sensor and equation (2.21) gives that  $b = 0.003$ , but because of the low resolution of the torque sensor the friction measurement is probably not that accurate. Also, with this method it is only the friction of the IM that is obtained. It is assumed that the friction of the servo motor and the torque sensor is small, compared to the friction of the induction machine.

### 4.1.2 Startup procedure

At first the torque of the AC servo machine is set to the reference value. The counteracting torque from the servo machine is not produced until the the induction machine starts to rotate. Before that happens, a brake is applied on the servo. How the torque response from the servo motor behaves is not known but it will not match the ideal constant torque used in the simulations. In order to start the drive system, the boolean variables responsible for the activation of the PWM switching and the current controller is set. This causes the machine to be magnetized. After a second or two, a manual speed step is set that activates the speed controller and the estimator and the whole drive system is initiated. In other words, this manual speed step is set after the machine has been magnetized.

## 4.2 Implementation of the controller structure

### 4.2.1 Fixed point DSP

The control system was implemented in a fixed point DSP, capable of handling 32-bit variables. It is operating at a switching frequency of 4kHz. The largest issue when it comes to programming in fixed point DSPs is that decimal numbers can not be used. This means that all values must be scaled with an arbitrarily chosen constant in order for the implementation to succeed. For example, the integer

#### 4.2. Implementation of the controller structure

0.4999 will be seen as 0 for a fixed point DSP. To keep the resolution as high as possible, this integer could be scaled with a constant of 1000. The new value of the integer would then be 4999 and all the information is kept. However, a too large scaling constants might result in a overflow because of the limitations imposed by the 16- and 32-bit limit. The limitations for the 16-and 32-bit integers used in the DSP can be found in Table 4.1. The scaling constants used in the implementation were times 100 for current, times 10 for voltage, times 1000 for flux, times 10000 for inductance and times 100 for speed in Hz.

Unfortunately the data acquisition logging tool could not handle variables larger than 16-bits. A simple workaround was simply to always use 32-bit variables because this made sure that no overflow occurred and in the end the 32-bit variables were rescaled to 16-bit variables that could be read from the PC interface. This means that the code is not really optimized, in general it is also written as to ease the comprehension of the code. In all, 10 variables could be read and plotted at the same times. Another limitation that the logging tool imposed was that the sample frequency was quite low. This depended on how many variables that was plotted and read out, but normally the sample rate seemed to be about 5 samples per second. This means that the resolution of the transient behavior will be poor and conclusions concerning current- and voltage step responses can not be drawn.

Table 4.1: Maximum values for different integers.

| Integer type | Signed/unsigned | Max value                 |
|--------------|-----------------|---------------------------|
| 16-bit       | Unsigned        | 65,535                    |
| 16-bit       | Signed          | -32,768 to 32,767         |
| 32-bit       | Unsigned        | 4,294,967,295             |
| 32-bit       | Signed          | -2147483648 to 2147483647 |

#### 4.2.2 Software implementation

Aros Electronics already had written code modules that were used for controlling PMSMs. The speed and current controllers were already in place, and the PWM algorithm was already written. Due to company secrecy, the exact function of the software can not be explained here but a principle flow chart could be seen in Figure 4.2. The PWM algorithm as mentioned previously uses space vector modulation but with power invariant transformation. Since the simulations were done with amplitude invariant transformation, the code was changed into amplitude invariant transformation. There was also an encoder module that were used. Finally, there was also a module with Volt/Hz control that could be used for running the motor. The current model algorithm was first implemented and the SCVM could then be executed along the current model in order to verify if the estimated speeds were equal to the true speeds, and also if the estimated flux was equal to the rated flux of the machine. One important part to mention is that neither active damping or decoupling of the  $d$ - and  $q$ -currents were used. The speed controller directly puts out a current reference to the current controller. In the simulations however, the speed controller puts out a torque reference first, but this is simply not necessary and can be avoided in order to simplify the code.



# Chapter 5

## Results

The results acquired in the simulation and the motor bench is compared to verify the model and to check if the system is working as it should. Both the CM and the VM was executed with the controller parameters presented in Table 5.1 and the measured machine parameters from Table A.4. This corresponds to a bandwidth of the current controller of 1500 rad/s and a bandwidth for the speed controller of about 30 rad/s.

Table 5.1: Controller parameters for the speed and current controller

| Variable        | Value           |
|-----------------|-----------------|
| $K_{pc}$        | 27              |
| $K_{ic}$        | 6400            |
| $K_{p\omega}$   | 0.175           |
| $K_{i\omega}$   | 1.5             |
| $B_a$           | 0               |
| $R_a$           | 0               |
| $\lambda$       | $\sqrt{2}$      |
| $\mu$           | -1              |
| $\alpha_f$      | $\alpha_\omega$ |
| $I_{srated}$    | 9A              |
| $ v_s^s _{max}$ | 282V            |
| $ v_s^s _{fw}$  | 325V            |

### 5.1 Drive system using the current model flux observer

In this section, the performance of the drive system using the CM flux observer is evaluated. The implemented control system is the same as shown in Figure 3.1 with the difference that the CM is used as a flux observer instead of the SCVM. The CM is implemented at first, because it is easier to implement compared to the SCVM. After the CM has been implemented, the SCVM estimations can then be executed together with the CM. It is then possible to see if the SCVM estimates the speed correctly. If the SCVM estimates the speeds correctly, it would indicate that the flux observer is working as it should.

## Chapter 5. Results

The setup of the the CM drive system is tested at nominal torque and speed. Figure 5.1 shows the mechanical rotorspeed and Figure 5.2 shows the estimated rotor flux, measured currents and voltage references in  $dq$ -coordinates. The solid line in the voltage graph is the  $U_q$  voltage and the dashed line is the  $U_d$  voltage. During the measurement, data was sampled at 5.024 samples per second. This is considered much too low for detecting the step response and transient behaviour in the current and voltage measurements.

At  $t=2.5$  seconds, the flux reference is stepped up to 0.5 Wb. The speed reference step is applied at  $t=4$  seconds, with a value of the nominal speed of 1400 RPM. The servo motor is applying a constant braking torque of 7.5 Nm. At  $t=11$  seconds the speed reference is set to 0 RPM. The machine then deaccelerates down 0 RPM and a small negative speed is measured at  $t=11.8$  seconds. The negative speed is probably caused by the servo motors torque response which continues to apply a counteracting torque some time after the machine reaches 0 RPM.

With  $\alpha_\omega = 30$  rad/s, the rise time should be equal to 73 ms. Instead, the rise time is equal to 630 ms. This might be caused by the constant braking torque from the servo of 7.5 Nm that the integrator needs to integrate up when the machine starts. Because there is no active damping, the integrator is weak and it takes more time. Again it needs to be mentioned that the voltages are actually the references from the current controller. It is then simply assumed that the power electronics realize these voltage references to the machine.

Comparing the current references with their respective measurements, it can be seen that the current controllers are almost working as they should. One problem is that the  $i_d$  current does not reach its reference until after the speed step (which causes the step in  $i_q$  current), has been given. The reason for this, might have something to do with that the active damping and cross coupling terms are missing. Secondly, it can be seen that there is a small overshoot in the  $i_q$  current which is probably caused by the missing active damping component.

5.1. Drive system using the current model flux observer

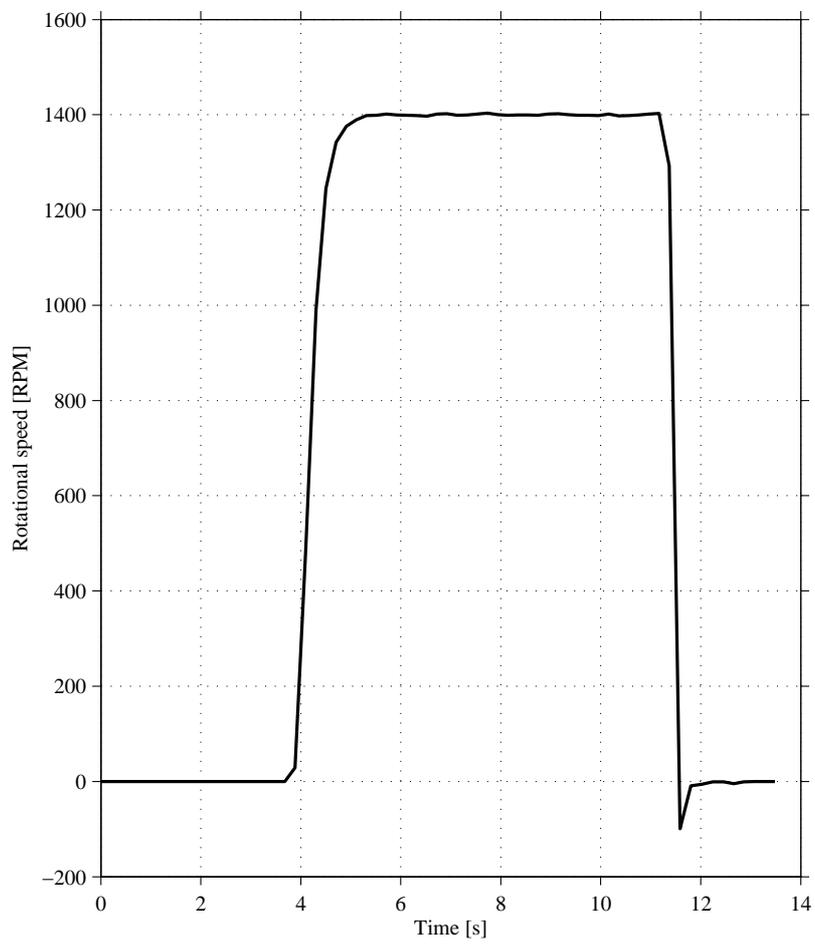


Figure 5.1: Measured speed step response up to nominal speed of 1400 RPM with nominal load torque of 7.5 Nm for the implemented drive system using the CM.

