

CHALMERS



Parameter Identification of a Permanent Magnet Synchronous Motor

Master's Thesis in Systems, Control & Mechatronics

HENRIK NEUGEBAUER

Department of Signals & Systems

CHALMERS UNIVERSITY OF TECHNOLOGY

Gothenburg, Sweden 2012

Master's Thesis EX011/2012

To my family Kerstin, Ulf, Sarah, Gustav, Jonathan and Marita

ABSTRACT

In this thesis, two identification methods for time varying electrical permanent magnet synchronous motor (PMSM) parameters are studied and compared. The Recursive Least Square (RLS) with forgetting factor and the Normalized Projection Algorithm (NPA) are applied to a salient PMSM model in dq -coordinates. The stator resistance and the stator inductances are identified both online and offline. The simulation results illustrate the efficiency of the two methods. It is shown that both algorithms are capable to identify the parameters both online and offline, but compared to the NPA, the RLS tracks the parameter variations much better.

Computational time per iteration is also estimated and it is shown that the NPA is approximately three times faster than the RLS.

By experimental verification from real motor data it is shown that the algorithms do not convergence to the same parameter values.

The stator resistance and the stator inductances have also been measured directly on the motor windings and compared with the identified parameter values from the algorithms. It is shown that they deviate more than expected. Analysis of the obtained estimates indicates that the parameter values could have been fitted to some disturbance in the measured motor currents and/or that the applied voltages to the motor deviate from the reference voltages used as input to the algorithms.

Acknowledgments

This is the report of my final Master's Thesis in the Master's Program *System, Control and Mechatronics* at Chalmers University of Technology in Sweden. The Master's Thesis was done in cooperation with Aros electronics AB.

First of all, I would like to thank my supervisor at Aros electronics AB, Ph.D. Mikael Alatalo, for all his help and encouragement.

I would also like to thank my examiner Prof. Jonas Sjöberg for his help with theoretical questions and alternative approaches to problems.

Further, I would like to thank all employees at Aros electronics AB for all their kindness and support, and for making me feel most welcome during my thesis work. Extra thanks to Andreas Pettersson and Daniel Chäderström for helping me with the software ArOS, and to Conny Dagerot for his help to calculate computational times on some digital signal processors.

I am also very grateful to my good friend Johan Petersson for endless proofreading of my thesis.

List of Contents

Dedication	iii
Abstract	v
Acknowledgments	vii
List of Tables	xii
List of Figures	xiii
List of Abbreviations	xvi
List of Symbols	xvii
1 Introduction	1
1.1 Motivation of Parameter Identification in Permanent Magnet Synchronous Motor Drives	1
1.2 Offline Parameter Identification	2
1.3 Online Parameter Identification	3
Spectral Analysis	3
Observer Based System	3
Model Reference Adaptive System	4
Other Methods	5
1.4 Outline of Thesis	5
	ix

1.5	Contributions	7
2	Modelling of Permanent Magnet Synchronous Motors	8
2.1	Three Phase Systems	8
2.2	Space Vectors	9
2.3	Modelling of a non salient PMSM	13
2.4	Modelling of a salient PMSM	14
3	Speed and Current Vector Control	17
3.1	Vector Control	17
3.2	Motivation for Online Parameter Identification within Sensor-less Vector Control.....	20
4	Discretization and Linear Regression Models	22
4.1	Discretization	22
4.2	Linear Regressions	24
5	Recursive Parameter Estimation	26
5.1	The Least Square Method.....	26
5.2	Recursive Least Square Algorithm	27
5.3	Recursive Least Square with Forgetting Factor.....	29
5.4	Normalized Projection Algorithm	30
6	Simulations	32
6.1	Simulation procedure	32
6.2	Simulation Results	35
7	Implementation	47
7.1	Implementation complexity	47
7.2	Implementation Set-up	48

7.3	Experimental Results	50
8	Conclusion and Future Work	55
8.1	Conclusion.....	55
8.2	Future Work	57
	Bibliography	58

List of Tables

6.1	Magnitude mean error for the parameters when RLS and NPA are used as commission offline parameter identification methods	35
6.2	Maximal magnitude of the parameter error for the RLS and NPA when the parameter changes sinusoidally	36
6.3	Steady state mean value of the parameters after the first step at 0.3s of simulation in Figure 6.5. T_L increases from 0.3Nm to 0.5Nm, L_d decrease with 5%, L_q decrease with 50% and R_s increase with 40%	36
6.4	Steady state mean value of the parameters after the first step at 0.3s of simulation in Figure 6.6. T_L decrease from 0.3Nm to 0.1Nm by a step, L_d increase with 5%, L_q increase with 10% and R_s decrease with 20%	37
6.5	Mean parameter errors when an angular rotor flux error of 40% is introduced in the vector control system	38
7.1	Number of operations that are needed for implementation of the RLS and the NPA.	48
7.2	Estimated computational time for RLS and NPA.	48

List of Figures

1.1	MRAS based identification of PMSM parameters	5
2.1	Ideal three phase currents	9
2.2	Construction of a current space vector.	10
2.3	Coordinates of a salient PMSM.	15
2.4	Dynamic equivalent circuits for the salient PMSMs electrical subsystem.	16
3.1	Transformation of the PMSM model into dq coordinates.	18
3.2	Vector control system with cascaded current and speed control loops.	19
3.3	Online identification of electrical PMSM parameters within sensorless vector control.	21
6.1	Inner current loop of the vector control system together with the identification part.	34
6.2	RLS and NPA used as commission offline parameter identifica- tion methods in stand still. The start values are: $L_{d0} = 0.5L_d$, $L_{q0} = 0.5L_q$, $R_{s0} = 0.5R_s$	39
6.3	RLS and NPA used as commission offline parameter identi- fication methods in speed 600[rpm] and no load. The start values are: $L_{d0} = 0.5L_d$, $L_{q0} = 0.5L_q$, $R_{s0} = 0.5R_s$	40

6.4	Simulations when the algorithms are used as online parameter estimation methods. R_s , L_d and L_q are here defined as variables of times that changes sinusoidal.	41
6.5	A simulation were the load torque and parameters are changed with steps. After 0.3s T_L increases from 0.3Nm to 0.5Nm by a step, L_d decrease with 5%, L_q decrease with 50% and R_s increase with 40% in the first step.	42
6.6	A simulation were the load torque and parameters are changed with steps. After 0.3s T_L decrease from 0.3Nm to 0.1Nm by a step, L_d increase with 5%, L_q increase with 10% and R_s decrease with 20%.	43
6.7	A simulation were the load torque and parameters are changed with steps similar to Figure 6.6, but now the the covariance matrix $P(t)$ is retested when a step change in the parameters occur.	44
6.8	A simulation were an angular rotor flux error is introduced in the vector control system. The angular error increments from 0° to 10°	45
6.9	A simulation were an angular rotor flux error is introduced in the vector control system. The angular error increments from 10° to 40°	46
7.1	Vector control system together with the identification signals implemented in a DSP. The inputs and outputs are recorded and stored in a memory.	49
7.2	Result from a realization were $u_{sq,x}$ and $u_{sd,x}$ switches between $\pm 4V$ with a approximately probability of 20 percent in each sample. $T_{L,extra} = 0$ and $\omega_{m,ref} = 0$	52

- 7.3 Result from a realization were $u_{sq,x}$ and $u_{sd,x}$ switches between $\pm 4V$ with a approximately probability of 20 percent in each sample, $T_{L,extra} = 0.5[Nm]$ and $\omega_{m,ref} = 600[RPM]$ 53
- 7.4 Result from a realization were $u_{sq,x}$ and $u_{sd,x}$ switches between $\pm 3V$ with a approximately probability of 6 percent in each sample, $T_{L,extra} = 0.1[Nm]$ and $\omega_{m,ref} = 600[RPM]$ 54

List of Abbreviations

PMSM	Permanent Magnet Synchronous Motor.....	1
EKF	Extended Kalman Filter	3
MRAS	Model Reference Adaptive System	4
RLS	Recursive Least Square	3
NN	Neural Networks	5
NPA	Normalized Projection Algorithm	6
IM	Induction Machine	17
SM	Synchronous Machine	17
PI	Proportional Integral (Controller)	19
PWM	Pulse Width Modulation	18
DSP	Digital Signal Processor	22
ZOH	Zero Order Hold	22

List of Symbols

i_{sa}, i_{sb}, i_{sc}	Three phase stator currents	9
$\alpha\beta$	Stator fixed coordinate system	9
$i_{s\alpha}, i_{s\beta}$	$\alpha\beta$ components of the stator currents	9
\mathbf{i}_s^s	Complex stator current space vector	9
j	$\sqrt{-1}$	9
ω_s	Stator frequency	9
T_{32}	Three phase to two phase transformation matrix	10
\hat{I}_s	Stator current peak voltage	11
i_{sd}, i_{sq}	dq components of the stator currents	11
T_{23}	Two phase to three phase transformation matrix	11
ϕ	phase shift	11
\mathbf{i}_s	Complex stator current vector in dq coordinates	12
i_s^s	Real valued stator current space vector	12
J	matrix, corresponding to the imaginary unit j	12
$\hat{d}\hat{q}$	Estimated field synchronous coordinate system	15
$\hat{\omega}_r$	Estimator of ω_r	15
$T_{dq}(\cdot)$	Transformation matrix, from $\alpha\beta$ in to dq	13
$T_{\alpha\beta}(\cdot)$	Transformation matrix, from dq in to $\alpha\beta$	13
R_s	Stator per phase resistance	13
ψ_m	Rotor flux linkage due to PM excitation	13

List of Figures

dq	Field synchronous coordinate system	13
\mathbf{v}_s^s	Complex stator voltage space vector	13
ψ_s^s	Complex stator flux linkage space vector	13
ψ_m^s	Complex rotor flux linkage space vector	13
L_s	Stator per phase inductance	13
θ_r	electrical rotor angle	13
ω_r	electrical rotor frequency	14
L_d, L_q	dq components of the stator per phase inductance ..	14
u_{sd}, u_{sq}	dq components of the stator voltages	14
n_p	Number of pole pairs of the motor	16
T_e	Electrical torque of the motor	16
J_{ml}	Inertia of the motor and load	16
T_L	Motor load	16
B_{ml}	Viscous friction of the motor and load	16
$T_{L,extra}$	The load connected to the motor	16
ω_m	mechanical rotor speed	18
$u_{s\alpha}, u_{s\beta}$	$\alpha\beta$ components of the stator voltages	18
u_{sa}, u_{sb}, u_{sc}	Three phase stator voltages	18
G_v	Voltage source inverter equivalent gain	19
$i_{s,abc}$	Vector containing i_{sa}, i_{sb}, i_{sc}	21
$u_{s,abc}$	Vector containing u_{sa}, u_{sb}, u_{sc}	21
$\hat{R}_s, \hat{L}_d, \hat{L}_q, \hat{\psi}_m$	Estimation of R_s, L_d, L_q, ψ_m	21
$\hat{\theta}_r,$	Estimation of θ_r	21
A	System Matrix	22
B	Input Matrix	22
C	Output Matrix	22
$u(t)$	Input vector	22

$x(t)$	State vector	22
$y(t)$	Output vector	22
I	Identity matrix	22
A_d	Discretized system matrix	23
B_d	Discretized input matrix	23
C_d	Discretized output matrix	23
Ψ	Discretization matrix	23
$\varphi(t)$	Regression vector at time t	24
θ	Parameter Matrix	24
\bar{u}_{sq}	$u_{sq} - \omega_r \psi_m$	24
$V(\theta, t)$	Loss function to be minimized	27
$P(t)$	Matrix used in recursive calculations	27
$L(t)$	Vector used in recursive calculations	28
$\varepsilon(t)$	Output error vector from the model	28
λ	Forgetting factor	30
γ	Stepsize factor used in the NPA	31
$u_{sd,x}, u_{sq,x}$	dq components of identification signals	33
n_d, n_q	dq components current measurement disturbances ..	33
u_{sd}^*	$u_{sd,ref} + u_{sd,x}$	33
u_{sq}^*	$u_{sq,ref} + u_{sq,x}$	33
$u_{sd,ref}, u_{sq,ref}$	dq components of the reference stator voltages	34
$i_{sd,ref}, i_{sq,ref}$	dq components of the reference stator currents	34
$i_{sd,m}, i_{sq,m}$	dq components of the measured stator currents	34
L_{d0}, L_{q0}, R_{s0}	Start values when estimating L_d, L_q, R_s	35
$\bar{L}_{d,npa}, \bar{L}_{q,npa}$	Mean value of estimated L_d, L_q with NPA	39
$\bar{L}_{d,rls}, \bar{L}_{q,rls}$	Mean value of estimated L_d, L_q with RLS	39
$\bar{R}_{s,npa}$	Mean value of estimated R_s with NPA	39

List of Figures

$\overline{R}_{s,rls}$	Mean value of estimated R_s with RLS	39
\bar{e}_{npa}	Mean error of estimated parameter with NPA	39
\bar{e}_{rls}	Mean error of estimated parameter with RLS	39
$u_{sa,ref}^*$	Reference voltage to the PWM for build up u_{sa}	49
$u_{sb,ref}^*$	Reference voltage to the PWM for build up u_{sb}	49
$u_{sc,ref}^*$	Reference voltage to the PWM for build up u_{sc}	49
$u_{s\alpha}^*, u_{s\alpha}^*$	$\alpha\beta$ components of u_{sd}^*, u_{sq}^*	49
$i_{sa,m}$	Measured i_{sa}	49
$i_{sb,m}$	Measured i_{sb}	49
$i_{sc,m}$	Measured i_{sc}	49
$i_{sd,m}, i_{sq,m}$	dq components of the measured stator currents	49
$i_{s\alpha,m}, i_{s\beta,m}$	$\alpha\beta$ components of the measured stator currents	49
$\omega_{r,m}$	Measured electrical rotor frequency	49
$L_{d,m}, L_{q,m}$	Measured values of L_d, L_q	50
$R_{s,m}$	Measured value of R_s	50
$\bar{\delta}_{npa}$	Mean parameter deviation, NPA vs. measurements .	52
$\bar{\delta}_{rls}$	Mean parameter deviation, RLS vs. measurements ..	52

Chapter 1

Introduction

In this chapter, a motivation for parameter identification in permanent magnet synchronous motor drives is given. The concept of offline and online parameter identification in motor drives is introduced and a brief overview of different methods found in the literature is presented. The focus is put on parameter identification methods useful for motor control. The outline of the thesis, as well as scientific contributions, are also presented.

1.1 Motivation of Parameter Identification in Permanent Magnet Synchronous Motor Drives

A permanent magnet synchronous motor (PMSM) is frequently used in the industry due to its high efficiency and quick mechanical dynamics. To design high performance control systems it is important to have accurate knowledge about the motor parameters. The data sheet, containing the motor parameters, that comes from the manufacturer of an electrical motor is not always

available, and when a new PMSM should be installed in an application, it is therefore desirable to be able to identify the parameters automatically with parameter identification. This is also more time efficient compared to manually changing the parameter values. Even when the data sheet is available, it does not usually take into account that some parameters tend to vary during different operation points. Classical motor control techniques such as Vector Control (see Chapter 3) require a position sensor to control the PMSM, and in such case the parameter information from the data sheet is normally sufficient. But since a sensor brings high cost and sometimes low reliability, it is desirable to remove the sensor [18] which will require more advanced control techniques. Many position sensorless control methods are reliable on an accurate motor model and by introducing online parameter identification in such drive systems, it is possible to improve the control performance (see section 1.3 for explanation of the concept online parameter identification).

1.2 Offline Parameter Identification

Techniques for determine motor parameters in standstill or in rotation without any load are called "offline parameter identification methods". The offline methods can further be categorized in self-commission and commission [1]. In self-commission all required motor parameters suitable for control are determined without any previous knowledge about the parameter values. In commission some previous information about the parameter values is known.

