View metadata, citation and similar papers at core.ac.uk

brought to you by CORE

provided by VBN

Aalborg Universitet

AALBORG UNIVERSITY
DENMARK

Discrete-Time LPV Current Control of an Induction Motor

Bendtsen, Jan Dimon; Trangbæk, Klaus

Publication date:
2003

Document Version
Også kaldet Forlagets PDF

Link to publication from Aalborg University

Citation for published version (APA):
Bendtsen, J. D., & Trangbæk, K. (2003). Discrete-Time LPV Current Control of an Induction Motor.

General rights
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

? Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
? You may not further distribute the material or use it for any profit-making activity or commercial gain
? You may freely distribute the URL identifying the publication in the public portal ?

Take down policy
If you believe that this document breaches copyright please contact us at vbn@aub.aau.dk providing details, and we will remove access to the work immediately and investigate your claim.

Downloaded from vbn.aau.dk on: April 25, 2017

Discrete-Time LPV Current Control of an Induction Motor

Jan Dimon Bendtsen Klaus Trangbæk
Department of Control Engineering, Aalborg University,
Fredrik Bajers Vej 7C, DK-9220 Aalborg East, Denmark.
Email: {dimon,ktr}@control.auc.dk

Abstract

In this paper we apply a new method for gain-scheduled output feedback control of nonlinear systems to current control of an induction motor. The method relies on recently developed controller synthesis results for linear parameter-varying (LPV) systems, where the controller synthesis is formulated as a set of linear matrix inequalities with full-block multipliers. A standard nonlinear model of the motor is constructed and written on LPV form. We then show that, although originally developed in continuous time, the controller synthesis results can be applied to a discrete-time model as well without further complications. The synthesis method is applied to the model, yielding an LPV discrete-time controller. Finally, the efficiency of the control scheme is validated via simulations as well as experimentally on the actual induction motor, both in open-loop current control and when an outer speed control loop is closed around the current loop.

1 Introduction

Gain scheduling is a well-known and common approach to control of well-behaved nonlinear systems. The classical approach to gain-scheduling control has been to linearise the plant model in some set of operating points and design one or more linear, possibly robust, controllers for the system in said points. The gains of these individual linear controllers are then interpolated between the different operating points. This approach has been used in a multitude of applications and often works well as long as the scheduling variable, i.e., the variable according to which the controllers are interpolated, varies slowly. However, as pointed out in [13], the rate of change of the parameter variation imposes fundamental limitations on the achievable performance of classical gain scheduling controllers. Also, the classical gain scheduling methods are generally somewhat ad hoc.

More recent work on *linear parameter varying* (LPV) control has addressed these issues by devising rigorous methods in which it is possible to compensate for known, fast parameter variations directly in the control design [9, 11, 12]. Linear parameter-varying systems are linear systems whose system matrices depend on some time-varying parameter vector that is either fully known or at least known to be contained in some known set. In LPV control design this knowledge is employed to provide systematic gain scheduling in which stability and performance of the closed loop can be guaranteed. The controller synthesis is cast as a set of matrix inequalities based on the varying system matrices and the plant-controller interconnection, along with a set of multipliers, which must satisfy these matrix inequalities.

One problem with these types of approaches has so far been that it can be difficult to obtain non-conservative controllers for a given plant if the plant parameter variations are considerable and restrictions are placed on the controller synthesis in the form of pre-imposed structures in the aforementioned multipliers. In [2] a controller synthesis with structural constraints on the multipliers was achieved for parameter dependencies entering the system via a *linear fractional transformation* (LFT) description. The structural constraints were dealt with in case of affine parameter dependencies in [3], but it is only recently that it has been shown how they can be lifted in case of more general, rational parameter dependencies, i.e., in LFT descriptions, as well. The resulting synthesis matrix inequalities yielding the controllers can be solved by using the so-called *full-block S-procedure* [12]. In essence, this results in an automated controller design method for nonlinear systems which permit an LPV description. To the best of our knowledge, this paper presents the first actual implementation of an LPV controller designed via the full-block S-procedure.

In this paper we will use this novel technique to design and test a rotor flux oriented current controller for an induction motor. Several continuous-time control schemes that take the induction motor dynamics into account have been applied to this problem [6, 7, 10, 14, 15]. A drawback of most of these methods is, however, that they require a considerable amount of tuning and engineering insight. In this paper we will demonstrate that LPV controller synthesis can be applied to the problem and achieve satisfying perfor-

mance basically without any ad hoc tuning. Another general problem with these schemes is that it is unclear whether or not they will work well when implemented in discrete time at a sampling frequency which is not considerably faster than the motor dynamics. This is generally important in connection with practical implementations if there are limitations on the sample rate compared to the dynamics of the plant, since the accuracy of a continuous-time nonlinear design may not be sufficient if the sample rate cannot be chosen high enough. If the sample rate cannot be chosen freely, a continuous-time controller synthesis may result in discrete-time controllers with very high gains or unstable open-loop dynamics, which could result in the designed closed-loop behaviour not being preserved when the controller is implemented. In this paper we show that the LPV controller synthesis can be carried out in discrete time and applied to the current control problem. A few other examples of discrete-time designs for induction motors can be found in [4] and [17].