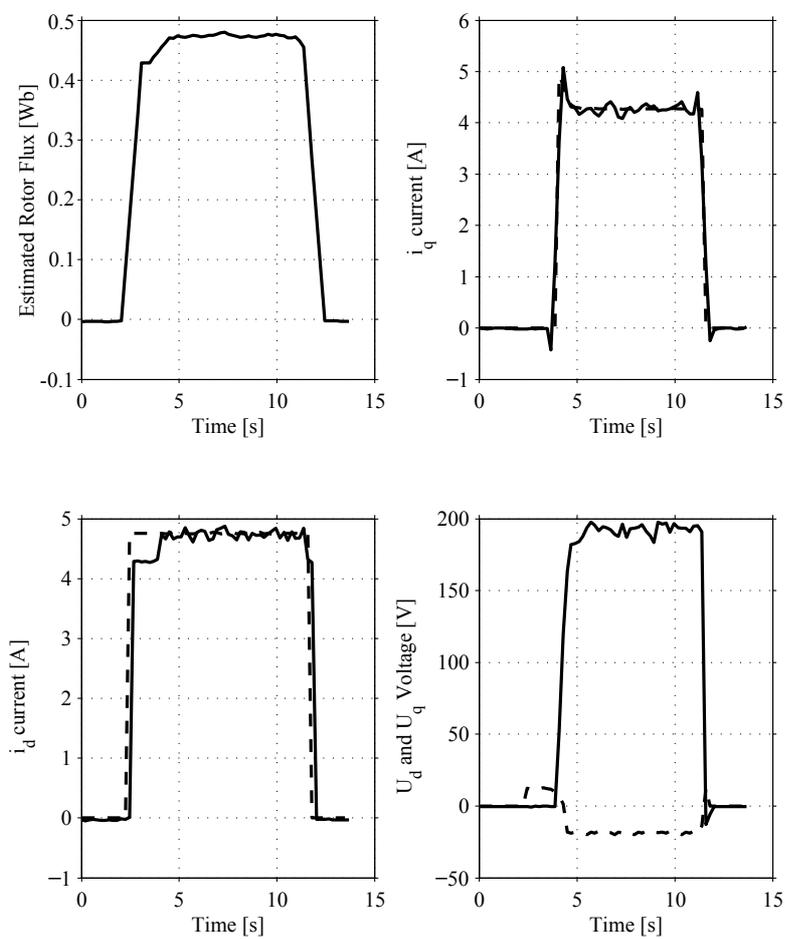


Figure 5.2: Measured speed reference step up to nominal speed of 1400 RPM with nominal load torque of 7.5Nm for the implemented drive system using the CM. In the voltage plot, the solid line is the  $U_q$  reference voltage and the dashed line is the  $U_d$  reference voltage. In the  $i_d$  and  $i_q$  plots, the dashed line is the current reference and the solid line is the measured current.

## 5.2 Drive system using the SCVM flux observer

In this section the performance of the implemented drive system with the SCVM flux observer is evaluated and compared with both simulations and the motor bench implementation of the current model flux observer. The implemented control system is the same as the one shown in Figure 3.1, with the parameters from Table 5.1 and Table A.4.

### 5.2.1 Nominal speed and torque

This measurement was performed with one speed reference step up to nominal speed at 4 s and a step down to 0 RPM at 13 s, with a load torque of 7.5 Nm. Figure 5.3 presents the estimated and measured speed from the motor bench measurements, while Figure 5.4 presents the simulated values. The number of samples per second for the measurements is 4.94, which is too small for detecting current step response and transient behaviour for the measured values. The simulated speed step response is not a first order response. This is believed to be caused by the torque step that is applied in the simulations. This differs from the load that the servo represents in that the servo is applying a braking torque that the integrator needs to integrate up when the IM starts.

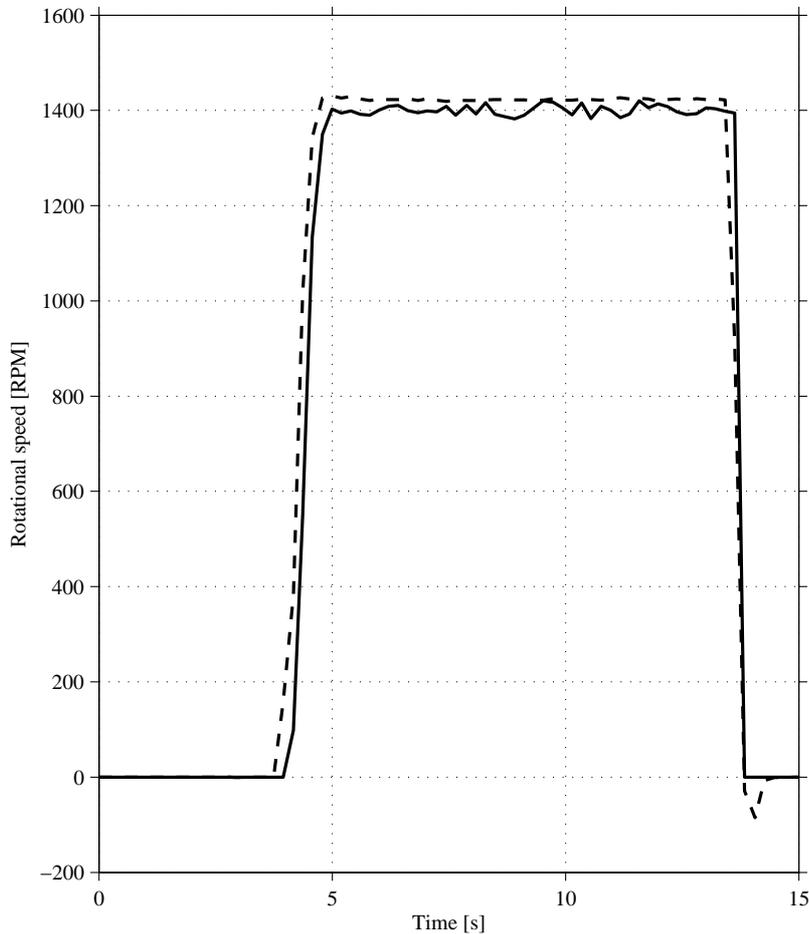


Figure 5.3: Implementation results for a speed reference step to 1400 RPM at 4 s and a step down to 0 RPM at 13 s, with a load torque of 7.5 Nm. Solid line is estimated mechanical rotor speed and dashed line is measured speed.

In the simulation, the speed reference is reached in around 0.5 seconds while the measured speed reaches its reference at around 1 seconds. This difference can depend on several factors. One reason for this difference could be the torque response from the AC servo motor. Since the servo motor is applying a brake until the IM starts to rotate, the AC servo motor is already applying a torque. When the IM starts to rotate the servo will release its brake until the torque reference is reached. This will cause a different load torque compared to the constant load torque used in the simulations. The load torque in the simulation is applied and reaches nominal torque exactly when the speed step is applied. This will cause a higher load torque for the motor bench implementation during some time.

Comparing Figure 5.5 and Figure 5.6 shows that the estimated rotor flux and the voltages are virtually the same. The largest difference can be found in the  $i_q$  currents, where the simulation puts out slightly more current than the implementation. The  $i_q$  current in the simulation reaches 5.2 A while the  $i_q$  current in the implementation reaches about 4.8 A. This indicates that the steady state load torque of the motor bench implementation is less than the load torque simulation. The explanation for this might be that the servo motor is not actually following its reference value and puts out less load torque than what is expected. The fast oscillation peaks in Figure 5.6 are too fast

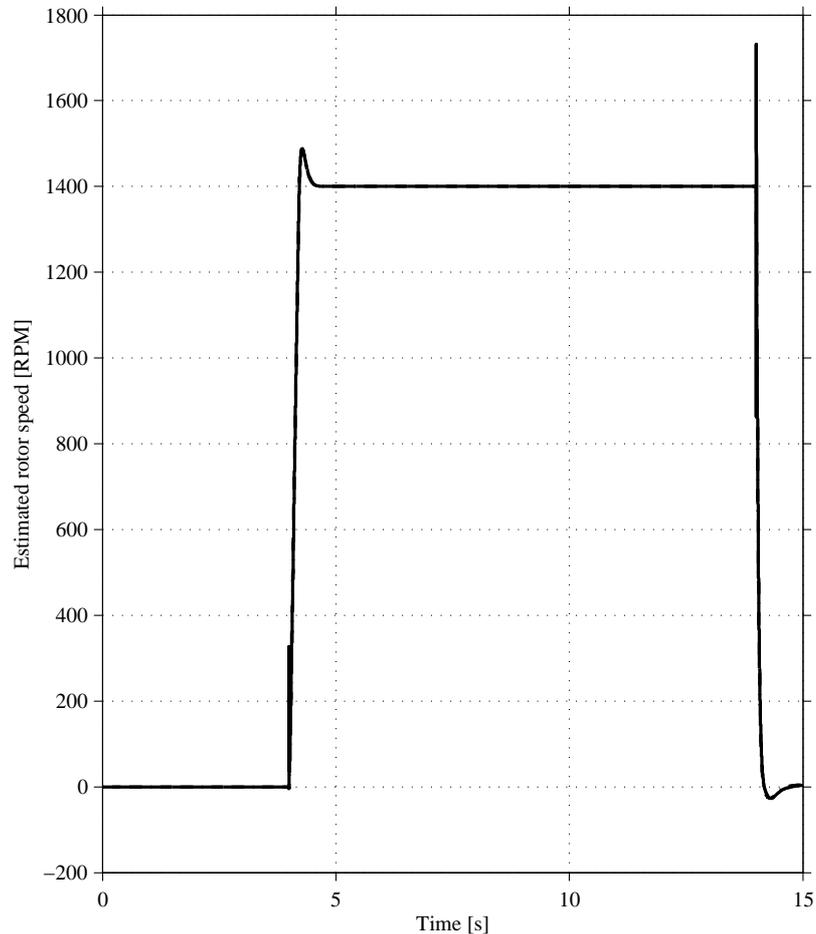


Figure 5.4: Simulation results for a speed reference step to 1400 RPM at 4 s and a step down at 14 s, with a constant load torque of 7.5 Nm for the time the speed reference is greater than 0 RPM. The solid line is the estimated mechanical rotor speed. The actual rotor speed is so close to this estimated speed that they are virtually identical and therefore it cannot be seen.