1.3 Online Parameter Identification

The techniques that adapt the motor parameters during the drive operations are called "online parameter identification methods" [1]. Many online identification algorithms with different nature have been proposed in earlier literature and they can be categorized in four groups:

Spectral Analysis

In spectral analysis techniques the motor parameters are determined from a measured response from a deliberate injection test signal or an existing characteristic harmonic in the voltage/current spectrum [1]. The stator current and/or voltage are sampled and analysed with spectral analysis and the motor parameters are then derived.

Observer Based System

The well known Extended Kalman Filter(EKF) can be used as an observer for online parameter identification of a PMSM and is especially suitable in a noisy environment. Other observers can also be used but the most frequently used observers today seem to be based on the Kalman Filter. The full order EKF is sometimes replaced by a reduced order system due to the fact the the full order version is very computationally demanding [1]. The Recursive Least Square (RLS) algorithm that we are going to discuss in detail in Chapter 5 is a special case of the Kalman Filter.

Model Reference Adaptive System

This method is based on principles of model reference adaptive control. The Model reference adaptive system (MRAS) techniques are relatively simple to implement and are not as computationally demanding as other methods. Because of this the MRAS has attracted the most attention in the literature [1]. The principle is built on the fact that some quantities can be calculated in two ways. The first are calculated from a model and the others from measured signals. A few parameters are assumed to be known in the model and the difference between the quantities calculated from the model and the measured ones is assigned to the parameter errors from the unknown parameters in the model. The errors in the output signals are then used to drive an adaptive mechanism that provides a correction of the unknown parameters. Figure 1.1 illustrates a plant and control system in MRAS based identification of PMSM parameters. The drawback is that the method is only suitable in stationary operation where the other parameters that are not going to be adapted can be assumed to have constant known values. The method does not work well in standstill and with low torque. In [2], a MRAS based method is proposed for online identification of PMSM parameters. Online estimation of the rotor resistance with MRAS is successfully applied to a sensorless control algorithm in [3]. A comparison between the MRAS and EKF is treated in [4, 5, 6]. The conclusion from those papers is that a MARS method using decoupling of the stator currents has the best characteristics and the executing time is also much shorter compared to the EKF.

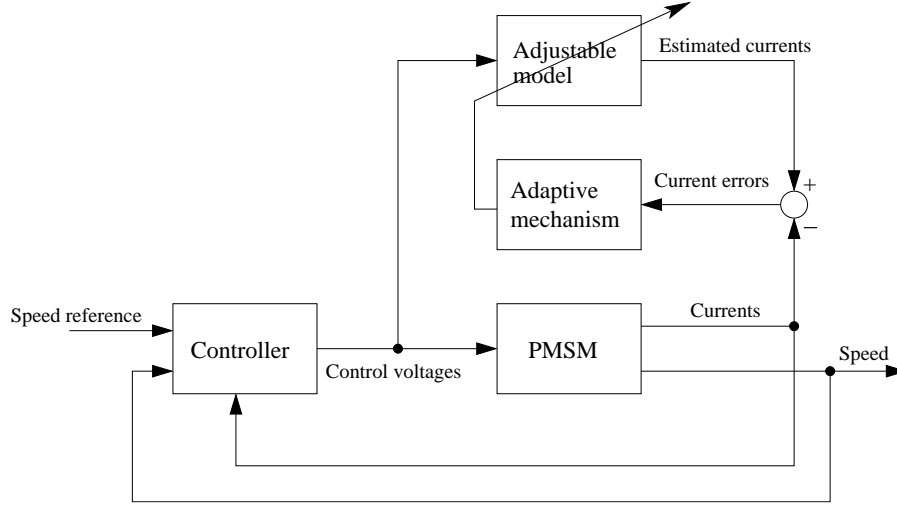


Figure 1.1: MRAS based identification of PMSM parameters

Other Methods

There are a number of other methods for identification of PMSM parameters and many of them come from the field of artificial intelligence (AI). In MRAS the adapting mechanism is mostly linear and for parameter identification in a non linear system neural networks (NN) have shown to give good results. A MRAS algorithm using NN in the adaptation mechanism is proposed in [7]. Another paper that proposes a NN based algorithm is [8]. Particle swarm optimization based parameter identification applied to PMSM is proposed in [9] and a fuzzy hybrid swarm optimization algorithm is presented in [10].

1.4 Outline of Thesis

In this thesis, two identification methods for time varying electrical permanent magnet synchronous motor (PMSM) parameters are studied and compared. The RLS with forgetting factor and the Normalized Projection

Algorithm (NPA) are applied to a salient PMSM model in dq -coordinates. The stator resistance and the stator inductances are identified both online and offline

In the second chapter, mathematical modelling of PMSMs for control purposes is derived for both salient and non salient PMSMs. The idea is to describe the salient model, derived from the non model, of the motor that is used in the parameter identification algorithms in the following chapters. In order to give the reader some crucial background knowledge, this chapter starts with introducing the concept of three phase systems and space vectors.

In the third chapter, the concept of vector control within motor drives is introduced. The intention is to give an overview of the PMSM control system used in this thesis, as well as present applications of online parameter identification within vector control.

In Chapter 4, the non salient model is discretized and transformed into a linear regression model. This makes the model suitable for standard parameter identification algorithms.

The fifth chapter presents and derives the two algorithms used later in the thesis to estimate the motor parameters, namely the RLS with forgetting factor and the NPA. The intention is not only to present the algorithms, but also give the reader some background knowledge of how the algorithms can be derived.

The sixth chapter concerns simulations of the RLS with forgetting factor and the NPA used for estimations of the motor parameters. First, the simulation procedure is described and then the results from the simulations are presented.

In order to further verify the performance of the proposed identification methods, in Chapter 7, recorded data from a testbed PMSM drive sys-

tem similar to the simulation set-up in Chapter 5 have been used in MATLAB/SIMULINK together with the algorithms. A complexity analysis of the RLS with forgetting factor and the NPA have also been done to investigate if the algorithms are suitable for implementation in existing DSP:s at Aros electronics AB.

In the last chapter, the conclusions from the thesis are presented, and some future work is suggested.

1.5 Contributions

The main scientific contribution in the thesis, is in the comparison of the parameter identification algorithms RLS with forgetting factor and the NPA within PMSM drives. The general properties of the RLS and the NPA are well treated and compared in [11]. The RLS with forgetting factor for identification of synchronous reluctance motor parameters is proposed in [12, 13, 14]. In this thesis, the RLS with forgetting factor is compared with the NPA for tracing electrical PMSM parameters both online and offline.

Chapter 2

Modelling of Permanent Magnet Synchronous Motors

In this chapter, mathematical modelling of PMSMs for control purposes is derived for both salient and non salient PMSMs. The idea is to describe the salient model, derived from the non salient model, of the motor that is used in the parameter identification algorithms in the following chapters. In order to give the reader some crucial background knowledge, this chapter starts with introducing the concept of three phase systems and space vectors.

2.1 Three Phase Systems

Three phase systems are very common for powering electronic motors [15]. The basic idea is to use three sinusoidal AC voltages separated by 120° for each phase [16]. Figure 2.1 shows the phase currents of an ideal three phase system. For an ideal symmetric system the sum of the instantaneous values are zero and the power is constant. The three phase currents have the

following relations:

$$i_{sa} + i_{sb} + i_{sc} = 0. \quad (2.1)$$

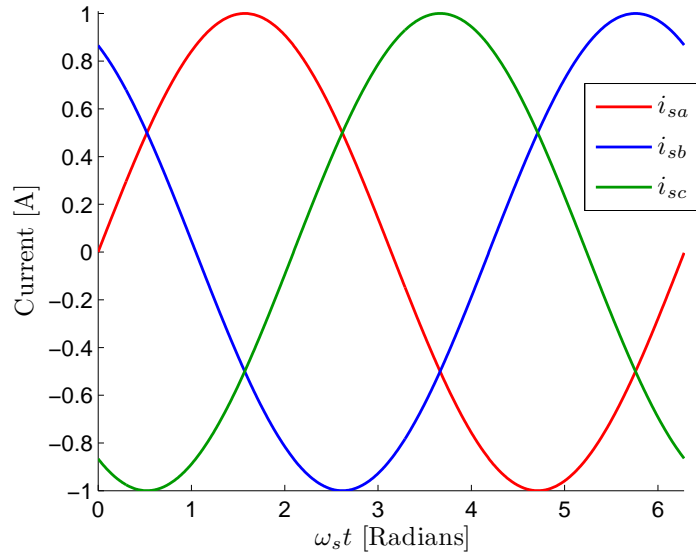


Figure 2.1: Ideal three phase currents

2.2 Space Vectors

It is possible to describe the phase currents by introducing a space vector. The coordinate system for the space vector is called $\alpha\beta$ coordinates [15]. The relation between the three phase currents and space vector is as follows:

$$\mathbf{i}_s^s(t) = i_{sa}(t) + j i_{sb}(t) = \frac{2}{3} [i_{sa}(t) + i_{sb}(t)e^{j\gamma} + i_{sc}(t)e^{j2\gamma}] \quad \text{with } \gamma = \frac{2\pi}{3} \quad (2.2)$$

Figure 2.2 illustrates how this is done. Now we have a complex vector circulating with the stator frequency ω_s that is a projection of the three physical

Figure 2.2: Construction of a current space vector.

phase currents. Other three phase quantities are obtained in a similar way by introducing space vectors. The space vector can equally be represented on matrix form as:

$$\begin{bmatrix} i_{s\alpha}(t) \\ i_{s\beta}(t) \end{bmatrix} = \underbrace{\begin{bmatrix} \frac{2}{3} & -\frac{1}{3} & -\frac{1}{3} \\ 0 & \frac{1}{\sqrt{3}} & -\frac{1}{\sqrt{3}} \end{bmatrix}}_{T_{32}} \begin{bmatrix} i_{sa}(t) \\ i_{sb}(t) \\ i_{sc}(t) \end{bmatrix} \quad (2.3)$$

The backward transformation from the space vector to the three phase quantities is given by:

$$\begin{bmatrix} i_{sa}(t) \\ i_{sb}(t) \\ i_{sc}(t) \end{bmatrix} = \underbrace{\begin{bmatrix} 1 & 0 \\ -\frac{1}{2} & \frac{\sqrt{3}}{2} \\ -\frac{1}{2} & -\frac{\sqrt{3}}{2} \end{bmatrix}}_{T_{23}} \begin{bmatrix} i_{s\alpha}(t) \\ i_{s\beta}(t) \end{bmatrix}. \quad (2.4)$$

If we have a symmetric three phase system with the currents as

$$\begin{bmatrix} i_{sa}(t) \\ i_{sb}(t) \\ i_{sc}(t) \end{bmatrix} = \begin{bmatrix} \hat{I}_s \cos(\omega_s t + \phi) \\ \hat{I}_s \cos(\omega_s t - 2\pi/3 + \phi) \\ \hat{I}_s \cos(\omega_s t - 4\pi/3 + \phi) \end{bmatrix} \quad (2.5)$$

the space vector will then be

$$\mathbf{I}_s^s = \hat{I}_s e^{j\omega_s t + \phi} \quad (2.6)$$

which corresponds to a rotating phasor aligned with phase a .

When it comes to control the speed and torque in a PMSM drive system, the three phase motor currents need to be controlled somehow with the help of the three phase voltages fed to the PMSM. By projecting the three phase voltages and currents into space vectors, and then introducing a new rotating coordinate system called dq coordinates [16], rotating with the stator frequency ω_s , the quantities will not be seen to rotate at all. The three phase currents and voltages transformed into dq coordinates can be seen as DC components in steady state.

With

$$\theta_s = \int_0^t \omega_s dt \quad (2.7)$$

the equation for the dq transformation is given by

$$\mathbf{i}_s = e^{-j\theta(t)} \mathbf{i}_s^s = i_{sd}(t) + j i_{sq}(t) \quad (2.8)$$

and the reverse transformation back to $\alpha\beta$ coordinates is then

$$\mathbf{i}_s^s = e^{j\theta(t)} \mathbf{i}_s = i_{s\alpha}(t) + j i_{s\beta}(t). \quad (2.9)$$

It is not always possible to use complex space vectors when it comes to digital implementation of the control algorithms (and also when modelling a salient PMSM as we will do later on). The corresponding real valued vector to \mathbf{i}_s^s is donated as i_s^s and we have the following equality:

$$\mathbf{i}_s^s = i_{s\alpha}(t) + j i_{s\beta}(t) \iff i_s^s = \begin{bmatrix} i_{s\alpha}(t) \\ i_{s\beta}(t) \end{bmatrix}. \quad (2.10)$$

By introducing a matrix

$$J = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \quad (2.11)$$

that corresponds to the imaginary unit j , the equation for the dq transformation is then given by

$$i_s = e^{-J\theta(t)} i_s^s \quad (2.12)$$

and the reverse transformation back to $\alpha\beta$ coordinates is then

$$i_s^s = e^{J\theta(t)} i_s. \quad (2.13)$$

We also have that

$$e^{-J\theta(t)} = \underbrace{\begin{bmatrix} \cos \theta_s & \sin \theta_s \\ -\sin \theta_s & \cos \theta_s \end{bmatrix}}_{T_{dq}(\theta_s)} \quad (2.14)$$

and

$$e^{-J\theta(t)} = \underbrace{\begin{bmatrix} \cos \theta_s & -\sin \theta_s \\ \sin \theta_s & \cos \theta_s \end{bmatrix}}_{T_{\alpha\beta}(\theta_s)}. \quad (2.15)$$

The matrices $T_{dq}(\theta_s)$ and $T_{\alpha\beta}(\theta_s)$ are frequently used in Digital Signal Processors (DSPs) when doing the dq transformation.

2.3 Modelling of a non salient PMSM

The stator voltage to the non salient PMSM is given by

$$\mathbf{v}_s^s = R_s \mathbf{i}_s + \frac{d\psi_s^s}{dt} \quad (2.16)$$

and the equation for the stator flux is

$$\psi_s^s = L_s \mathbf{i}_s + \psi_m^s. \quad (2.17)$$

ψ_m^s is the rotor flux linkage vector and is given by

$$\psi_m^s = \psi_m e^{j\theta_r} \quad (2.18)$$

where the ψ_m is the rotor flux linkage from the permanent magnets, θ_r is the electrical rotor angle and L_s is the stator inductance. By assuming that the linkage inductance and the flux linkage from the permanent magnet are

varying slowly [16], the derivative of the stator flux is then

$$\frac{d\psi_s^s}{dt} = L_s \frac{d\mathbf{i}_s^s}{dt} + j\omega_r \psi_m e^{j\theta_r} \quad (2.19)$$

where ω_r is the electrical rotor frequency. Putting equation (2.19) into (2.16) gives

$$L_s \frac{d\mathbf{i}_s^s}{dt} = \mathbf{v}_s^s - R_s \mathbf{i}_s^s - j\omega_r \psi_m e^{j\theta_r}. \quad (2.20)$$

Assuming perfect field orientation, the equation (2.20) in the dq coordinates is then

$$L_s \frac{d\mathbf{i}_s}{dt} = \mathbf{v}_s - (R_s + j\omega_r L_s) \mathbf{i}_s - j\omega_r \psi_m. \quad (2.21)$$

Splitting up equation (2.21) in d and q directions gives

$$u_{sd} = R_s i_{sd} + L_s \frac{di_{sd}}{dt} - \omega_r L_s i_{sq} \quad (2.22)$$

$$u_{sq} = R_s i_{sq} + L_s \frac{di_{sq}}{dt} + \omega_r L_s i_{sd} + \omega_r \psi_m. \quad (2.23)$$

In the next section, the salient PMSM model equations are derived from 2.22 and 2.23.