to be seen in the measurements. Also, the simulated flux overshoots in the simulation due to the neglected current derivative in (2.51). This is not seen in the measurements because the estimator is locked, until the speed step is set. This is done in order to prevent the estimator from "drifting away" in the time window between the magnetization and the speed step.

The same analysis that was done for the current controllers in the CM drive system chapter 5.1 can also be made here.

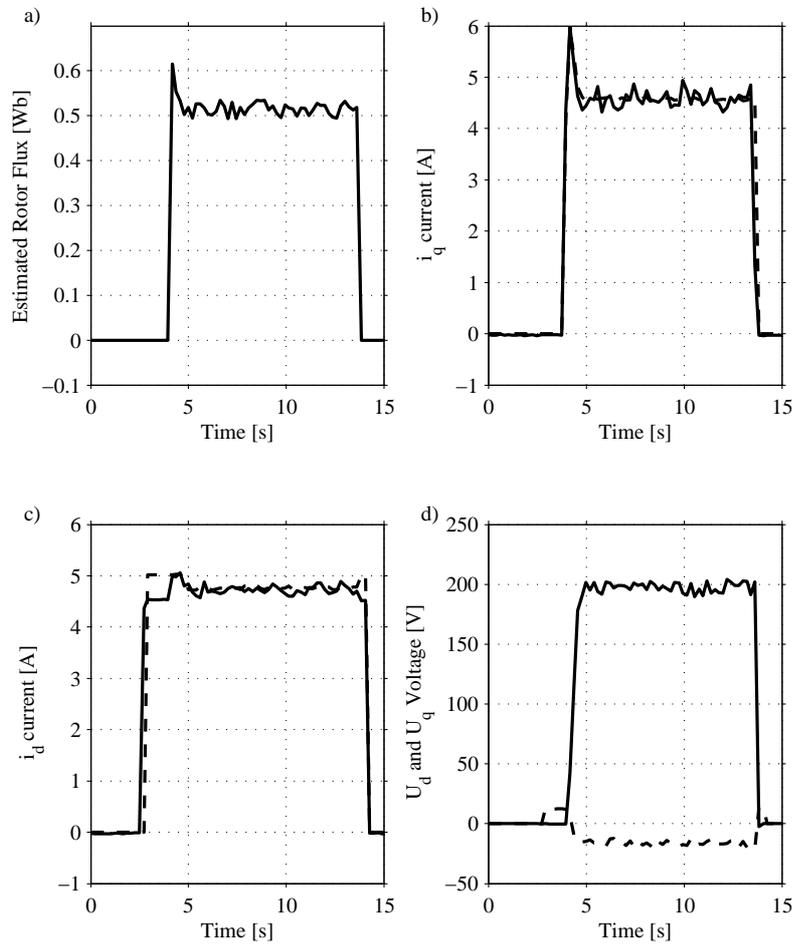


Figure 5.5: Implementation results for a speed step to 1400 RPM with a torque of 7.5 Nm. The  $U_d$  reference voltage is represented by a dashed line and the  $U_q$  reference voltage with a solid line. In the  $i_d$  and  $i_q$  plots, the dashed line is the current reference and the solid line is the measured current.

5.2. Drive system using the SCVM flux observer

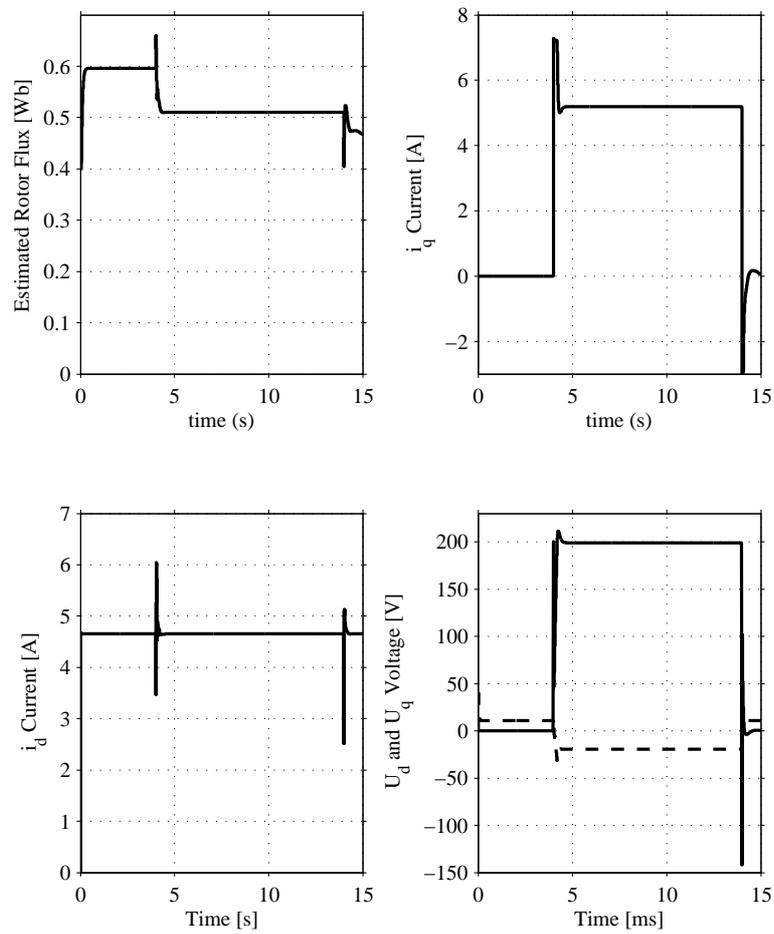


Figure 5.6: Simulation results for a speed step to 1400 RPM with a torque of 7.5 Nm. The  $U_d$  reference voltage is represented by a dashed line and the  $U_q$  reference voltage with a solid line. It is impossible to distinguish the current references from their estimated values because they follow each other so well.

### 5.2.2 Twice the nominal speed with a 5.5 Nm load torque

A speed step to twice the nominal speed is applied and the servo is set to produce a load torque of 5.5 Nm. Figure 5.7 presents the measured data while Figure 5.8 presents the simulated data. It is possible, during shorter periods of time, to increase the power in the machine thus achieving a higher torque compared to only using rated power. A torque of 5.5 Nm gives about 1.6 kW which is a bit higher compared to the rated power of 1.1 kW. The drawback of overloading the IM will be that critical parameters such as  $R_s$  and  $L_\sigma$  will change more due to the larger current which will heat resistances and saturate inductances. This will cause larger estimation errors than running the machine at nominal power. During the motor bench testing of the SCVM, it was found that 5.5 Nm was the highest torque with stable operations up to twice the nominal speed.

The estimated flux is kept at 0 before the speed step is applied because the estimator is being locked on purpose. This is done in order to prevent the estimator from "drifting away" in the time period between the magnetization and the speed step. In the simulations, the speed reference is reached in about 1 second, while the motor bench implementation reaches the speed reference in about 2 seconds. It can also be seen that the simulated speed step response is not a first order response. This is believed to be caused by the torque step that is applied in the simulations. This differs from the load that the servo represents in that the servo is applying a braking torque that the integrator needs to integrate up. Furthermore, it can be seen in Figure 5.9 and Figure 5.10 that the estimated rotor flux hits the lower limit at 0.4 Wb. In this test both the  $i_d$ - and the  $i_q$  current from the simulation is higher than the measured currents. Examining  $U_q$  and  $i_q$  in the simulation and the motor bench implementation, it can be noticed that the  $i_q$  current is about 6.0 A in the simulation while it is about 5 A in the implementation. The  $U_q$  voltage is about the same in both, about 276 – 280 V. Furthermore, the  $U_d$  voltage is slightly larger in the simulations, -70 V, where as in the implementation it is around -50 V. This causes a lower  $i_d$  current in the implementation which together with the lower  $i_q$  current will cause a lower torque and slower acceleration. The lower  $i_d$  current in the implementation is believed to be caused by the FW algorithm. It is probably not working as intended. The reason for the lower  $i_q$  current might be because the servo is incapable of creating the requested load torque. There might be an error between the actual torque and the reference. Furthermore there might be some error in the friction measurement that contributes to this difference.

5.2. Drive system using the SCVM flux observer

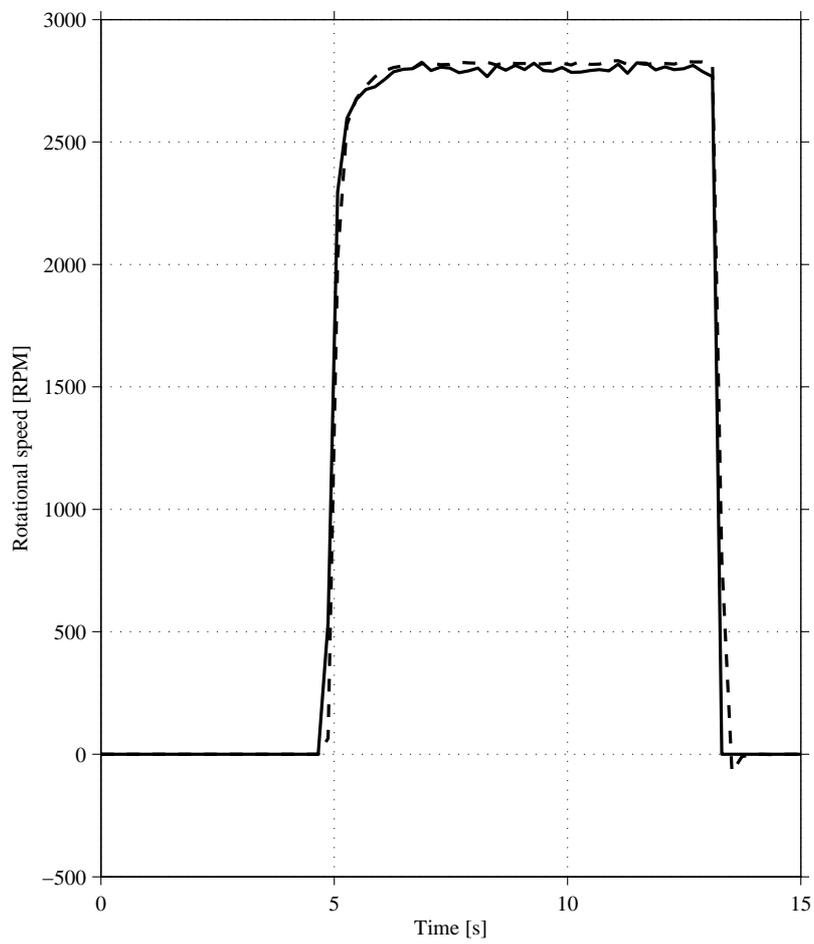


Figure 5.7: Implementation results for a speed step to 2800 RPM with a load torque of 5.5 Nm. Solid line is estimated mechanical rotor speed and dashed line is encoder speed.

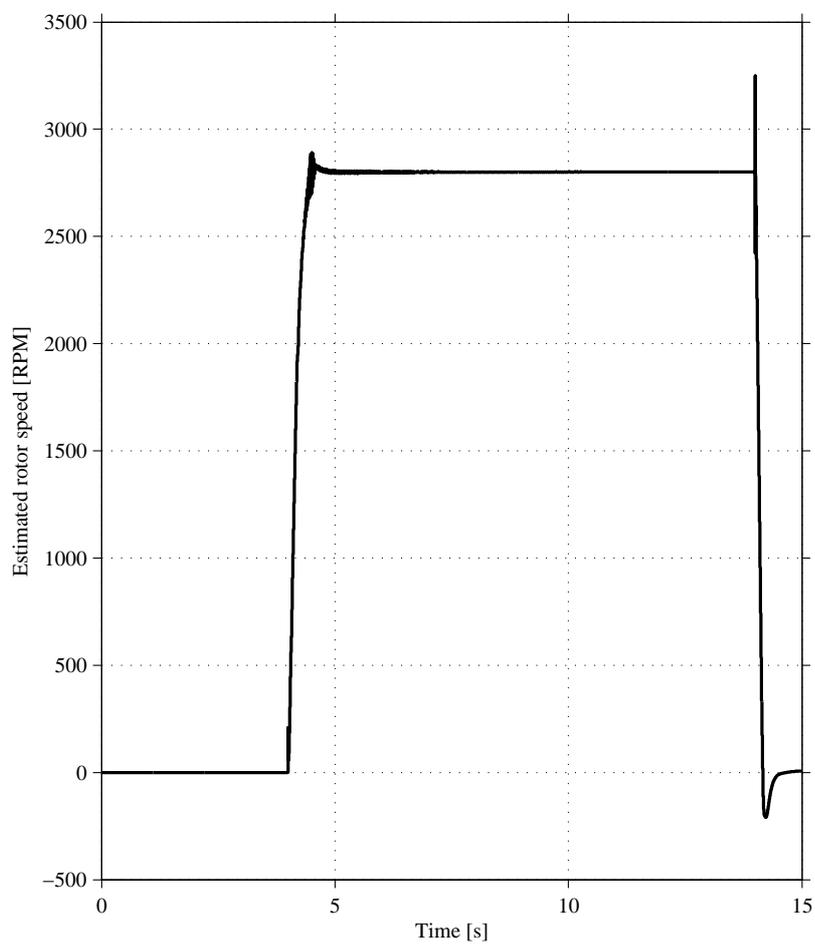


Figure 5.8: Simulation results for a speed step to 2800 RPM with a load torque of 5.5 Nm for the time the speed reference is greater than 0 RPM.

Figure 5.9 presents the measured data while figure 5.10 presents the simulated data.

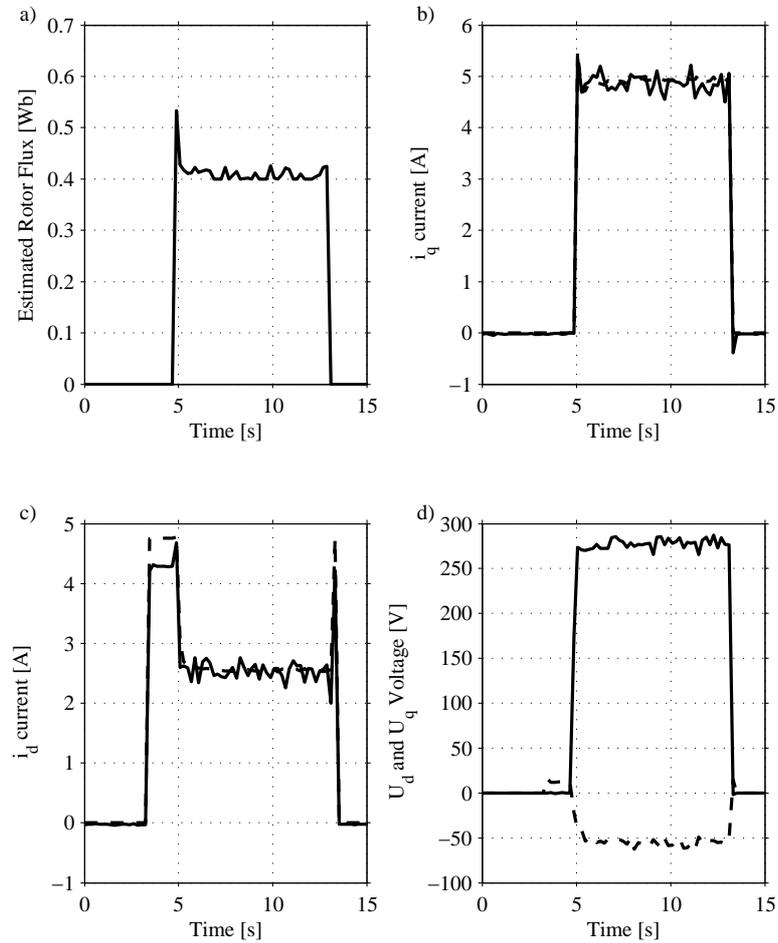


Figure 5.9: Implementation results: A speed step is set to 2800 RPM and a load torque of 5.5 Nm is applied. The  $U_d$  reference voltage is represented by a dashed line and the  $U_q$  reference voltage with a solid line. In the  $i_d$  and  $i_q$  plots, the dashed line is the current reference and the solid line is the measured current.

Figure 5.11 shows the  $U_s$  voltage for the simulation and the implementation. The field weakening algorithm tries to keep the  $U_s$  voltage constant during speeds above nominal speed. In the simulation the field weakening algorithm keeps the voltage constant, while in the implementation the  $U_s$  voltage is fluctuating more. Still, the voltage is more or less constant during the test. It is interesting to note that  $U_s$  reaches its maximum amplitude, which is about 280 V.