2.4 Modelling of a salient PMSM

For a salient PMSM the stator inductance is not equal around the airgap. The coordinates of a salient PMSM are illustrated in Figure 2.3. In dq coordinates the stator inductance is equal to L_d on the d axis and equal to L_q on the q axis [16].



$$u_{sd} = R_s i_{sd} + L_d \frac{di_{sd}}{dt} - \omega_r L_q i_{sq} \quad (2.24)$$

$$u_{sq} = R_s i_{sq} + L_q \frac{di_{sq}}{dt} + \omega_r L_d i_{sd} + \omega_r \psi_m. \quad (2.25)$$

Equation 2.24 and 2.25 can also be written in state space form as

$$\frac{d}{dt} \begin{bmatrix} i_d \\ i_q \end{bmatrix} = \begin{bmatrix} -\frac{R_{sd}}{L_d} & \frac{\omega_r L_q}{L_d} \\ -\frac{\omega_r L_d}{L_q} & -\frac{R_{sq}}{L_q} \end{bmatrix} \begin{bmatrix} i_d \\ i_q \end{bmatrix} + \begin{bmatrix} \frac{1}{L_d} & 0 \\ 0 & \frac{1}{L_q} \end{bmatrix} \begin{bmatrix} u_d \\ u_q - \omega_r \psi_m \end{bmatrix}. \quad (2.26)$$

The electrical torque T_e is given by [16] as

$$T_e = \frac{3n_p}{2}[\psi_m i_{sq} + (L_d - L_q)i_{sd}i_{sq}] \quad (2.27)$$

where n_p is the number of pole pairs of the motor (which is well described in [17]). According to [16] the Speed of the rotor can further be expressed as

$$\frac{J_{ml}}{n_p} \frac{d\omega_r}{dt} = T_e - T_L. \quad (2.28)$$

J_{ml} here represents the inertia of the motor plus load, and the motor load T_L is given by

$$T_L = \frac{B_{ml}}{n_p} \omega_r + T_{L,extra}. \quad (2.29)$$

B_{ml} is the viscous friction of the motor and load and $T_{L,extra}$ is the load connected to the motor.

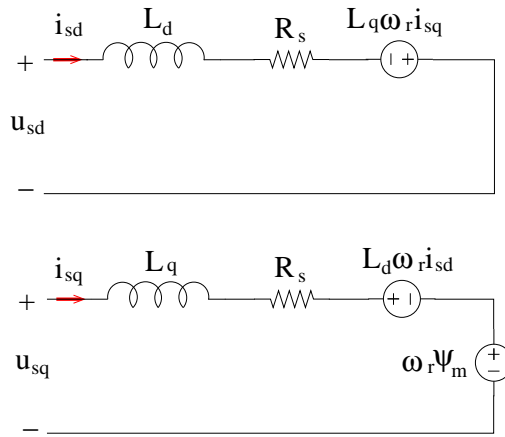


Figure 2.4: Dynamic equivalent circuits for the salient PMSMs electrical subsystem.

Chapter 3

Speed and Current Vector Control

In this chapter, the concept of vector control within motor drives is introduced. The intention is to give an overview of the PMSM control system used in this thesis, as well as present a motivation of online parameter identification within sensorless vector control.

3.1 Vector Control

Vector control is today the most common ac machine control technique and can be applied for both induction machines (IM) and synchronous machines (SM). The method is widely used in high performance industrial application of electric drives. The basic idea is to force the AC machine to behave dynamically as a DC machine by the use of field orientation and feedback control [16]. A PMSM can be seen as a DC motor turned inside out and after field orientation is obtained the control design for a DC motor can be used.

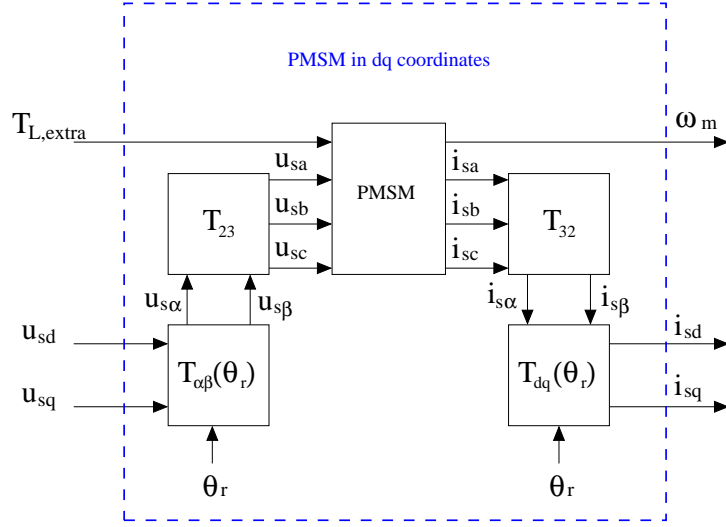


Figure 3.1: Transformation of the PMSM model into dq coordinates.

The procedure for transforming the PMSM model into dq coordinates is as follow:

- Measures the three phase currents and projects them into a space vector in the $\alpha\beta$ -reference frame.
- Transforms the current vector in the $\alpha\beta$ -reference frame in to the dq -reference frame.
- Transforms the desired dq -frame stator voltage in to the $\alpha\beta$ -reference frame.
- Transforms the $\alpha\beta$ -frame stator voltage to three phase voltages. The three phase voltages are then used to form the control signal to the pulse with modulation(PWM) of the inverter (power electronic amplifier).

This transformation of the PMSM model into dq coordinates without considering the inverter and PWM, is illustrated in Figure 3.1 (theory of inverters and PWM can be found in [16, 15]). The decoupling of flux and torque is

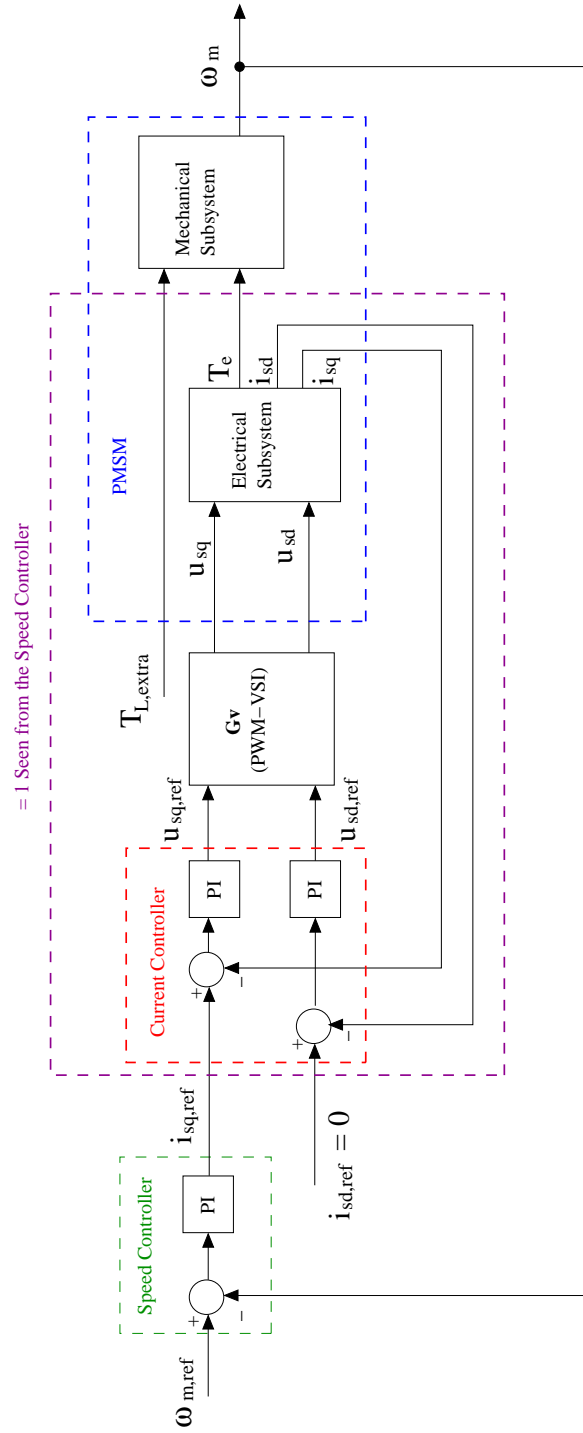


Figure 3.2: Vector control system with cascaded current and speed control loops.

done by letting the d -axis be aligned with ψ_m (see Figure 2.3). Now, the input and output to the drive system are in the dq -frame and the PMSM behaves as a DC motor in steady state. All torque lies in the q -axis direction and by varying the stator current component i_q the torque can be changed. The flux is changed by varying the current component i_d . The control system to the PMSM can now be designed with inner model control techniques. First a current control loop is added and then cascade coupled with a speed control loop as in Figure 3.2

3.2 Motivation for Online Parameter Identification within Sensorless Vector Control

The basic vector control algorithms require a position sensor to control the motor currents, and since a sensor brings high cost and sometimes low reliability it is desirable to remove the sensor [18]. The concept of sensorless vector control is basically vector control without any speed or position sensor. In this control strategy, the speed and position need to be determined by an observer [1]. In salient PMSM drives the observer is normally built on the electrical motor equations (2.24) and (2.25). This means that if the electrical PMSM parameters do not match the actual values, a speed and position error will occur which affects the control performance of the PMSM. By introducing online parameter identification of electrical PMSM parameters it is possible to design accurate adaptive speed and position observers to sensorless vector control systems. Figure 3.3 illustrates control and system plant for a sensorless vector control PMSM system based on an adaptive speed and position observer with online identification of electrical PMSM parameters.

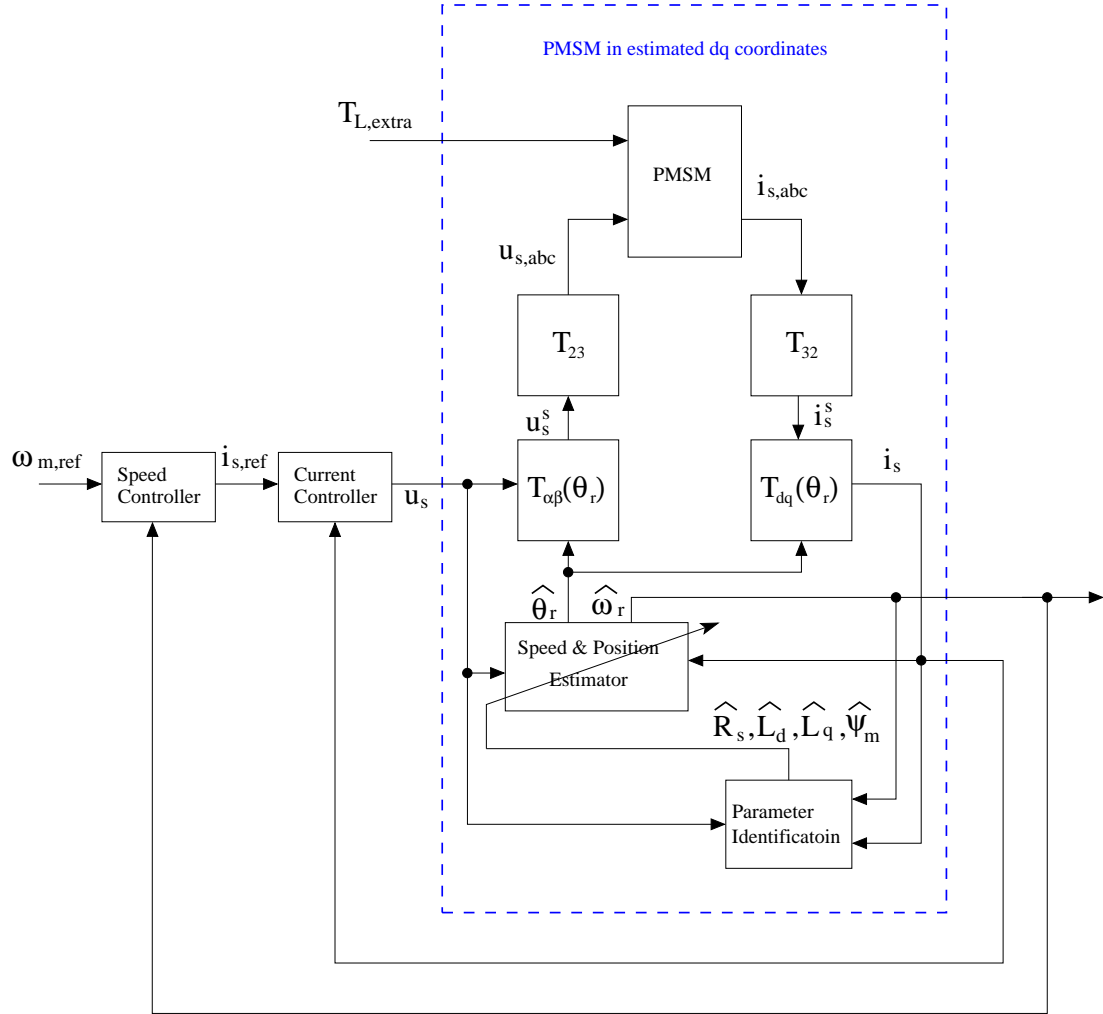


Figure 3.3: Online identification of electrical PMSM parameters within sensorless vector control.

Chapter 4

Discretization and Linear Regression Models

In this chapter, the non salient model is discretized and transformed into a linear regression model. This makes the model suitable for standard parameter identification algorithms.

4.1 Discretization

Since the control system is implemented in a digital signal processor (DSP), a discretized model of the PMSM is necessary. If a general continuous time invariant system given in the state space form

$$\begin{aligned}\frac{dx}{dt} &= Ax(t) + Bu(t) \\ y(t) &= Cx(t)\end{aligned}\tag{4.1}$$

is zero-order-hold (ZOH) sampled with period h , the system equation will then be

$$\begin{aligned}x(kh + h) &= A_d(kh) + B_d u(kh) \\ y(kh) &= C_d x(kh)\end{aligned}\tag{4.2}$$

where

$$\begin{aligned}A_d &= e^{Ah} \\ B_d &= \int_0^h e^{As} ds B \\ C_d &= C.\end{aligned}\tag{4.3}$$

It is necessary to approximate the matrices A_d and B_d . One way to do that is to introduce a matrix Ψ [19] defined as

$$\Psi = \int_0^s h^{As} ds \simeq \sum_{i=0}^n \frac{A^i h^{i+1}}{(i+1)!}\tag{4.4}$$

and then calculate A_d and B_d as

$$\begin{aligned}A_d &= I + A\Psi \\ B_d &= \Psi B.\end{aligned}\tag{4.5}$$

When n goes to infinity the approximation error will go to zero. The approximation where $n = 0$ is equal to the Euler Forward difference and corresponds to replacing the derivatives with

$$\frac{dx(t)}{dt} \simeq \frac{x(t+h) - x(t)}{h}.\tag{4.6}$$

Now, by writing the electrical space equation (4.7) for the salient PMSM as

$$\frac{d}{dt} \begin{bmatrix} i_{sd} \\ i_{sq} \end{bmatrix} = \underbrace{\begin{bmatrix} a_{11} & a_{12} \\ a_{12} & a_{22} \end{bmatrix}}_A \begin{bmatrix} i_{sd} \\ i_{sq} \end{bmatrix} + \underbrace{\begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{bmatrix}}_B \begin{bmatrix} u_{sd} \\ \bar{u}_{sq} \end{bmatrix}\tag{4.7}$$

where

$$\begin{aligned}
 A &= \begin{bmatrix} a_{11} & a_{12} \\ a_{12} & a_{22} \end{bmatrix} = \begin{bmatrix} -\frac{R_{sd}}{L_d} & \frac{\omega_r L_q}{L_d} \\ -\frac{\omega_r L_d}{L_q} & -\frac{R_{sq}}{L_q} \end{bmatrix} \\
 B &= \begin{bmatrix} b_{11} & b_{12} \\ b_{12} & b_{22} \end{bmatrix} = \begin{bmatrix} \frac{1}{L_d} & 0 \\ 0 & \frac{1}{L_q} \end{bmatrix} .
 \end{aligned} \tag{4.8}$$

$$\bar{u}_{sq} = u_{sq} - \omega_r \psi_m$$

the discretization model with Euler Forward difference gives

$$\begin{bmatrix} i_{sd}(t+1) \\ i_{sq}(t+1) \end{bmatrix} = \underbrace{\begin{bmatrix} a_{11_d} & a_{12_d} \\ a_{12_d} & a_{22_d} \end{bmatrix}}_{A_d} \begin{bmatrix} i_d(t) \\ i_q(t) \end{bmatrix} + \underbrace{\begin{bmatrix} b_{11_d} & b_{12_d} \\ b_{21_d} & b_{22_d} \end{bmatrix}}_{B_d} \begin{bmatrix} u_d(t) \\ \bar{u}_{sq}(t) \end{bmatrix} \tag{4.9}$$

were

$$\begin{aligned}
 A_d &= \begin{bmatrix} a_{11_d} & a_{12_d} \\ a_{12_d} & a_{22_d} \end{bmatrix} = \begin{bmatrix} a_{11}h + 1 & a_{12}h \\ a_{12}h & a_{22}h + 1 \end{bmatrix} = \begin{bmatrix} -\frac{R_{sd}}{L_d}h + 1 & \frac{\omega_r L_q}{L_d}h \\ -\frac{\omega_r L_d}{L_q}h & -\frac{R_{sq}}{L_q}h + 1 \end{bmatrix} \\
 B_d &= \begin{bmatrix} b_{11_d} & b_{12_d} \\ b_{12_d} & b_{22_d} \end{bmatrix} = \begin{bmatrix} b_{11}h & b_{12}h \\ b_{12}h & b_{22}h \end{bmatrix} = \begin{bmatrix} \frac{1}{L_d}h & 0 \\ 0 & \frac{1}{L_q}h \end{bmatrix}
 \end{aligned} \tag{4.10}$$

4.2 Linear Regressions

The model structure

$$y(t) = \varphi(t)^T \theta \tag{4.11}$$

is well known in the statistic as liner regressions, assuming that (4.11) is linear in θ . The vector $\varphi(t)$ is named linear regression vector, and the elements in $\varphi(t)$ are called regessors [20]. The name comes from the fact that we try to calculate/describe $y(t)$ by going back (regress) to $\varphi(t)$. The vector θ is

called parameter vector and is used to parametrize the model. θ consists of model parameters. For the multivariable case θ is a matrix. The algorithms for parameter identification that are going to be used later need to have the discretized motor model on linear regression form. Time shifting equation (4.9) one step backwards in time gives:

$$\begin{bmatrix} i_{sd}(t) \\ i_{sq}(t) \end{bmatrix} = \underbrace{\begin{bmatrix} a_{11_d} & a_{12_d} \\ a_{21_d} & a_{22_d} \end{bmatrix}}_{A_d} \begin{bmatrix} i_{sd}(t-1) \\ i_{sq}(t-1) \end{bmatrix} + \underbrace{\begin{bmatrix} b_{11_d} & b_{12_d} \\ b_{21_d} & b_{22_d} \end{bmatrix}}_{B_d} \begin{bmatrix} u_{sd}(t-1) \\ \bar{u}_{sq}(t-1) \end{bmatrix}. \quad (4.12)$$

This model can then be written on linear regression form as:

$$\underbrace{\begin{bmatrix} i_{sd}(t) & i_{sq}(t) \end{bmatrix}}_{y(t)} = \underbrace{\begin{bmatrix} i_{sd}(t-1) & i_{sq}(t-1) & u_{sd}(t-1) & u_{sq}(t-1) \end{bmatrix}}_{\varphi^T(t)} \underbrace{\begin{bmatrix} a_{11_d} & a_{21_d} \\ a_{12_d} & a_{22_d} \\ b_{11_d} & b_{21_d} \\ b_{12_d} & b_{22_d} \end{bmatrix}}_{\theta}. \quad (4.13)$$

With Euler Forward difference, the motor parameters can be driven from θ as:

$$\begin{aligned} R_s &= \frac{2 - a_{11_d} - a_{22_d}}{b_{11_d} + b_{22_d}} \\ L_d &= \frac{h}{b_{b11}} \\ L_q &= \frac{h}{b_{b22}} \end{aligned} \quad (4.14)$$

Chapter 5

Recursive Parameter Estimation

This chapter presents and derives the two algorithms used later in the thesis to estimate the motor parameters, namely recursive least square algorithm with forgetting factor and the normalized projection algorithm. The intention is not only to present the algorithms, but also give the reader some background knowledge of how the algorithms can be derived.

5.1 The Least Square Method

Assume that the model is written on linear regression form and that we want to estimate the parameter matrix θ from the measured variable vector $y(t)$ and the regression vector $\varphi(t)$ in the sense of least squares. Then we can introduce a loss function

$$V(\theta, t) = \frac{1}{2} \sum_{i=1}^t (y(i) - \hat{y}(i))^2 = V(\theta, t) = \frac{1}{2} \sum_{i=1}^t (y(i) - \varphi^T(i)(\theta))^2. \quad (5.1)$$

By minimizing $V(\theta, t)$ with respect to θ we get the least square error. This can be done in many ways. One way is to set the derivative to zero due to the fact that $V(\theta, t)$ is quadratic in θ [20]. This gives

$$0 = \frac{d}{d\theta} V(\theta, t) = \sum_{i=1}^t \varphi(i)(y(i) - \varphi^T(i)\theta) \quad (5.2)$$

and then we have

$$\hat{\theta} = \left[\sum_{i=1}^t \varphi(i)\varphi^T(i) \right]^{-1} \sum_{i=1}^t \varphi(i)y(i) \quad (5.3)$$

where $\hat{\theta}$ is the estimated parameter matrix. Equation (5.3) can then be solved by numerical algorithm from recorded data containing $\varphi(t)$ and $y(t)$ over a time interval. It is also possible to make the computations recursively in real time.

5.2 Recursive Least Square Algorithm

For identification of parameters in real time, the algorithm is done recursively. Measurement from time $t-1$ is used to predict parameters at time t . By introducing a new matrix $P(t)$ [20] defined as

$$P^{-1}(t) = \sum_{i=1}^t \varphi(i)\varphi^T(i) = \underbrace{\sum_{i=1}^{t-1} \varphi(i)\varphi^T(i)}_{P^{-1}(t-1)} + \varphi(t)\varphi^T(t), \quad (5.4)$$

equation (5.3) can be expressed as

$$\hat{\theta}(t) = P(t) \sum_{i=1}^t \varphi(i)y(i) = P(t) \left(\sum_{i=1}^{t-1} \varphi(i)y(i) + \varphi(t)y(t) \right). \quad (5.5)$$

From equations (5.3) and (5.4) we also have that

$$\sum_{i=1}^{t-1} \varphi(i)y(i) = P^{-1}(t-1)\hat{\theta}(t-1) = P^{-1}(t)\hat{\theta}(t-1) - \varphi(t)\varphi(t)^T\hat{\theta}(t-1). \quad (5.6)$$

And the parameter matrix $\hat{\theta}(t)$ can then be written as

$$\hat{\theta}(t) = \hat{\theta}(t-1) + P(t)\varphi(t)(y(t) - \varphi(t)^T\hat{\theta}(t-1)) = \hat{\theta}(t-1) + L(t)\epsilon(t) \quad (5.7)$$

where

$$L(t) = P(t)\varphi(t) \quad (5.8)$$

$$\epsilon(t) = y(t) - \varphi(t)^T\hat{\theta}(t-1). \quad (5.9)$$

To calculate $P(t)$ from $P(t-1)$ the matrix inversion lemma is used [11]:

$$(A + BCD)^{-1} = A^{-1} - A^{-1}B(C^{-1} + DA^{-1}B)^{-1}DA^{-1}. \quad (5.10)$$

Equation (5.4) implies that

$$P(t) = (P^{-1}(t-1) + \varphi(t)\varphi^T(t))^{-1} = (P^{-1}(t-1) + \varphi(t)I\varphi^T(t))^{-1}, \quad (5.11)$$

and applying the matrix inversion lemma gives

$$P(t) = P(t-1) - P(t-1)\varphi(t)(I + \varphi^T(t)P(t-1)\varphi(t))^{-1}\varphi^T(t)P(t-1). \quad (5.12)$$

By combining equations (5.8) with (5.12) and applying the matrix inversion lemma again, $L(t)$ can be calculated as

$$L(t) = P(t)\varphi(t) = P(t-1)\varphi(t)(I + \varphi(t)^T P(t-1)\varphi(t))^{-1} \quad (5.13)$$

which implies that

$$P(t) = (I - L(t)\varphi^T(t))P(t-1). \quad (5.14)$$

The recursive scheme to calculate the parameter matrix is given bellow:

<ol style="list-style-type: none"> 1. $\varepsilon(t) = (y(t) - \varphi^T(t)\hat{\theta}(t-1))$ 2. $L(t) = P(t-1)\varphi(t)(I + \varphi(t)^T P(t-1)\varphi(t))^{-1}$ 3. $P(t) = (I - L(t)\varphi^T(t))P(t-1)$ 4. $\hat{\theta}(t) = \hat{\theta}(t-1) + L(t)\varepsilon(t)$ 5. $t = t + 1$ 6. Go to step 1. 	(5.15)
--	--------

5.3 Recursive Least Square with Forgetting Factor

In the scheme (5.15) the parameter matrix θ is assumed to be time invariant and for the case where we have time varying parameters, the algorithm needs to be extended. Recursive least square with forgetting factor is an extended version that can handle slow time varying parameters [11]. The idea is to weight data depending on how old it is. The most recent data will have the highest weight and the oldest data will have the lowest weight. This can be

done by replicating the loss function (5.1) with

$$V(\theta, t) = \frac{1}{2} \sum_{i=1}^t \lambda^{t-i} [y(i) - \varphi^T(i)(\theta)]^T [y(i) - \varphi^T(i)(\theta)]. \quad (5.16)$$

This loss function also includes the multivariable case. The variable λ is called the forgetting factor and is in the range $]0, 1]$. Now, the most recent data is weighted with λ and data that is n units old is weighted with λ^n . With the same approach as for the non multivariable case without forgetting factor a recursive scheme can be obtained. A recursive scheme including the multivariable case and forgetting factor is given below:

<ol style="list-style-type: none"> 1. $\varepsilon(t) = [y(t) - \varphi^T(t)\hat{\theta}(t-1)]$ 2. $L(t) = P(t-1)\varphi(t)[\lambda I + \varphi(t)^T P(t-1)\varphi(t)]^{-1}$ 3. $P(t) = \frac{1}{\lambda} [P(t-1) - L(t)\varphi^T(t)P(t-1)]$ 4. $\hat{\theta}(t) = \hat{\theta}(t-1) + L(t)\varepsilon(t)$ 5. $t = t + 1$ 6. Go to step 1. 	(5.17)
---	--------

5.4 Normalized Projection Algorithm

In the RLS algorithm the updating of the P matrix takes most computational effort. Several simplified algorithms that do not need to update the P matrix are proposed in the literature. In this way the computational effort decreases. One of the algorithms is called normalized projection algorithm (NPA) or simply projection algorithm [11]. The estimate $\hat{\theta}(t)$ is here chosen so that $\|\hat{\theta}(t) - \hat{\theta}(t-1)\|$ is a minimized subject to the constraint $y(t) = \varphi^T(t)\hat{\theta}(t)$. The constraint can be handled by introducing a Lagrangian multiplier $\bar{\alpha}$.

The loss function that is going to be minimized is given by

$$V = \frac{1}{2}[\hat{\theta}(t) - \hat{\theta}(t-1)]^T[\hat{\theta}(t) - \hat{\theta}(t-1)] + \bar{\alpha}[y(t) - \varphi(t)^T \hat{\theta}(t)]. \quad (5.18)$$

Putting derivatives with respect to $\hat{\theta}(t)$ and $\bar{\alpha}$ to zero and solving the equations gives the updating formula

$$\hat{\theta}(t) = \hat{\theta}(t-1) + \frac{\varphi(t)}{\varphi^T(t)\varphi(t)}[y(t) - \varphi^T(t)\hat{\theta}(t-1)] \quad (5.19)$$

Multiplying the nominator with a factor γ and replace the denominator $\varphi^T\varphi$ to $\varphi^T\varphi + \alpha$ makes it possible to change the step length of the parameter adjustment and potential problem with $\varphi = 0$ is avoided. These changes to the updating formula (5.19) gives the final NPA as

$$\hat{\theta}(t) = \hat{\theta}(t-1) + \frac{\gamma\varphi(t)}{\alpha + \varphi(t)^T\varphi(t)}[y(t) - \varphi^T(t)\hat{\theta}(t-1)] \quad (5.20)$$

where $\alpha \geq 0$ and $0 < \gamma < 2$ to get the algorithm stable. The recursive scheme to calculate the parameter matrix with the NPA is given bellow:

<ol style="list-style-type: none"> 1. $\varepsilon(t) = [y(t) - \varphi^T(t)\hat{\theta}(t-1)]$ 2. $\hat{\theta}(t) = \hat{\theta}(t-1) + \frac{\gamma\varphi(t)}{\alpha + \varphi(t)^T\varphi(t)}\varepsilon(t)$ 3. $t = t + 1$ 4. Go to step 1. 	(5.21)
--	--------

Chapter 6

Simulations

This chapter concerns simulations of the RLS with forgetting factor and the NPA used for estimations of the motor parameters. First, the simulation procedure is described and then the results from the simulations are presented.

6.1 Simulation procedure

A three phase PMSM with salient poles is simulated in MATLAB/SIMULINK to investigate the performance of the RLS with forgetting factor and NPA algorithms applied to the electrical motor parameter identification. In the simulation, the controller to the PMSM drive system with cascaded PI-controllers as in Figure 3.2 is used. A similar drive system is used in the implementation presented in chapter 5. The mechanical parameters of the simulated PMSM are chosen identical to the test motor data sheet parameters. The nominal electrical motor parameters L_d , L_q and R_s that are identified in the simulations are measured direct on the motor windings. ψ_m is also measured and assumed to be known in all simulations.

Figure 6.1 shows the inner current loop of the vector control system to-

gether with the identification part. $u_{sq,x}$ and $u_{sd,x}$ represent the identification signals. In order to identify the electrical motor parameters, the identification signals must ensure that u_{sq}^* and u_{sd}^* satisfy the condition of persistent excitation [20]. The identification signals are here binary signals that switches between $\pm 5V$ ($\simeq \pm 30\%$ of the voltage limit to the motor) with a probability of 20 percent in each sample. The measured disturbances are represented in the simulation by n_d and n_q . They are created from Gaussian noise signals with variance 1 multiplied with 1.5 percent of the actual current values. In all simulations the sampling time $h = 0.25\text{ms}$. The initiated P matrix and λ for the RLS are selected as follows:

$$\lambda = 0.99;$$

$$P = \begin{bmatrix} 0.1 & 0 & 0 & 0 \\ 0 & 0.1 & 0 & 0 \\ 0 & 0 & 0.1 & 0 \\ 0 & 0 & 0 & 0.1 \end{bmatrix} \quad (6.1)$$

For the NPA γ is chosen to:

$$\gamma = 0.01 \quad (6.2)$$

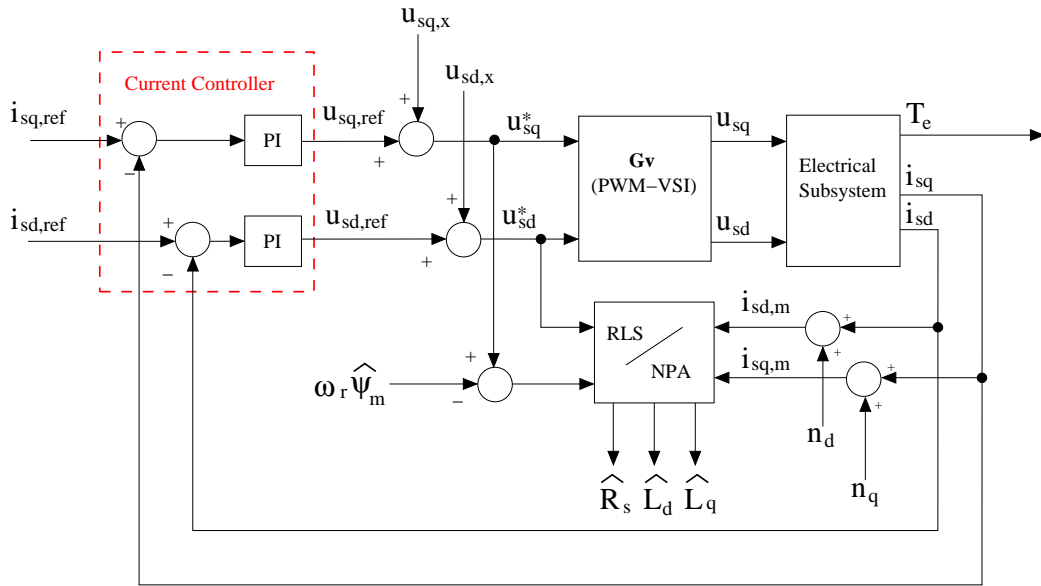


Figure 6.1: Inner current loop of the vector control system together with the identification part.