Chapter 5. Results

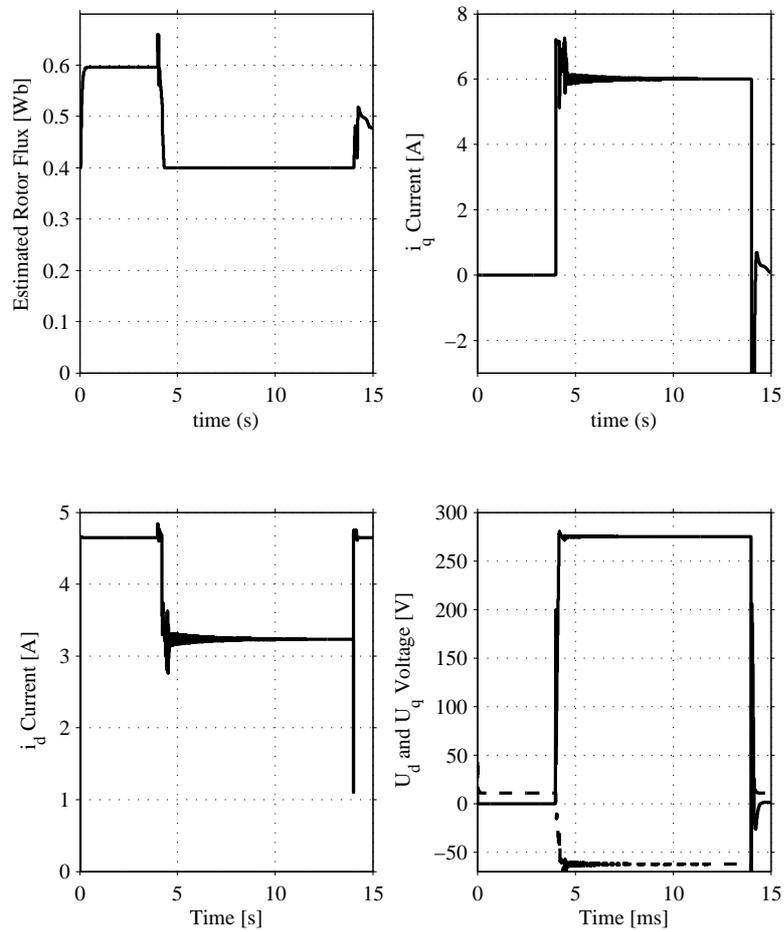


Figure 5.10: Simulation results: A speed step is set to 2800 RPM and a load torque of 5.5 Nm is applied.  $U_d$  is represented by a dashed line and  $U_q$  with a solid line.

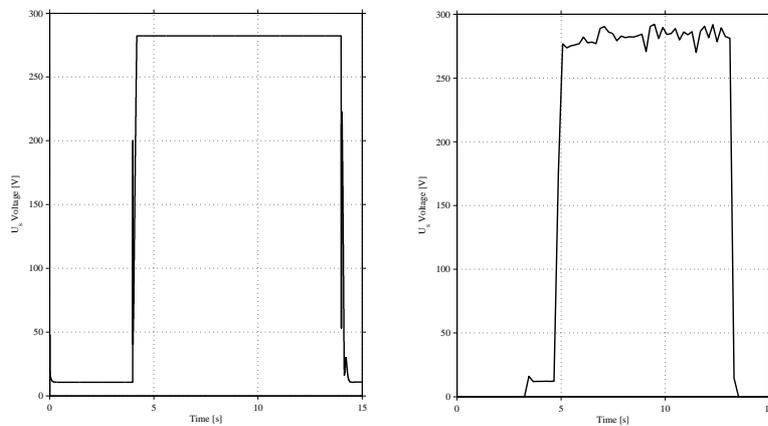


Figure 5.11: The left picture shows the stator magnitude voltage  $U_s$  for the simulation and the picture to the right shows the same voltage, but for the implementation.

### 5.2.3 Testing the FW-algorithm with several speed steps

To test the field weakening algorithm and see that the flux follows the  $i_d$  current, several speed steps were applied up to twice the nominal speed and the down again to 0 RPM. Figure 5.12 shows the various speedsteps and figure 5.13 shows the current, flux and voltage responses.

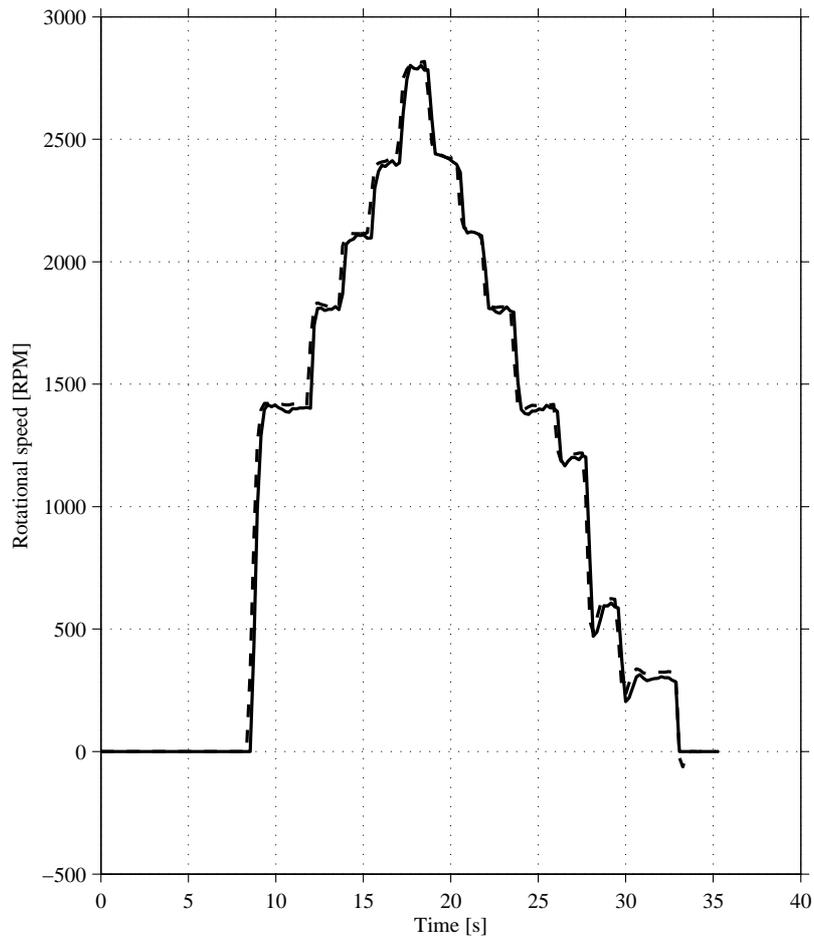


Figure 5.12: Various speed steps to test FW algorithm

The purpose of the field weakening algorithm is to lower the flux slightly before the voltage output from the converter has reached its maximum value. Looking at Figure 5.13 it can be seen that the estimated flux is decreased before the maximum voltage of 280 V is reached.  $|v_s^s|_{fw}$  is set to 325 V, which means that the flux should not start to decrease until  $|v_s^{ref}|$  exceeds  $|v_s^s|_{fw}$  [2]. This indicates that the field weakening algorithm is not working as intended. The reason for these errors must be that mistakes have been made when implementing the algorithm, since it determines the  $i_d$  reference and when the field weakening should start. However, during the implementation it was found that  $V_{base} = 325$  V worked, and was therefore chosen.

The estimated flux increases to above rated flux at about 25 s and the  $i_d$  current increases to a value that is about 4.7 A, which is above the value of 4.55 A, seen in Figure 5.5.

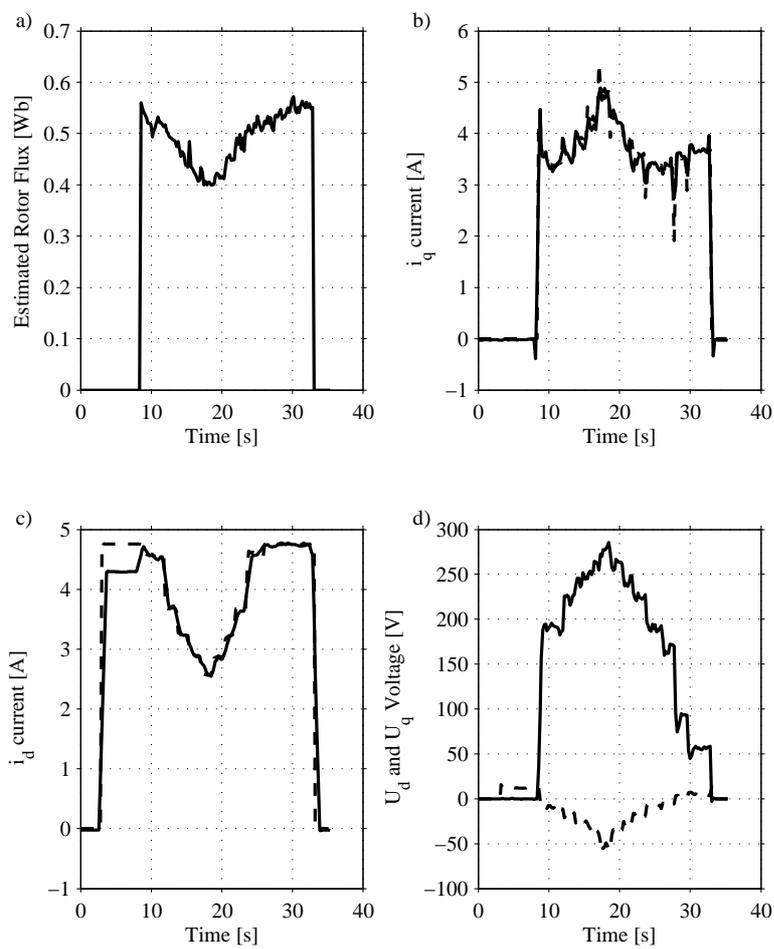


Figure 5.13: Responses in current, flux and voltage. The  $U_d$  reference voltage is represented by a dashed line and the  $U_q$  reference voltage with a solid line. In the  $i_d$  and  $i_q$  plots, the dashed line is the current reference and the solid line is the measured current.

### 5.2.4 Maximum speed with a load torque of 5.5 Nm

The motor was accelerated up to twice the nominal speed and then the speed was increased to 3000 RPM. At  $t=10$  s, a final speed step to 3120 RPM is set, and the voltage reference reaches 296 V which is higher than the maximum voltage the converter can put out. This makes the the current uncontrollable and the motor turns unstable and operation fails. This instability is characterized by large oscillation swings in the estimated mechanical rotor speed and so the actual rotor starts to oscillate. Flux, currents and voltages also start to oscillate. The AC servo machine was set to produce a load torque of 5.5 Nm. In this test, the lower limit of the flux was set to 0 Wb. Figure 5.14 shows the speeds and 5.15 shows the flux, currents and voltages.