6.2 Simulation Results

All combinations of start positions of the parameter matrix θ with

$$\begin{aligned} R_{s0} &= (1 \pm 0.5)R_s \\ L_{d0} &= (1 \pm 0.5)L_d \\ L_{q0} &= (1 \pm 0.5)L_q \end{aligned} \tag{6.3}$$

have been tested with the two algorithms used as commission offline parameter identification methods. The motor is in standstill with non load and in each case both algorithms converge. The efficiency is quite clear, but it can be seen that the NPA is less efficient than the RLS as expected. The settling time for the NPA is about 1.2 seconds and for the RLS 0.02 seconds, so the RLS is 6 times faster. Both methods converge to nearly the same parameter values, but the NPA gives a large overshoot or undershoot when estimating R_s . Figure 6.2 illustrates simulation results where the start position of θ is chosen so that $R_{s0} = 0.5R_s$, $L_{d0} = 0.5L_d$ and $L_{q0} = 0.5L_q$. Figure 6.3 illustrates simulation results when the rotor speed $\omega_r = 600[rpm]$ instead, the same results are obtained. Table 6.1 shows the magnitude mean errors for the estimated parameters that are valid for both algorithms. Figure

	Mean Error
L_d	$\leq 5\%$
L_q	$\leq 4\%$
R_s	$\leq 0.7\%$

Table 6.1: Magnitude mean error for the parameters when RLS and NPA are used as commission offline parameter identification methods

6.4 illustrates a simulation where the algorithms are used as online parameter estimation methods. R_s , L_d and L_q are here defined as variables of

times that change sinusoidally. The PMSM is loaded with a constant torque $T_L = 0.3Nm$. The RLS tracks the parameter variations much better compared to the NPA and especially when it comes to R_s . Table 6.2 shows the maximal magnitude of the parameter error for the two algorithms in this simulation. Figure 6.5 shows a simulation where the load torque and parameters

	$(\text{Max Error})_{RLS}$	$(\text{Max Error})_{NPA}$
L_d	$\leq 18\%$	$\leq 42\%$
L_q	$\leq 16\%$	$\leq 32\%$
R_s	$\leq 12\%$	$\leq 70\%$

Table 6.2: Maximal magnitude of the parameter error for the RLS and NPA when the parameter changes sinusoidally

are changed with a step during the simulation. After 0.3s T_L increases from 0.3Nm to 0.5Nm by a step, L_d decrease with 5%, L_q decrease with 50% and R_s increase with 40%. The settling time is around 0.08s for both algorithms after the steps are introduced. The two algorithms do not converge to the same parameter values when estimating L_q and R_s . NPA also gives a very noisy estimate of R_s . Table 6.3 shows the steady state mean value of the parameters after the first step at 0.3s. Figure 6.6 shows a simulation quite

	$(\text{Mean Error})_{RLS}$	$(\text{Mean Error})_{NPA}$
L_d	$\leq 6\%$	$\leq 6\%$
L_q	$\leq 10\%$	$\leq 71\%$
R_s	$\leq 0.5\%$	$\leq 4.7\%$

Table 6.3: Steady state mean value of the parameters after the first step at 0.3s of simulation in Figure 6.5. T_L increases from 0.3Nm to 0.5Nm, L_d decrease with 5%, L_q decrease with 50% and R_s increase with 40%

similar to Figure 6.5, but now T_L decrease from 0.3Nm to 0.1Nm by a step, L_d increase with 5%, L_q increase with 10% and R_s decrease with 20% instead.

In this case, both algorithms convergence and the mean parameter errors in steady state are nearly the same. Table 6.4 shows the steady state mean value of the parameters after the first step at 0.3s for this simulation. The reason

	(Mean Error) $_{RLS}$	(Mean Error) $_{NPA}$
L_d	3.2%	3.3%
L_q	2.7%	2.6%
R_s	0.9%	-0.4%

Table 6.4: Steady state mean value of the parameters after the first step at 0.3s of simulation in Figure 6.6. T_L decrease from 0.3Nm to 0.1Nm by a step, L_d increase with 5%, L_q increase with 10% and R_s decrease with 20%

way the RLS convergence much slower when the parameters change stepwise during the time is because we have same memory stored from old parameter values in the covariance matrix $P(t)$ and the step size $L(t) = P(t)\varphi(t)$ is now much smaller compared to the initialization. RLS with forgetting factor is based of the assumption that the behaviour of the algorithm is homogeneous in time [11] and step changes in the parameters are therefore less suitable. If the forgetting factor λ is decreased, the algorithm will be faster due to the fact that $P(t)$ contains less information of previous date. The cost of that is noisier parameter estimation. In a situation with step changing (or very fast changing) parameters, it is more appropriate to reset the $P(t)$ to a large value matrix when the parameter changes occur. The problem here will be to detect the step changing in the parameters. Figure 6.6 illustrate a simulation similar to Figure 6.6, but now the the covariance matrix $P(t)$ is reset when a step change in the parameters occur. The mean parameter error after convergence is more or less the same, but settling time for the RLS is equally fast as for the offline case.

In Figure 6.8 and Figure 6.9 angular errors of the rotor flux position are introduced in the vector control system to investigate the robustness of

the algorithms when the exact rotor flux position is not known. It can be seen that if 5% angular error is introduced the parameter errors from the RLS nearly do not change at all. For the NPA we also have good performance, but the estimation of R_s becomes more noisy and the mean error changes from -0.4% to -1.5% (in same simulation the mean error of R_s does not change at all). Generally, for both algorithms, when the angular error increases the absolute value estimation error of R_s increases and becomes more noisy. Table 6.5 shows the mean parameter errors when the angular error is as much as 40%.

	(Mean Error) $_{RLS}$	(Mean Error) $_{NPA}$
L_d	12%	14%
L_q	-5%	0.5%
R_s	-8%	-3%

Table 6.5: Mean parameter errors when an angular rotor flux error of 40% is introduced in the vector control system

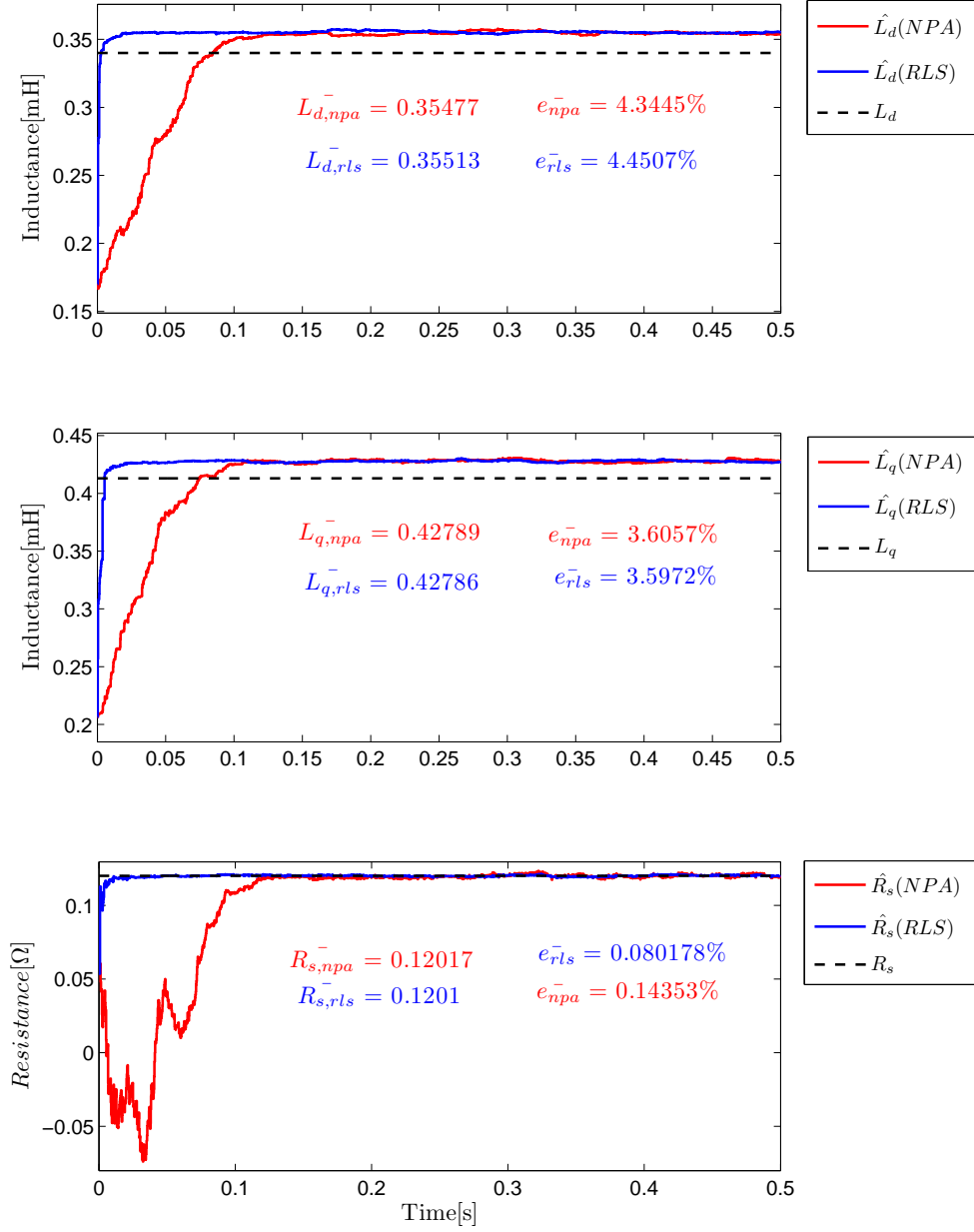


Figure 6.2: RLS and NPA used as commission offline parameter identification methods in stand still. The start values are: $L_{d0} = 0.5L_d$, $L_{q0} = 0.5L_q$, $R_{s0} = 0.5R_s$

6 Simulations

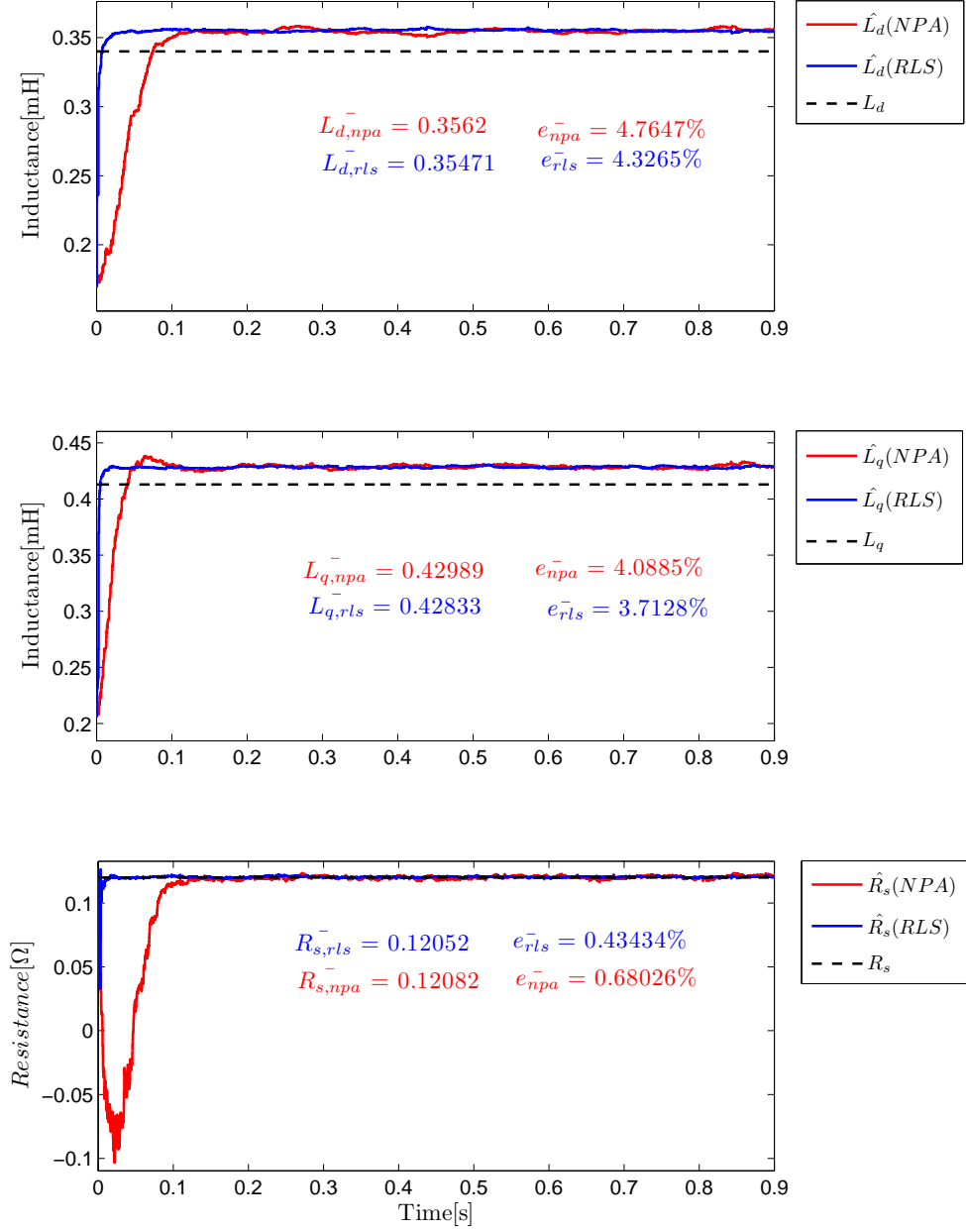


Figure 6.3: RLS and NPA used as commission offline parameter identification methods in speed 600[rpm] and no load. The start values are: $L_{d0} = 0.5L_d$, $L_{q0} = 0.5L_q$, $R_{s0} = 0.5R_s$

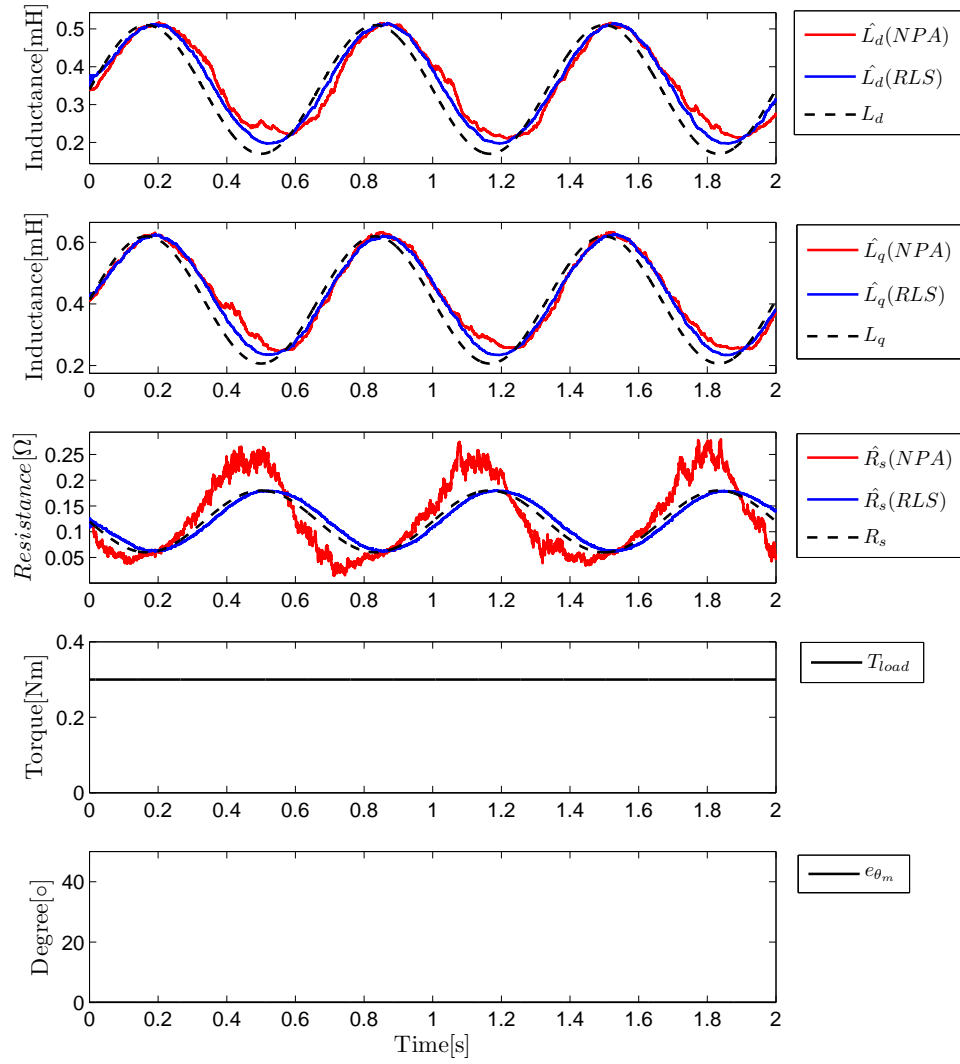


Figure 6.4: Simulations when the algorithms are used as online parameter estimation methods. R_s , L_d and L_q are here defined as variables of times that changes sinusoidal.