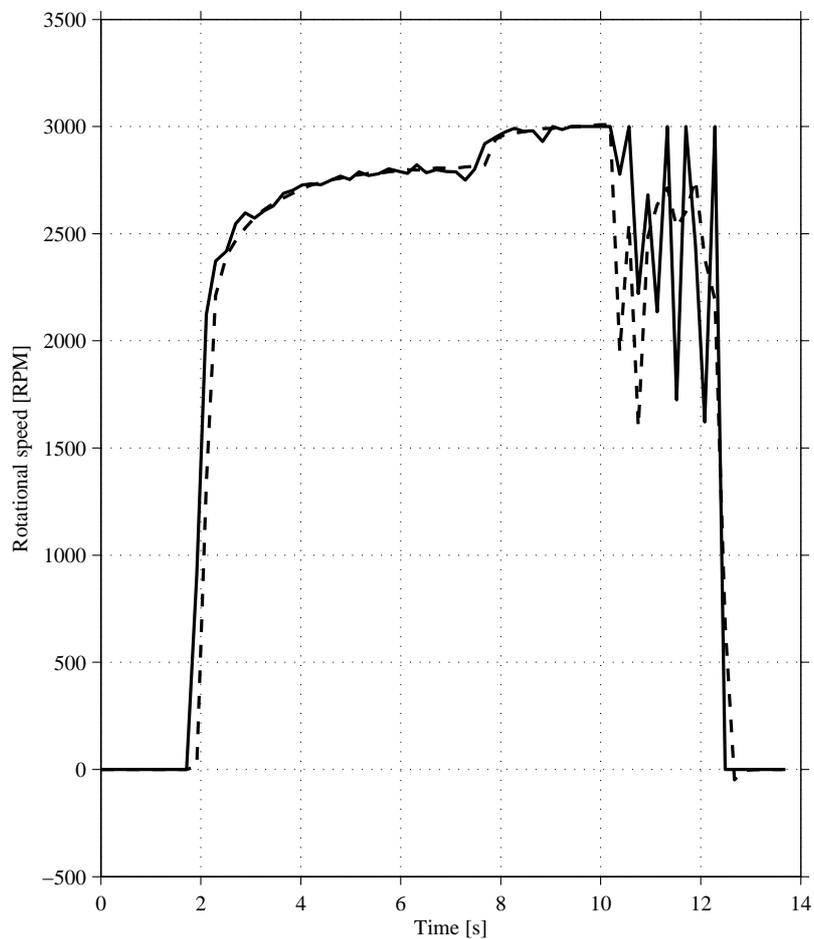


Figure 5.14: The SCVM turns unstable at 3060 RPM with a load torque of 5.5Nm

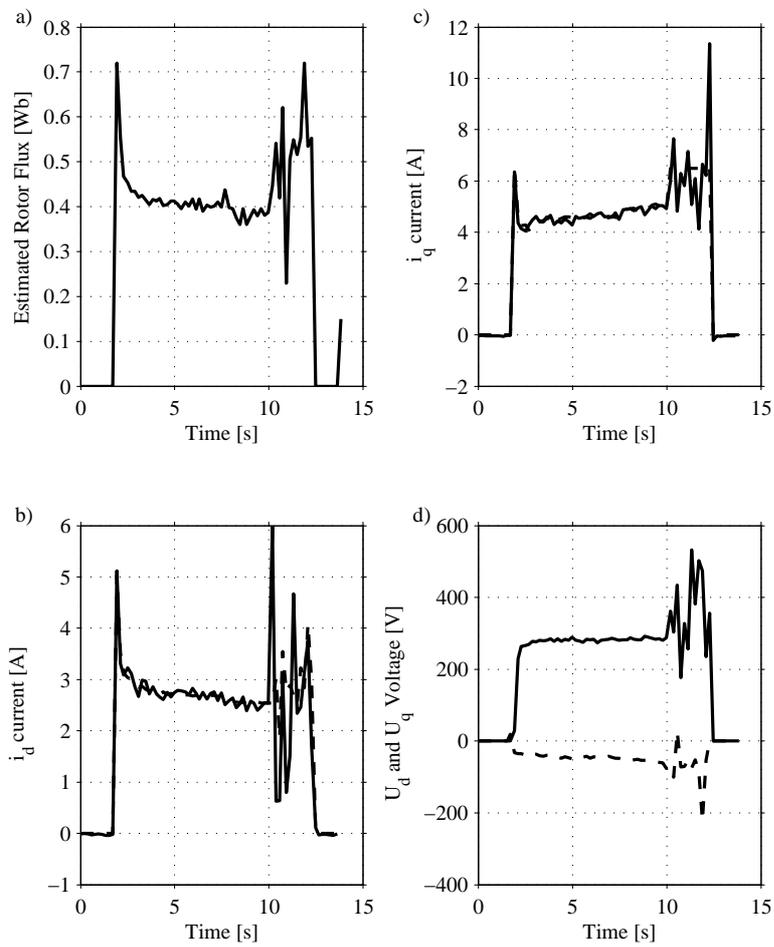


Figure 5.15: The voltage, current and flux as the SCVM turns unstable at 3060 RPM. The  $U_d$  reference voltage is represented by a dashed line and the  $U_q$  reference voltage with a solid line. In the  $i_d$  and  $i_q$  plots, the dashed line is the current reference and the solid line is the measured current.

## Chapter 6

# Discussion and conclusions

### 6.1 Discussion

Tests in the motor bench and simulations showed that the SCVM is sensitive to changes in controller parameters. These parameters need to be adjusted to the occurring load torque and the speed at which it should operate at in order to attain good performance. With the parameters in table 5.1 the system works well at nominal speed and up to twice the nominal speed. If the machine is to be run at lower speeds than nominal speed all the gain parameters in table 5.1 need to be decreased, otherwise the machine will have trouble starting.

It was also noted that the SCVM was sensitive to what bandwidth the lowpassfiltering of  $\omega_r$  was done with. Lower than 5000 rad/s and the machine would not be able to start. Higher than that and the estimated values would oscillate a great deal. There is also lowpassfiltering of  $\omega_1$  and  $\omega_2$  in order to further suppress oscillations. By trial and error, the bandwidth of these filters was set to 1500 rad/s.

The simulation and implementation were not in close agreement for the speed step up to twice the nominal speed. Comparing Figure 5.9 and Figure 5.10, it is noted that the  $i_d$  and  $i_q$  currents differed a lot. We believe that the difference in  $i_q$  currents are caused by different mechanical loads in the simulation and the implementation. The mechanical load in the simulation is an ideal constant torque which is applied together with the speed step, while the AC Servo motor applies a brake, that is released when the IM starts. After the brake has been released, the AC servo motor starts producing the torque that has been requested.

Furthermore it can be seen by comparing the results from chapter 5.2.1 and 5.2.2 that the  $i_q$  current differs more when the speed was increased to twice the nominal speed compared to the  $i_q$  current at nominal speed. An explanation for this could be that the friction measurement is wrong and it is probably nonlinear, meaning that it changes with the speed. The friction seems to be larger than what it actually is, since the  $i_q$  current in the motor bench implementation is lower than in the simulation. If the friction is lower, the load torque will of course be lower and a lower  $i_q$  current will then be needed. Another explanation could be that there is an offset between the reference and the actual value for the servo and that it in fact produces less torque than what is requested.

From Figure 5.13 it can be seen that the field weakening algorithm is working, but not as intended. Although  $|v_s^*|_{fw}$  was set to 325 V, the field weakening algorithm starts to decrease the flux at 200 V. The  $i_d$  currents also differ, especially for twice the nominal speed.

## 6.2 Conclusion

Both the current model and voltage model was implemented successfully. The SCVM works well at nominal speed with a nominal load torque of 7.5 Nm. It also works well at twice the nominal speed with a load torque of 5.5 Nm. If the speed then is increased further, it turns unstable and operation fails at 3060 RPM, thus the goal of reaching 4200 RPM was not met. From figure 5.15 it can be concluded that the maximum voltage is reached which makes the current uncontrollable. This causes the controller to turn unstable and operation fails.

At nominal speed and load torque, the simulation reaches its speed reference about 0.5 seconds faster than the implementation while the  $i_q$ -current is about 0.4 A higher in the simulation. When applying a speed step to twice the nominal speed step, the simulation is faster and reaches its speed reference about 1 seconds faster than the implementation. This difference is most probably caused by the non-ideal load torque produced by the AC servo motor and the way it operates, where in the simulation the load torque is a constant torque. The field weakening algorithm did not work as intended, because it started to decrease the flux before maximum voltage output was reached. This means that there is an error in the c-code implementation.

## 6.3 Future work

Implementing active damping could be done. This would make the system less sensitive to disturbances by effectively increasing the integrating part of the controllers. The system would also be less sensitive to parameter errors. Adding decoupling of the  $i_d$  and  $i_q$  currents would also be a good idea. As it is now, a step in the d-current affects the q-current and vice versa [2]. Furthermore, precise measurements and calculations of the friction would be helpful in order to better tune the controller gain parameters. Using the test setup in a real application would be of use. This could be done in order to test the SCVM together with the FW in applications where the speed is above rated speed. In order to thoroughly evaluate the FW algorithm, the setup should be used with a control board where measurements of the real  $U_d$  and  $U_q$  voltages are possible. As of now, it is assumed that their respective references from the current controller, are being realized by the inverter. This might not be the case. Also, it would be of interest to make a mechanical model of the servo. Unfortunately, this is not possible with the torque sensor that is being used. The torque sensor signal can only be routed to the control panel that controls the servo, and the resolution is quite bad. It is also not possible to sample the data to a graph.