6 Simulations

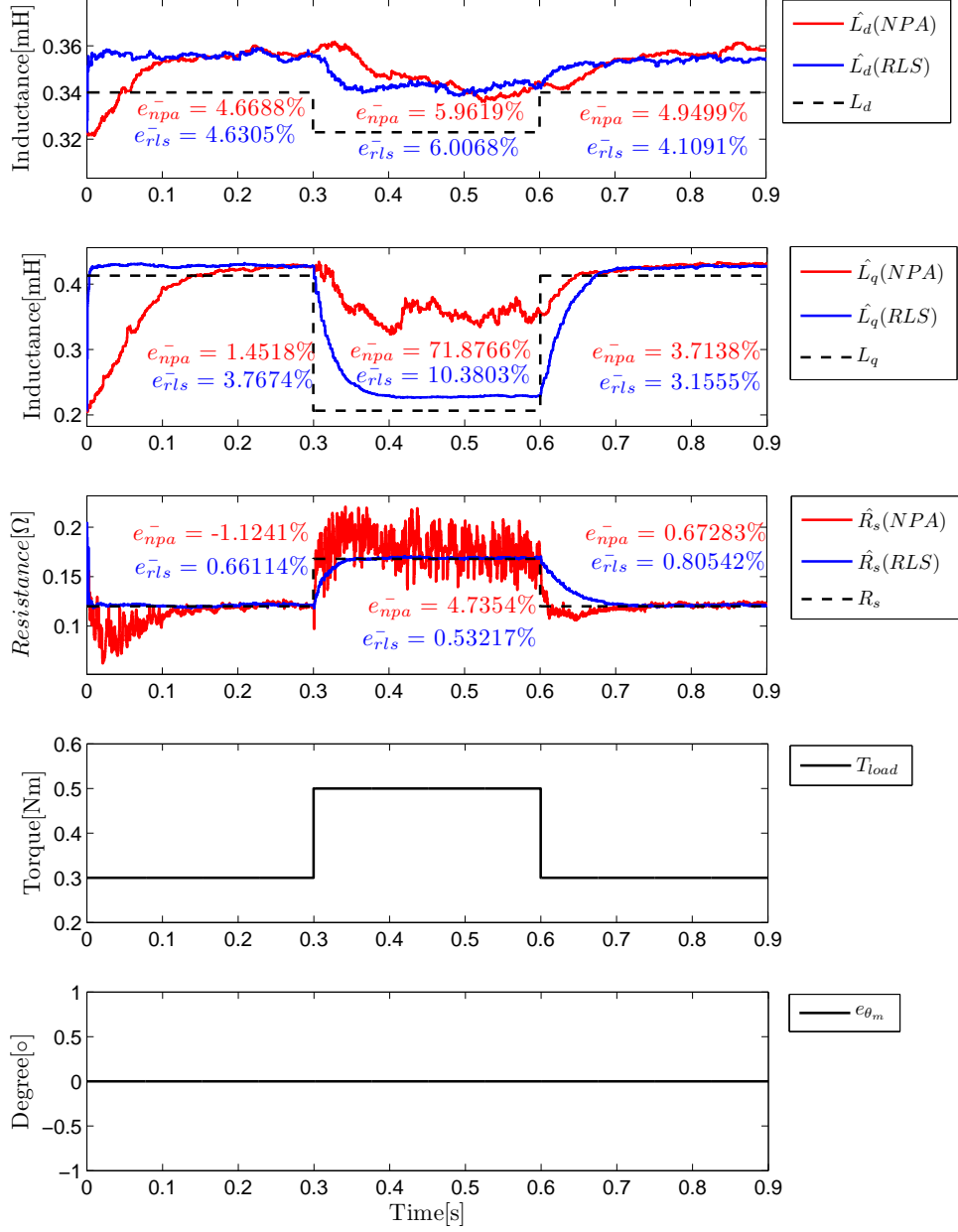


Figure 6.5: A simulation where the load torque and parameters are changed with steps. After 0.3s T_L increases from 0.3Nm to 0.5Nm by a step, L_d decreases with 5%, L_q decreases with 50% and R_s increases with 40% in the first step.

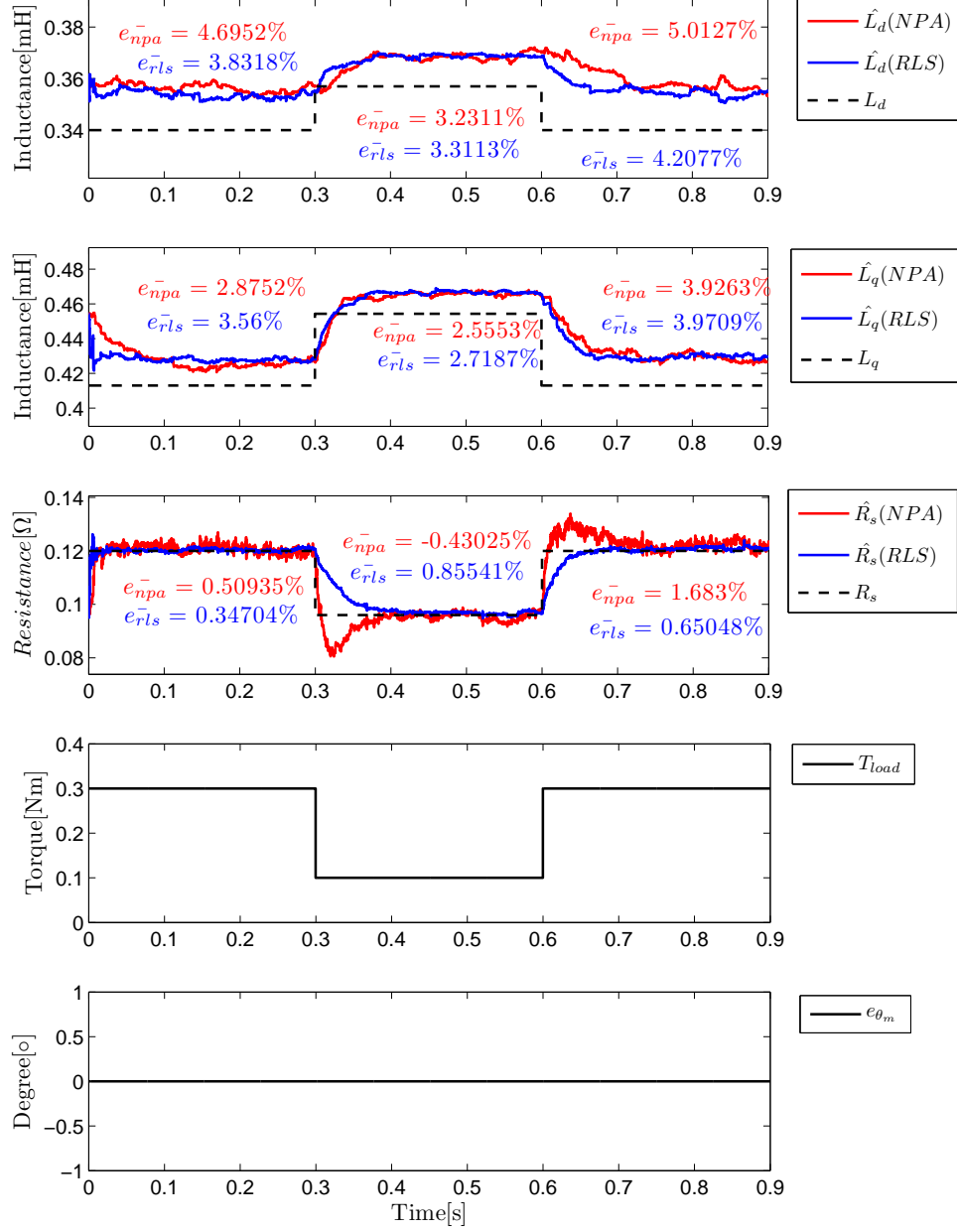


Figure 6.6: A simulation where the load torque and parameters are changed with steps. After 0.3s T_L decrease from 0.3Nm to 0.1Nm by a step, L_d increase with 5%, L_q increase with 10% and R_s decrease with 20%.

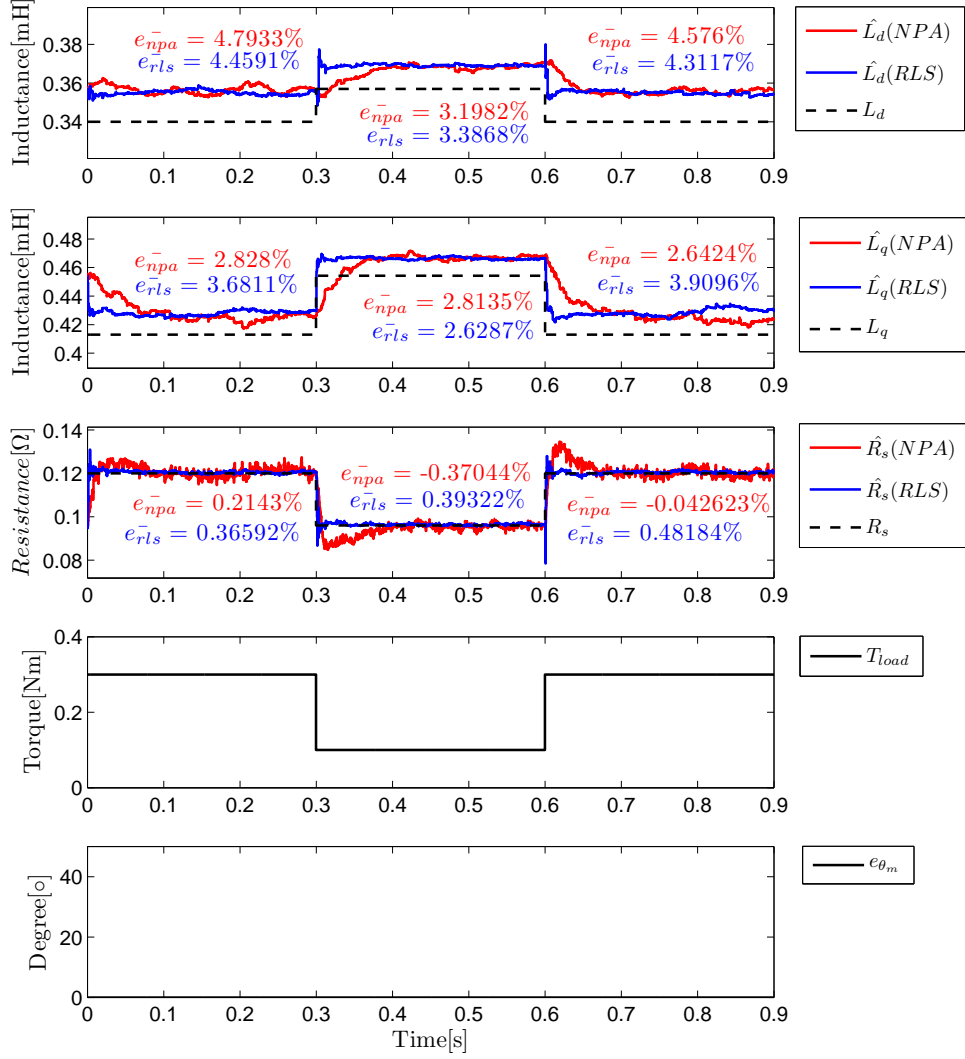


Figure 6.7: A simulation where the load torque and parameters are changed with steps similar to Figure 6.6, but now the covariance matrix $P(t)$ is retested when a step change in the parameters occur.

6.2 Simulation Results

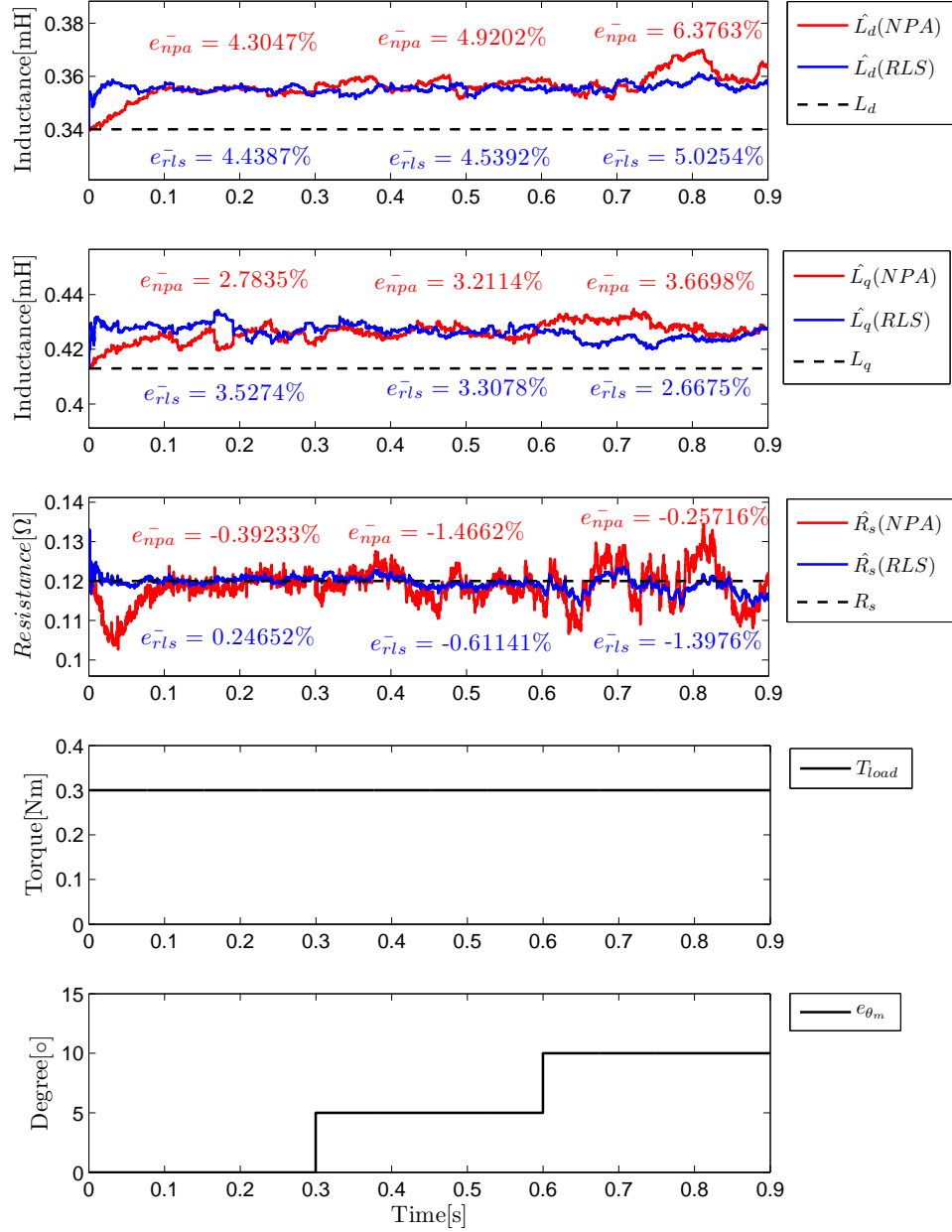


Figure 6.8: A simulation where an angular rotor flux error is introduced in the vector control system. The angular error increments from 0° to 10° .