# References

- [1] H. Carlsson, J. Bergerlind, “A sensorless induction machine drive,” Master thesis, Chalmers University of Technology, Aros Electronics AB, Goteborg, Sweden, 2006.
- [2] L. Harnefors, *Control of Variable-Speed Drives*. Department of electronics, Malardalen University, Vasteras, Sweden: Applied signal processing and control, 2002.
- [3] L. Harnefors, M. Jansson, R. Ottersten, and K. Pietilainen, “Unified sensorless vector control of synchronous and induction motors,” *Industrial Electronics, IEEE Transactions on*, vol. 50, no. 1, pp. 153–160, 2003.
- [4] L. Harnefors, K. Pietilainen, and L. Gertmar, “Torque-maximizing field-weakening control: design, analysis, and parameter selection,” *Industrial Electronics, IEEE Transactions on*, vol. 48, no. 1, pp. 161–168, 2001.
- [5] A. Hughes, *Electric Motor and Drives*. Elsevier Ltd., 2006.
- [6] R. Ottersten, “On control of back-to-back converters and sensorless induction machine drives,” Ph.D Thesis, Chalmers University of Technology, Department of Electric Power Engineering, Goteborg, Sweden, 2003.
- [7] G. Slemon, “Modelling of induction machines for electric drives,” *Industry Applications, IEEE Transactions on*, vol. 25, no. 6, pp. 1126–1131, 1989.
- [8] J. Witkins. (2012) A push to make motors with fewer rare earths. [Online]. Available: [http://www.nytimes.com/2012/04/22/automobiles/a-push-to-make-motors-with-fewer-rare-earths.html?pagewanted=all&\\_r=0](http://www.nytimes.com/2012/04/22/automobiles/a-push-to-make-motors-with-fewer-rare-earths.html?pagewanted=all&_r=0)

## *References*

# Appendix A

## Appendix

### A.1 Matlab code

#### A.1.1 Machine model

```
function [sys,x0,str,ts] = asymach3(t,x,u,flag,Rs,Rr,Lm,Ls,np,J,b,xi)
%CSFUNC An example M-file S-function for defining a continuous system.
% Example M-file S-function implementing continuous equations:
%      x' = Ax + Bu
%      y = Cx + Du
% See sfuntmpl.m for a general S-function template.
switch flag,

    %%%%%%%%%%%
    % Initialization %
    %%%%%%%%%%%
    case 0,
        [sys,x0,str,ts]=mdlInitializeSizes(t,x,u,flag,Rs,Rr,Lm,Ls,np,J,b,xi);

    %%%%%%%%%%%
    % Derivatives %
    %%%%%%%%%%%
    case 1,
        sys=mdlDerivatives(t,x,u,flag,Rs,Rr,Lm,Ls,np,J,b,xi);

    %%%%%%%%%%%
    % Outputs %
    %%%%%%%%%%%
    case 3,
        sys=mdlOutputs(t,x,u,flag,Rs,Rr,Lm,Ls,np,J,b,xi);

    %%%%%%%%%%%
```

Appendix A. Appendix

```

% Unhandled flags %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
case { 2, 4, 9 },
    sys = [];

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Unexpected flags %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
otherwise
    error(['Unhandled flag = ', num2str(flag)]);
end
% end csfunc
%
%=====
% mdlInitializeSizes
% Return the sizes, initial conditions, and sample times for the S-function.
%=====
%
function [sys,x0,str,ts]=mdlInitializeSizes(t,x,u,flag,Rs,Rr,Lm,Ls,np,J,b,xi)

sizes = simsizes;
sizes.NumContStates = 6;
sizes.NumDiscStates = 0;
sizes.NumOutputs = 8;
sizes.NumInputs = 3;
sizes.DirFeedthrough = 0;
sizes.NumSampleTimes = 1;

sys = simsizes(sizes);
x0 = xi;
str = [];
ts = [0 0];
% end mdlInitializeSizes
%
%=====
% mdlDerivatives
% Return the derivatives for the continuous states.
%=====
%
function sys=mdlDerivatives(t,x,u,flag,Rs,Rr,Lm,Ls,np,J,b,xi)

%state variables: stator and rotor currents in alphabeta, mechanical speed,
%rotor position(angle)
isa=x(1);

```

```

isb=x(2);
ira=x(3);
irb=x(4);
wrat=x(5);
theta=x(6);

% input signals from m-file
Usalpha=u(1);
Usbeta=u(2);
Tlextra=u(3);

%defining resistance matrix
RMAT=[Rs 0 0 0;0 Rs 0 0; 0 wrat*Lm Rr wrat*Ls; -wrat*Lm 0 -wrat*Ls Rr];
%defining inductance matrix
LMAT=[Ls 0 Lm 0; 0 Ls 0 Lm; Lm 0 Ls 0; 0 Lm 0 Ls];

%forming the state space A and B matrixes
BMAT=inv(LMAT);
AMAT=-inv(LMAT)*RMAT;

%formation of the 4 current derivatives
sys = AMAT*x(1:4)+ BMAT*[Usalpha;Usbeta;0;0];

%defining Te and Tl
Te=3*np/2*Lm*(ira.*isb-isa.*irb);

Tl=Tlextra+b*wrat/np;

%forming rotor speed derivative d/dt(wr)
sys(5)=np/J*(Te-Tl);

%forming of angle theta d/dt(theta)=wr
sys(6)=wrat;

% end mdlDerivatives
%
%=====
% mdlOutputs
% Return the block outputs.
%=====
%
function sys=mdlOutputs(t,x,u,flag, Rs,Rr,Lm,Ls,np,J,b,xi)

% the time t is already available in all function in the s-function

```

## Appendix A. Appendix

```
isa=x(1);
isb=x(2);
ira=x(3);
irb=x(4);
wrat=x(5);
theta=x(6);

%defining rotor flux
psi_ra=Ls.*ira+Lm.*isa;
psi_rb=Ls.*irb+Lm.*isb;

%calculating angle and magnitude of the rotor flux
psi_magn_r=sqrt(psi_ra^2+psi_rb^2);
psi_angle=atan2(psi_rb,psi_ra);

sys = [isa, isb, ira, irb, wrat, theta, psi_magn_r, psi_angle];
% end mdlOutputs
```

### A.1.2 simulation file with parameters

```
%The motor simulated is the motor from AROS electronics.
%Delta-ratings: 230V, 1.1kW, cosfi=0.81, Trated=7.5Nm, psirated=0.6Wb,
%Is=4.7A rms, ratedspeed=1400rpm, 23.3Hz, 2 pole pairs
%Motor parameters were measured with a locked rotor and no load test.
% Rs=2.3 ohm
% Rr = 2.4 ohm
% Lls=Llr = 9.6mH
% Lm = 118.5mH
%no cross coupling, feedforward of the backemf or active damping is used
%since the implemented system is using neither

clear all, close all, clc

% PART1 Parameter definitions
K=1; %amplitude invariant transformation, (K=sqrt(3/2) gives power invariant)
np=2; %pole pair
Israted=8.7; %4.7A rms rated current for the machine – 6.65 A peak
%(however in the lab it performed a lot better during starts with this limit)
w1_rated=2*pi*50/np;
psi_R_rated=0.56; %psi_r_rated=0.605 %psi_R_rated = 0.56
Lambda = sqrt(2); % leakage term for the observer
mu=-1; %-1 according to harnefors
Lls=0.0096; %stator leakage inductance Lls=Llr
Lm=0.118; %magnetizing inductance
```

```

Ls=Lm+Lls;%Ls=Lr
Lr=Ls;
Rs=2.3;%stator resistance
Rr=2.4;%rotor resistance
RR=Rr*(Lm/Lr)^2;%rotor resistance in inverse gamma
RR_hat=RR;%
Rs_hat=Rs;%
J=0.0025+0.00152+0.00127;
%inertia for IM, servo, and axial couplings
b=0.003;%measured friction coefficient
LM=(Lm^2/Lr);%magnetizing inductance in inverse gamma
alpha_c=1500;%bandwidth for currentcontroller in rad/s
alpha_w=30;%bandwidth for speedcontroller in rad/s
Lsigma_hat = (Ls-LM);%inverse gamma inductance
Ba=(alpha_w*J-b)*0;% active damping, put to 0
Ra=(Lsigma_hat*alpha_c-Rs-RR)*0;%see above
Kpc=alpha_c*Lsigma_hat;%prop gain for cc
Kic=alpha_c*(RR+Rs+Ra);%integral gain for cc
Kpw=alpha_w*J;%prop gain for sc
Kiw=alpha_w*(b+Ba)*15;%integral gain for sc

% setting up the simulation time and time step
Tstart=0;% Starting time of the simulation
Tstop=3;% End time of the simulation
xi=[0;0;0;0;0;0];%initial conditions,

TstepTime=0.5;% The time when the torque step is applied
TstepValue=5.0;%7.5Nm rated torque
%7.1tstep ger 7.5ish Te vid 1400rpm
%4.6tstep ger 5.5ish Te vid 2800rpm

% PART 2 simulation call
sim('Asynmach3',[Tstart,Tstop]);
postprocessingIM3
% Call for the m-file that plots the results of the simulation

```

### A.1.3 plot file

```

%calculate torque and power using alpha and beta voltage/current and
%amplitude invariant transformation K=1
wmech=wr/np;
Te=3*np/2*Lm*(ira.*isb-isa.*irb);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%figure%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
figure(1)

```

## Appendix A. Appendix

```
subplot(3,4,1) % subplot 3
plot(time, Psi_R_hat, 'b', time, Psiref, 'r', time, psi_magn_r, 'g')
grid on
xlabel('time (s)')
ylabel('Rotor flux [Wb]')
legend('Psi hat', 'Psi ref', 'psi magn')
```

```
subplot(3,4,2)
plot(time, theta_hat*(180/pi), 'b', time, theta*(180/pi), 'g')
grid on
xlabel('time (s)')
ylabel('Rotor flux angle [deg]')
```

```
subplot(3,4,3)
plot(time, w1_hat*60/(2*pi), 'b')
grid on
xlabel('time (s)')
ylabel('Stator frequency [RPM]')
```

```
subplot(3,4,4) % subplot 2
plot(time, wmech_hat*60/(2*pi), 'b', time, wmechref*60/(2*pi), 'r',
time, wmech*60/(2*pi), 'g')
grid on
xlabel('time (s)')
ylabel('Rotor speed [RPM]')
```

```
subplot(3,4,5) % subplot 2
plot(time, Psiref - Psi_R_hat, 'b')
grid on
xlabel('time (s)')
ylabel('Rotor flux error [Wb]')
```

```
subplot(3,4,6) % subplot 2
plot(time, unwrap(psi_angle) - theta_hat, 'b')
grid on
xlabel('time (s)')
ylabel('Rotor flux angle error [deg]')
```

```
subplot(3,4,7) % subplot 2
plot(time, w1_hat*60/(2*pi) - wr_hat*60/(2*pi), 'b')
grid on
xlabel('time (s)')
ylabel('Slip [RPM]')
```

## A.1. Matlab code

```
subplot(3,4,8) % subplot 2
plot(time ,wmech*60/(2*pi)-wmechref*60/(2*pi), 'b')
grid on
xlabel('time (s)')
ylabel('Speed error [RPM]')

subplot(3,4,9) % subplot 3
plot(time ,real(is_dq_est), 'b',time ,real(isref), 'r',
time ,real(is_dq_real), 'g')
grid on
xlabel('time (s)')
ylabel('id [A]')

subplot(3,4,10) % subplot 3
plot(time ,Te, 'g',time ,Teref, 'r')
grid on
xlabel('time (s)')
ylabel('Electrodynamical torque [Nm]')

subplot(3,4,11) % subplot 3
plot(time ,imag(is_dq_est), 'b',time ,imag(isref), 'r',
time ,imag(is_dq_real), 'g')
grid on
xlabel('time (s)')
ylabel('iq [A]')

subplot(3,4,12) % subplot 3
plot(time ,real(Us_dqlim_ref), 'b',time ,imag(Us_dqlim_ref), 'g')
grid on
xlabel('time (s)')
ylabel('U_s (abs)')
```

## A.2 IM ratings

The ratings of the IM can be found in table A.1

Table A.1: The IM ratings

| V $\Delta$ / Y | Hz | kW  | A $\Delta$ / Y | rpm  | $\cos(\phi)$ |
|----------------|----|-----|----------------|------|--------------|
| 230/400        | 50 | 1.1 | 4.7/2.7        | 1400 | 0.8          |

The motor was  $\Delta$  coupled and the rated torque and rated flux can be calculated by using

$$T = n_p i_q \psi_r \quad (\text{A.1})$$

$$T = \frac{P}{\omega_r} \quad (\text{A.2})$$

$$\psi_r = \frac{P}{\omega_r i_q n_p} \quad (\text{A.3})$$

where  $n_p$  is the number of pole pairs which is 2 for this motor and  $i_q$  is the peak value of the current. Using the parameters from table A.1 the rated torque becomes 7.5 Nm and the rated flux becomes 0.56 Wb. The inertia for the motor shaft was according to its data sheet  $0.0025 \text{ kgm}^2$  and for the servo motor it is  $0.00152 \text{ kgm}^2$ .

## A.3 Induction machine parameter measurements and calculations

The stator resistance,  $R_s$ , was measured with a multimeter and the value was  $R_s = 2.3 \Omega$ . The rest of the parameters of the induction machine was measured with an Infratek 106 A power analyzer during a locked-rotor and a no-load test. Table A.2 shows the measured values for the no-load test and table A.3 shows the measured values from the locked-rotor test.

Table A.2: Measured parameters from no-load test

| $I_0$     | $U_0$       | $P_{tot,0}$ | $Q_{tot,0}$ | $\cos(\phi)_0$ |
|-----------|-------------|-------------|-------------|----------------|
| 3.46A RMS | 135.27V RMS | 261 W       | 1400 VAR    | 0.183          |

Table A.3: Measured parameters from locked-rotor test

| $I_k$     | $U_k$     | $P_{tot,k}$ | $Q_{tot,k}$ | $\cos(\phi)_k$ |
|-----------|-----------|-------------|-------------|----------------|
| 4.75A RMS | 28.8V RMS | 320 W       | 409 VAR     | 0.616          |

By neglecting the magnetizing inductance, the leakage inductances can be calculated from the locked-rotor test as

### A.3. Induction machine parameter measurements and calculations

$$L_{lr} + L_{ls} = \frac{Q_{tot,k}}{3i_k^2 \omega} = \frac{409}{3 * 4,75^2 * 2\pi * 50} = 19.2mH \quad (A.4)$$

Assuming  $L_{lr}=L_{ls}$  the value of the inductances become

$$L_{lr} = L_{ls} = \frac{L_{lr} + L_{ls}}{2} = 9.6mH \quad (A.5)$$

The rotor resistance is given by

$$R_r = \frac{P_{tot,k}}{3i_k^2} - R_s = 2.43\Omega \quad (A.6)$$

With the measured parameters from the no-load test, the magnetizing inductance is calculated as

$$L_m = \frac{\frac{Q_{tot,0}}{3} - I_0^2 \omega L_{ls}}{I_m^2 \omega} \quad (A.7)$$

where  $I_m = I_0 \sin(\phi) = 3.46 \sin(79^\circ) = 3.4A$ . This gives

$$L_m = \frac{\frac{1400}{3} - 3.46^2 * 314 * 9.6 * 10^{-3}}{3.4^2 * 314} = 118.55mH \quad (A.8)$$

Using the transformation coefficient,  $\gamma$ , and equations 2.22 - 2.25, the Inverse gamma parameters can be calculated. All values are given in table A.4

Table A.4: The IM parameters

| Variable   | Value        |
|------------|--------------|
| $R_r$      | $2.4\Omega$  |
| $R_s$      | $2.3\Omega$  |
| $R_R$      | $2.05\Omega$ |
| $L_\sigma$ | $18.5mH$     |
| $L_{lr}$   | $9.6mH$      |
| $L_{ls}$   | $9.6mH$      |
| $L_r$      | $128mH$      |
| $L_s$      | $128mH$      |
| $L_m$      | $118.6mH$    |
| $L_M$      | $110mH$      |
| $\Psi_R$   | $0.52Wb$     |

## A.4 Overview of the Matlab block diagram

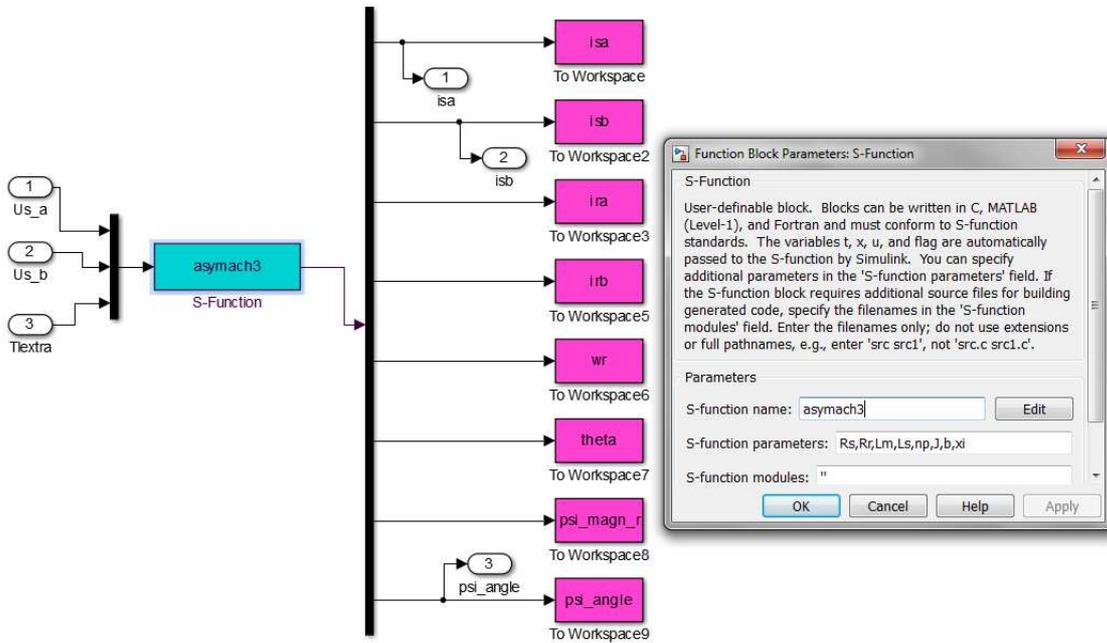


Fig. A.1 The Induction Machine block