6 Simulations

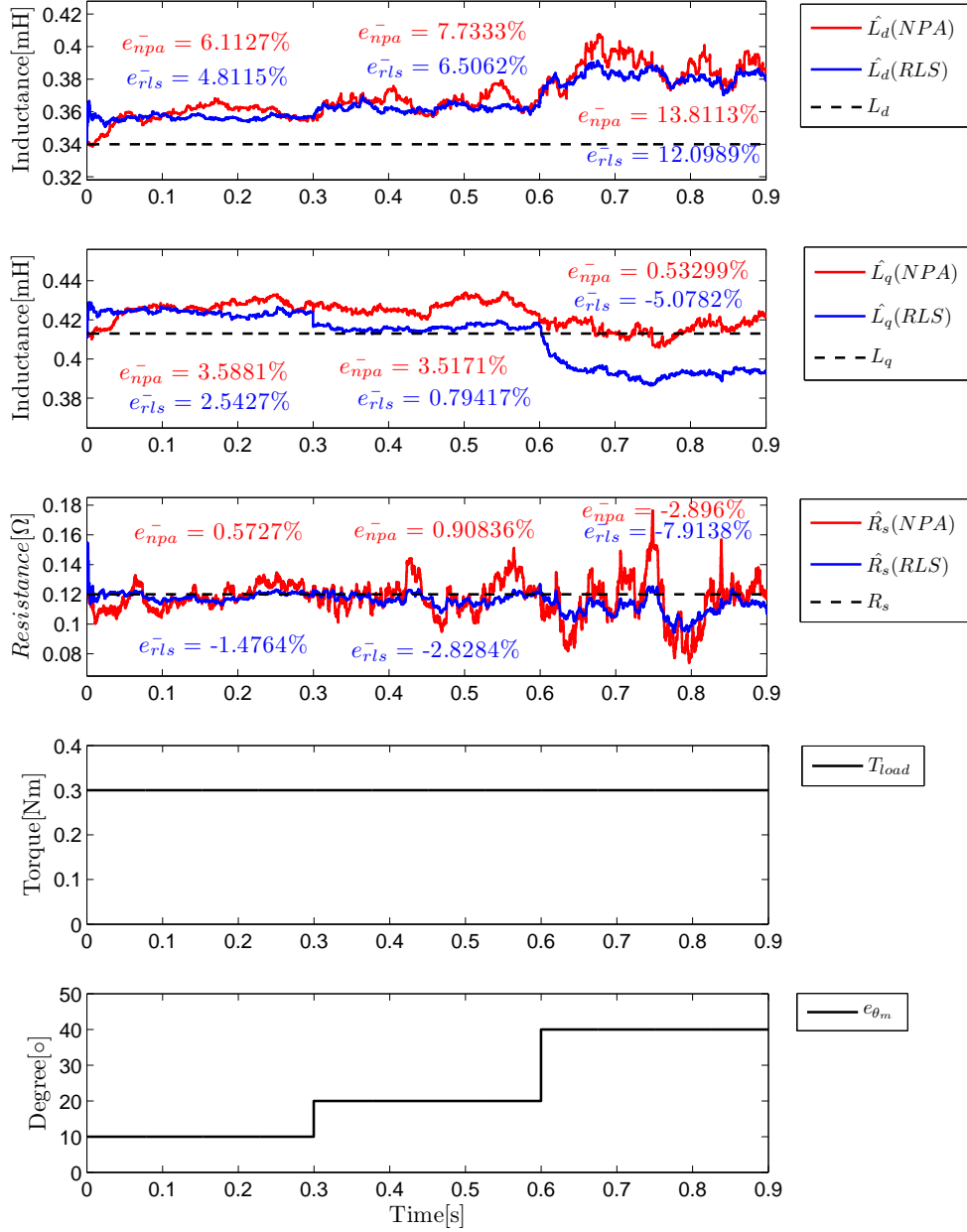


Figure 6.9: A simulation where an angular rotor flux error is introduced in the vector control system. The angular error increments from 10° to 40°

Chapter 7

Implementation

In order to further verify the performance of the proposed identification methods, recorded data from a testbed PMSM drive system similar to the simulation set-up in Chapter 5 have been used in MATLAB/SIMULINK together with the algorithms. A complexity analysis of the RLS with forgetting factor and the NPA have also been done to investigate if the algorithms are suitable for implementation in existing DSP:s at Aros electronics AB.

7.1 Implementation complexity

To investigate the implementation complexity of the two methods, the number of operations that are needed for each algorithm has been calculated. Table 7.1 shows the number of operations that are needed for implementation of the RLS and the NPA.

Computational time for both methods has been estimated from the number of operations that are required. Table 7.2 shows the estimated computational time for two DSP:s at Aros electronics AB.

	Multiplications	Additions	Divisions
NPA	36	14	1
RLS	104	72	2
Parameter Calculations	0	1	3

Table 7.1: Number of operations that are needed for implementation of the RLS and the NPA.

	DSP1	DSP2
NPA + Parameter Calculations	$24\mu s$	$4\mu s$
RLS + Parameter Calculations	$66\mu s$	$11\mu s$

Table 7.2: Estimated computational time for RLS and NPA.

7.2 Implementation Set-up

Figure 7.1 illustrates the implemented vector control system together with the identification signals. The inputs and outputs are recorded and stored in a memory. The recorded data are then used in MATLAB/SIMULINK together with the two algorithms.

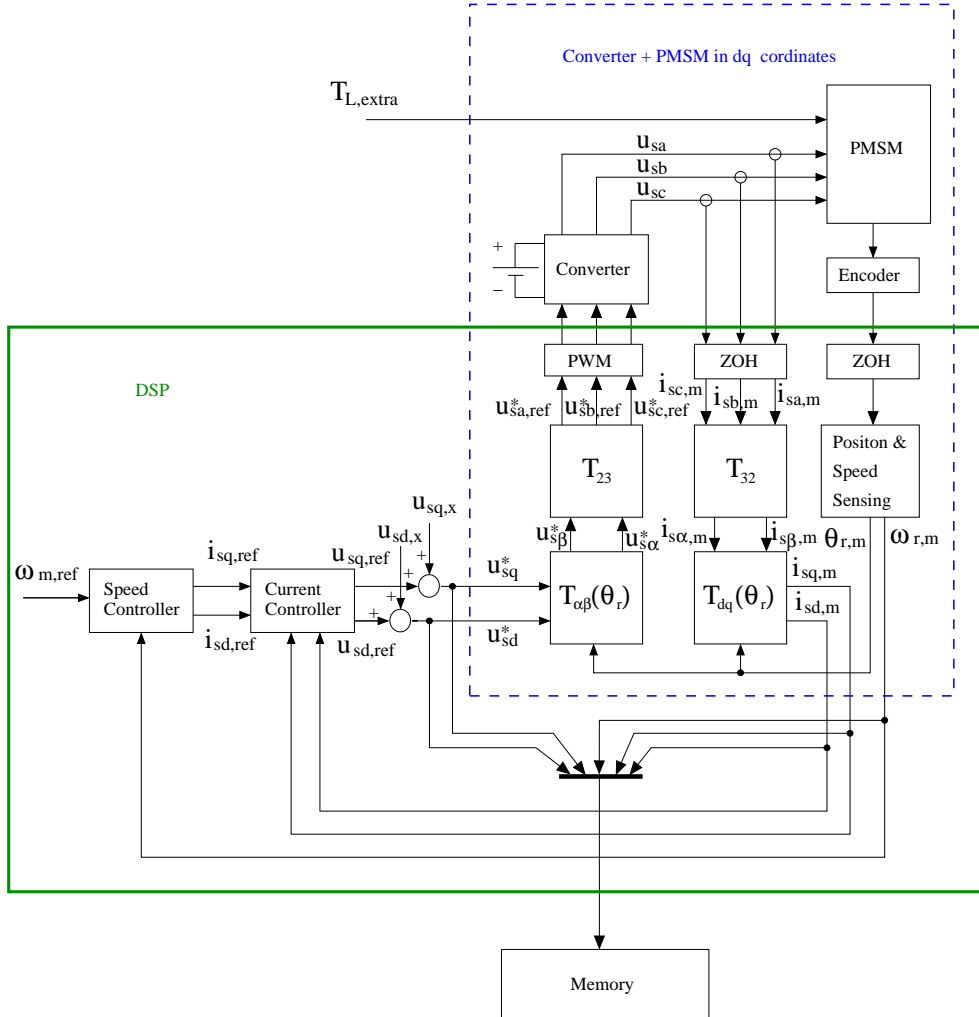


Figure 7.1: Vector control system together with the identification signals implemented in a DSP. The inputs and outputs are recorded and stored in a memory.

7.3 Experimental Results

In the experimental results, the recorded data from the realizations is used in MATLAB/SIMULINK in order to test the algorithms. The parameter variations due to different working points are assumed to be in the range

$$\begin{aligned} 0.8R_{s,m} < R_s < 1.4R_{s,m} \\ 0.95L_{d,m} < L_d < 1.05L_{d,m} \\ 0.5L_{q,m} < L_q < 1.1L_{q,m} \end{aligned} \tag{7.1}$$

where $R_{s,m}$, $L_{d,m}$ and $L_{q,m}$ correspond to the measured values.

Figure 7.2 shows the result from a realization where the identification signals $u_{sq,x}$ and $u_{sd,x}$ switches between $\pm 4V$ ($\simeq \pm 24\%$ of the voltage limit to the motor) with an approximate probability of 20 percent in each sample. The motor is non loaded and the mechanical rotor reference speed $\omega_{m,ref}$ is set to zero. The algorithms are here used as commission offline parameter identification methods. In this realization the loss function (5.1) converges to zero, so the quadratic error is therefore minimized. A theoretical parameter matrix θ has been calculated in MATLAB from equation (4.7) with the expected parameter values. The discretization method was ZOH ($n \rightarrow \infty$ in equation (4.4)) in order to minimize the approximation error. By comparing the estimated parameter matrix $\hat{\theta}$ with the theoretical θ , it is shown that all elements that are expected to be zero are close to zero and that the other elements have the right sign. Nothing here indicates bad quality of $\hat{\theta}$.

In order to test the algorithms as online methods, a realization where the $\omega_{m,ref} = 600[RPM]$ and $T_{L,extra} = 0.5[Nm]$ has further been done. The same identification signals $u_{sq,x}$ and $u_{sd,x}$ have been used as in the first realization (Figure 7.2). Results can be seen in Figure 7.3. In this realization, the loss

function (5.1) also converges to zero, but by comparing $\hat{\theta}$ with the theoretical θ (calculated in a similar way as in the first realization), it is shown that one of the elements has the opposite sign and that one of the other elements is close to zero which it should not be. This could indicate that $\hat{\theta}$ is fitted to some disturbance in the measured three phase currents and/or that the applied three phase voltages to the motor deviate from the reference voltages used as input to the algorithms (the three phase voltages are not measured).

Figure 7.4 illustrates another realization where $T_{L,extra} = 0.1[Nm]$, $\omega_{m,ref} = 600[RPM]$ and $u_{sq,x}$, $u_{sd,x}$ switch between $\pm 3V$ ($\simeq \pm 18\%$ of the voltage limit to the motor) with an approximate probability of 6 percent in each sample. Also here the loss function (5.1) converges to zero. Comparing $\hat{\theta}$ with the theoretical θ (calculated in a similar way as in the first realization) gives similar results as in the realisation illustrated in Figure 7.3 and the same conclusions can be drawn.

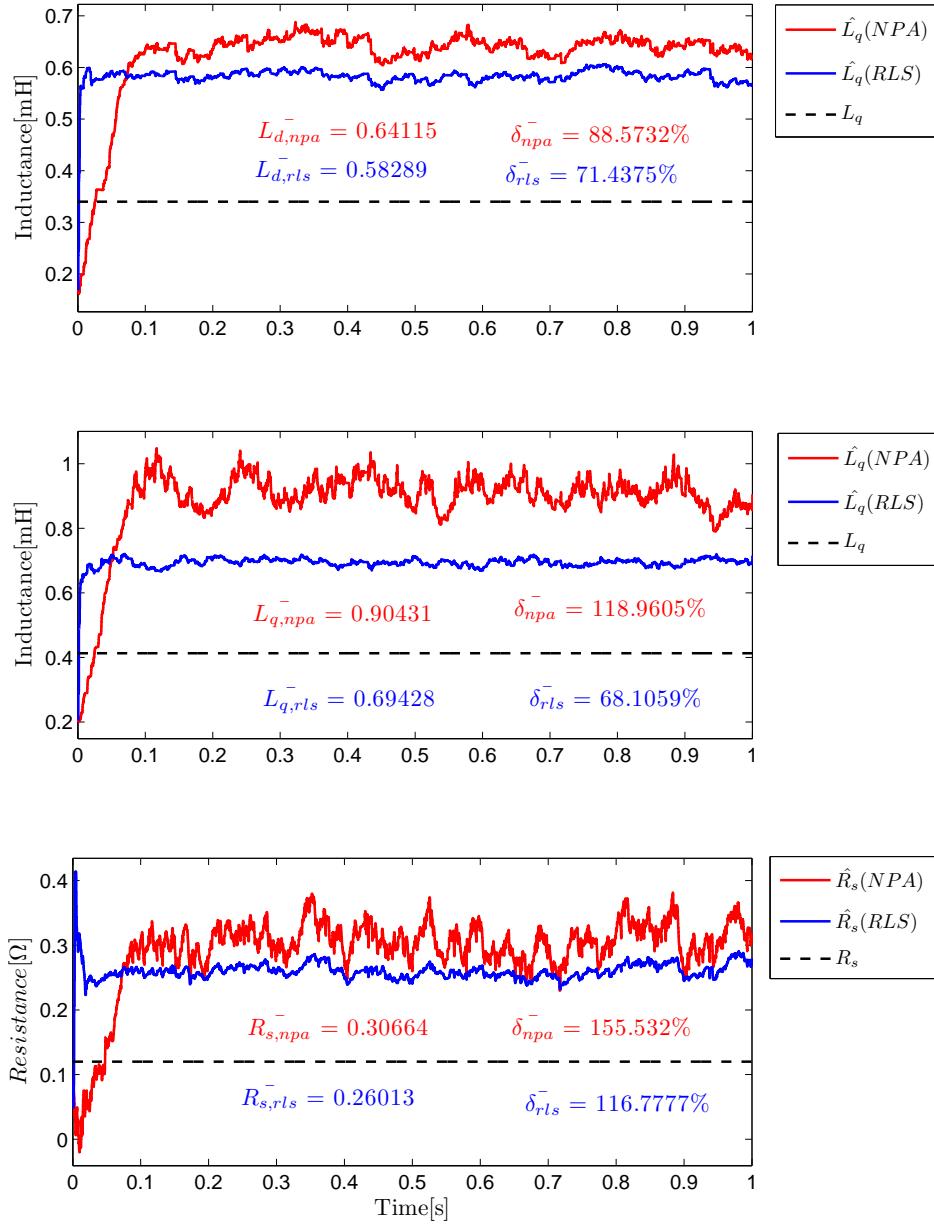


Figure 7.2: Result from a realization where $u_{sq,x}$ and $u_{sd,x}$ switches between $\pm 4V$ with a approximately probability of 20 percent in each sample. $T_{L,extra} = 0$ and $\omega_{m,ref} = 0$

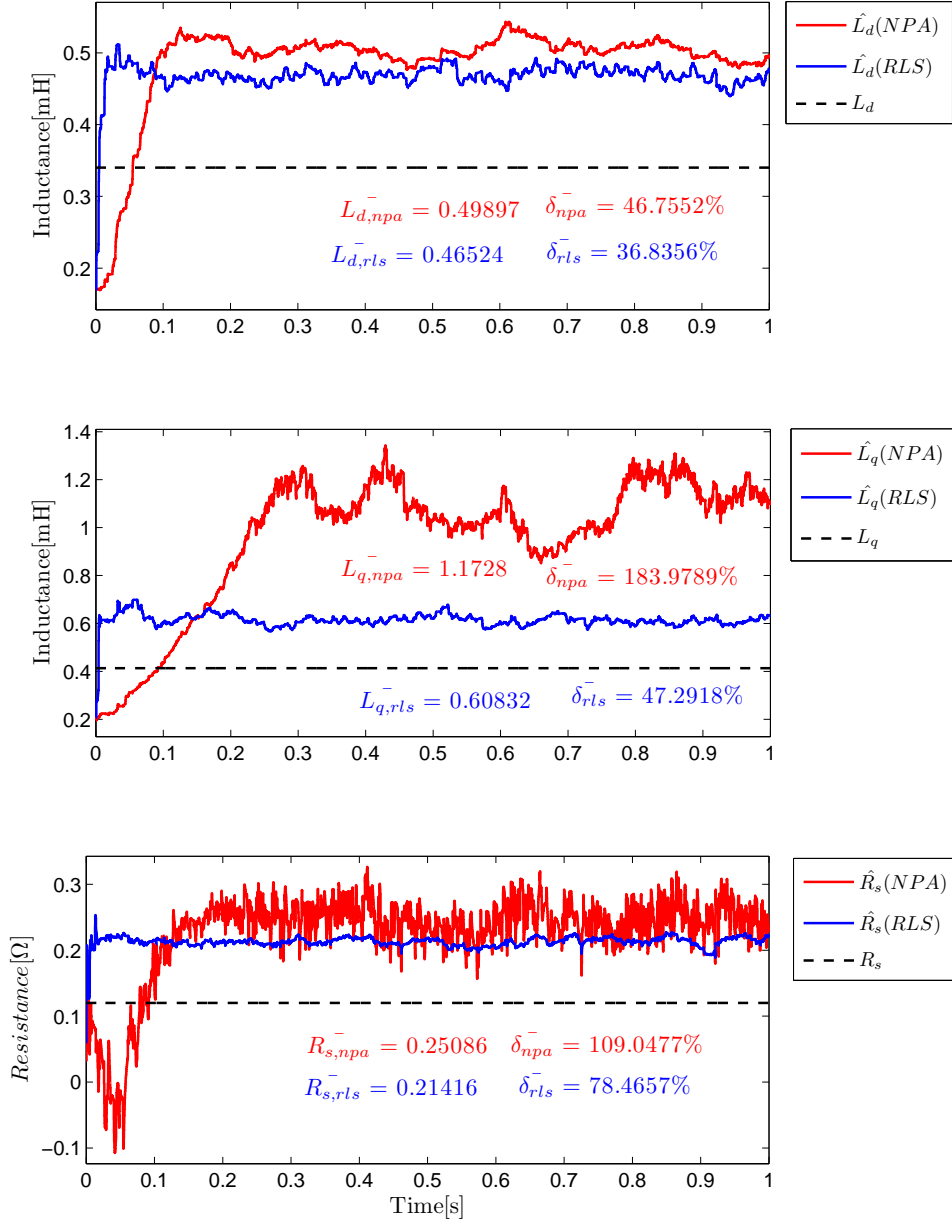


Figure 7.3: Result from a realization where $u_{sq,x}$ and $u_{sd,x}$ switches between $\pm 4V$ with a approximately probability of 20 percent in each sample, $T_{L,extra} = 0.5[Nm]$ and $\omega_{m,ref} = 600[RPM]$.

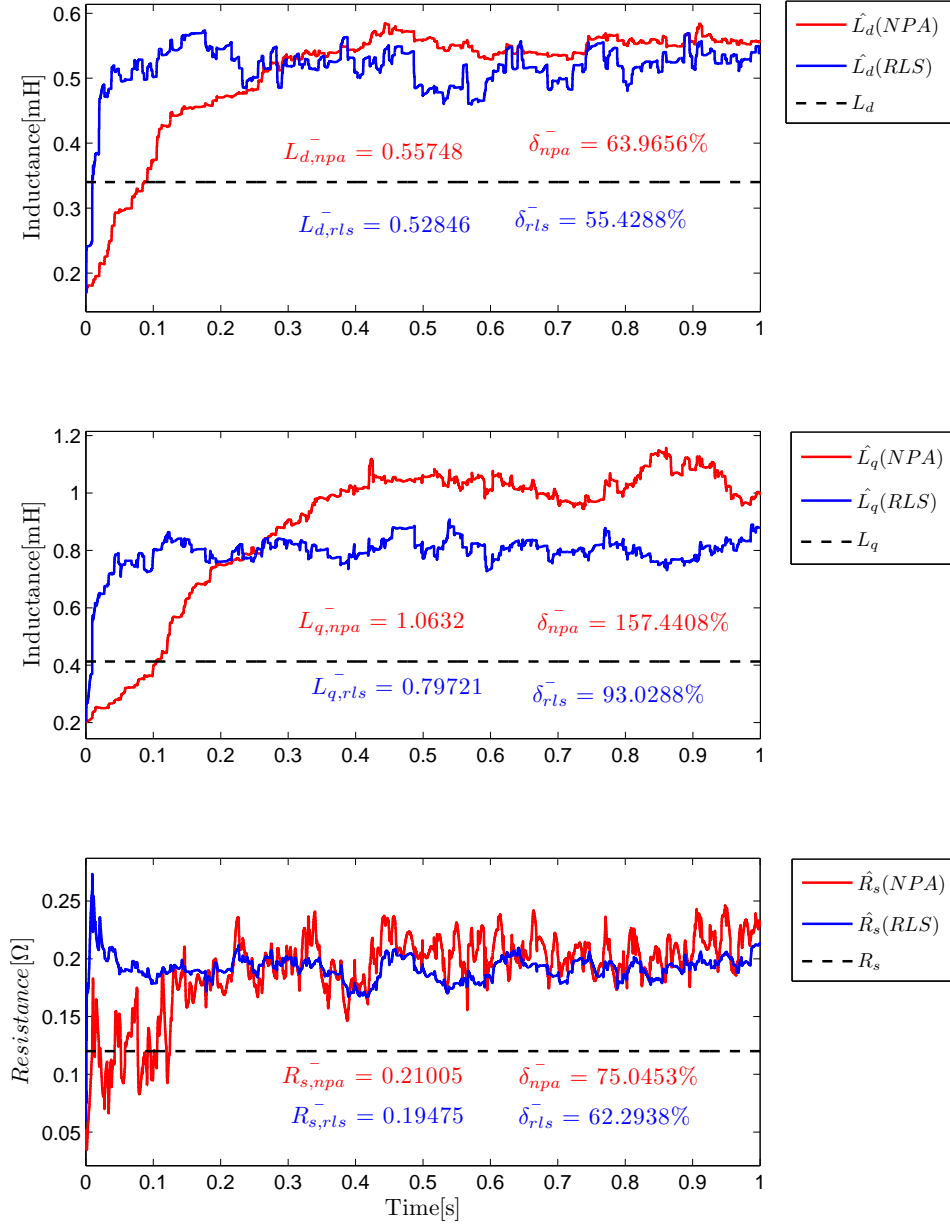


Figure 7.4: Result from a realization where $u_{sq,x}$ and $u_{sd,x}$ switches between $\pm 3V$ with a approximately probability of 6 percent in each sample, $T_{L,extra} = 0.1[Nm]$ and $\omega_{m,ref} = 600[RPM]$.

Chapter 8

Conclusion and Future Work

In this chapter, the conclusions from the thesis are presented, and some future work is suggested.

8.1 Conclusion

In this thesis, two identification methods for electrical PMSM parameters are studied and compared. The RLS with forgetting factor and the NPA are applied to the salient PMSM model in dq -coordinates. The electrical stator parameters R_s , L_d and L_q are identified both online and offline. The rotor flux linkage ψ_m is assumed to be known.

The following summarizes the most important conclusions of the simulations in chapter 6.

- It is shown that for the commission offline parameter identification case, both algorithms converge to the same parameter values, but the RLS is about 6 times faster and the NPA gives a large overshoot or undershoot when estimating R_s . The magnitude mean error of the

estimated parameter L_d is lower than 5% and for L_q lower than 4%. When it comes to R_s the magnitude mean error is lower than 0.7%.

- Both algorithms are capable of tracking time varying PMSM parameters, but the RLS tracks the parameter variations much better compared to the NPA and especially when it comes to R_s .
- The NPA gives noisier parameter estimations in general.
- Both algorithms are robust to angular error in ψ_m lower then 5%.

The following summarizes the most important conclusions from chapter 7, Implementation.

- Computational time for both methods has been estimated from the number of operations that are required and it is shown that the NPA is about 3 times faster compared to the RLS. The fastest DSP today at Aros electronics AB can approximately execute the NPA in $4\mu s$ and the RLS in $11\mu s$. It is important to clear out that this is a rough approximation.
- Experimental verification from real motor data has shown that the algorithms do not convergence to the same parameter values. The NPA estimates the parameter higher than the RLS and especially L_q .
- The stator resistance and the stator inductances have been measured directly on the motor windings and compared with the identified parameter values from the algorithms. It is shown that they deviate more than expected. Analysis of the obtained estimates indicates that the parameter values could have been fitted to some disturbance in the measured motor currents and/or that the applied voltages to the motor deviate from the reference voltages used as input to the algorithms.

8.2 Future Work

This thesis has considered parameter identification algorithms for electrical PMSM parameters. As always, much more aspects could be considered. The following gives some proposed future work to the thesis.

- It is a time delay from the reference voltage to the voltage applied to the motor. This time delay is not considered in this thesis. It would be interesting to investigate the impact of the time delay on the parameter identification algorithms.
- More investigations could be done to verify that the recorded motor data and the measured currents are correct.
- When the algorithms work properly on the recorded data the next step would be to implement one or both algorithms in real time.
- By identifying parameter variation in real time it is possible to detect drive faults. A lot of research has been done in this area and it would be interesting to investigate how well the algorithms work for drive fault diagnosis.

Bibliography

- [1] H. Toliyat, E. Levi, and M. Raina, “A review of rfo induction motor parameter estimation techniques,” *Energy Conversion, IEEE Transactions on*, vol. 18, no. 2, pp. 271 – 283, june 2003.
- [2] Q. An and L. Sun, “On-line parameter identification for vector controlled pmsm drives using adaptive algorithm,” in *Vehicle Power and Propulsion Conference, 2008. VPPC '08. IEEE*, sept. 2008, pp. 1 –6.
- [3] B. Nahid-Mobarakeh, F. Meibody-Tabar, and F.-M. Sargos, “Mechanical sensorless control of pmsm with online estimation of stator resistance,” *Industry Applications, IEEE Transactions on*, vol. 40, no. 2, pp. 457 – 471, march-april 2004.
- [4] B. Nahid Mobarakeh, F. Meibody-Tabar, and F. Sargos, “On-line identification of pmsm electrical parameters based on decoupling control,” in *Industry Applications Conference, 2001. Thirty-Sixth IAS Annual Meeting. Conference Record of the 2001 IEEE*, vol. 1, sep-4 oct 2001, pp. 266 –273 vol.1.
- [5] T. Boileau, B. Nahid-Mobarakeh, and F. Meibody-Tabar, “On-line identification of pmsm parameters: Model-reference vs ekf,” in *Industry Applications Society Annual Meeting, 2008. IAS '08. IEEE*, oct. 2008, pp. 1 –8.

- [6] T. Boileau, N. Leboeuf, B. Nahid-Mobarakeh, and F. Meibody-Tabar, “Online identification of pmsm parameters: Parameter identifiability and estimator comparative study,” *Industry Applications, IEEE Transactions on*, vol. 47, no. 4, pp. 1944–1957, july-aug. 2011.
- [7] M. Elbuluk, L. Tong, and I. Husain, “Neural-network-based model reference adaptive systems for high-performance motor drives and motion controls,” *Industry Applications, IEEE Transactions on*, vol. 38, no. 3, pp. 879–886, may/jun 2002.
- [8] W. Shaowei and W. Shanming, “Identify pmsm’s parameters by single-layer neural networks with gradient descent,” in *Electrical and Control Engineering (ICECE), 2010 International Conference on*, june 2010, pp. 3811–3814.
- [9] L. Liu, D. Cartes, and W. Liu, “Particle swarm optimization based parameter identification applied to pmsm,” in *American Control Conference, 2007. ACC ’07*, july 2007, pp. 2955–2960.
- [10] Q. Miao-wang, T. Guo-jun, and L. Zang, “Parameter identification of pmsm based on flpso algorithm,” in *Information Engineering and Computer Science (ICIECS), 2010 2nd International Conference on*, dec. 2010, pp. 1–4.
- [11] K. J. Åström and B. Wittermark, *Adaptive Control*, 2nd ed. Dover Publicationsl, 2008.
- [12] A. Iwata, S. Ichikawa, M. Tomita, S. Doki, and S. Okuma, “Position and velocity sensorless control of synrms using on-line parameter identification,” in *Industrial Electronics Society, 2003. IECON ’03. The 29th Annual Conference of the IEEE*, vol. 3, nov. 2003, pp. 2156–2161 Vol.3.

- [13] S. Ichikawa, M. Tomita, S. Doki, and S. Okuma, “Low speed sensorless control and initial position estimation of synchronous reluctance motors based on system identification theory,” in *Industrial Electronics, 2004 IEEE International Symposium on*, vol. 2, may 2004, pp. 1197 – 1202 vol. 2.
- [14] —, “Sensorless control of synchronous reluctance motors based on extended emf models considering magnetic saturation with online parameter identification,” *Industry Applications, IEEE Transactions on*, vol. 42, no. 5, pp. 1264 –1274, sept.-oct. 2006.
- [15] N. P. Quang and J.-A. Dittrich., *Vector Control of Three-Phase AC Machines: System Development in the Practice (Power Systems)*. Springer, 2008.
- [16] L. Harnefors, *Control of Variable-Speed Drives*. Västerås, Sweden: Department of Electronics, Mälardalen University, 2002.
- [17] W. Leonard, *Control of Electrical Drives*. Springer, 1996.
- [18] J.-I. Itoh, N. Nomura, and H. Ohsawa, “A comparison between v/f control and position-sensorless vector control for the permanent magnet synchronous motor,” in *Power Conversion Conference, 2002. PCC Osaka 2002. Proceedings of the*, vol. 3, 2002, pp. 1310 –1315 vol.3.
- [19] K. j. Åström and B. Wittermark, *Computer Controlled Systems: Theory and Design*, 3rd ed. Prentice Hall, 1997.
- [20] L. Ljung, *System Identification-Theory for the User*, 2nd ed. Prentice Hall, 1999.