The content of the paper is as follows. Section 2 briefly discusses the model of the induction motor on which the LPV control law is based. It is written as a complex-valued state space model which can easily be discretised. Next, in Section 3 we discuss the discrete-time formulation of the controller synthesis problem based on full-block multipliers and show that the problem can be solved with only small modifications compared to the continuous-time version. In Section 4 the control synthesis result presented in Section 3 is applied to the induction motor model of Section 2. Section 5 then demonstrates a few simulation results where it is seen that the controller performs as expected. Section 6 shows practical experiments on an induction motor. Finally, Section 7 sums up the conclusions of the work.

2 LPV Description of Motor Model

The induction motor setup we are considering in this paper is shown schematically in Figure 1. An inverter feeds three-phase alternating current (i_{sA}, i_{sB} and i_{sC}) to the motor based on the PWM voltages u_{sA}, u_{sB} and u_{sC}. The three-phase voltages and currents are transformed from/into a single complex voltage and current representation in a rotating reference frame, respectively, according to the relations

$$\begin{aligned} u_s &= u_{sd} + ju_{sq} = \frac{2}{3}\left(u_{sA} + u_{sB}e^{j\frac{2\pi}{3}} + u_{sC}e^{j\frac{4\pi}{3}}\right)e^{-j\rho} \\ i_s &= i_{sd} + ji_{sq} = \frac{2}{3}\left(i_{sA} + i_{sB}e^{j\frac{2\pi}{3}} + i_{sC}e^{j\frac{4\pi}{3}}\right)e^{-j\rho} \end{aligned}$$

where ρ is the angular position of the chosen reference frame. $u_{sd} = \Re\{u_s\}, u_{sq} = \Im\{u_s\}, i_{sd} = \Re\{i_s\}$, and $i_{sq} = \Im\{i_s\}$ are all real-valued signals. The aim we will pursue in this paper is to design an inner current control loop which can be placed in a cascade coupling with an outer shaft speed control loop, as indicated in Figure 1.

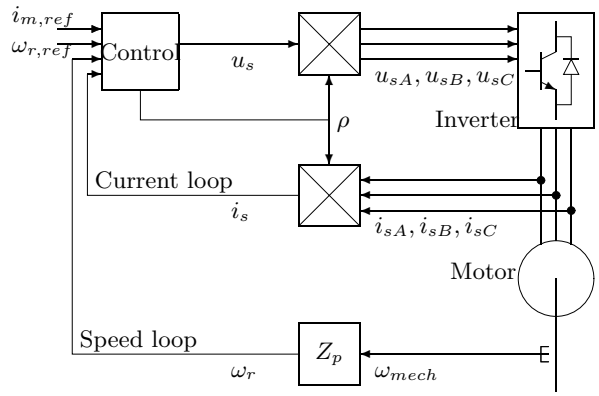

Figure 1: *Schematic drawing of the induction motor setup.*

With standard assumptions, the motor model is given by (see e.g. [8] or [10]):

$$\begin{aligned} \frac{di_s}{dt} &= -\left(\frac{R_s + R'_r}{L'_s} + j\omega\right)i_s \\ &\quad + \left(\frac{R'_r}{L'_s} - j\frac{L'_m}{L'_s}Z_p\omega_{mech}\right)i_m + \frac{u_s}{L'_s} \qquad (1) \\ \frac{di_m}{dt} &= \frac{R'_r}{L'_m}i_s - \left(\frac{R'_r}{L'_m} + j(\omega - Z_p\omega_{mech})\right)i_m \qquad (2) \end{aligned}$$

in which u_s and i_s are the complex stator voltage and current, respectively, and i_m is the complex magnetising current associated with the rotor flux $\Psi_r = L_m i_m$. The equations above are given in a reference frame which rotates with a rotational speed $\omega = \dot{\rho}$. L_m is the magnetising inductance. $R'_r = (L_m/L_r)^2 R_r, L'_s = L_s - L_m^2/L_r$ and $L'_m = L_m^2/L_r$ are the referred parameters used in the model, found based on identified values of the stator and rotor resistances and inductances R_s, R_r, L_s and L_r. Z_p is the number of pole pairs, while ω_{mech} is the motor shaft speed. The motor develops the electromagnetic torque

$$m_e = \Im\{\frac{3}{2}Z_p L'_m i_s i_m^*\}$$

while the load torque m_L acts as a disturbance via the mechanical relation

$$J\frac{d\omega_{mech}}{dt} = m_e - m_L$$

where J is the mechanical moment of inertia. We choose a reference frame rotating at the same angle as the magnetising current, since in this frame the steady-state signals are constant. Since, in reality, the magnetising current cannot be measured, we will use the following simple estimator. Let $T_r = L'_m/R'_r$ and $\omega_r = Z_p\omega_{mech}$ and compute the estimate of i_m, $\hat{i}_m$,

based on (2) as

$$\omega = \omega_r + i_{sq}/(T_r \hat{i}_{md}) \quad (3)$$

$$\frac{d\hat{i}_m}{dt} = \frac{1}{T_r} i_s - (\frac{1}{T_r} + j(\omega - \omega_r))\hat{i}_m. \quad (4)$$

In this reference frame, $\hat{i}_m$ is real. We choose the complex state vector $x = \begin{bmatrix} i_s^* & i_m^* \end{bmatrix}^*$ and insert (3) in (1)–(2) obtaining

$$\dot{x} = (A_0 + \delta_1 A_1 + \delta_2 A_2)\, x + B u_s, \quad i_s = Cx \quad (5)$$

in which

$$A_0 = \begin{bmatrix} -\frac{R_s + R_r'}{L_s'} & \frac{R_r'}{L_s'} \\ \frac{R_r'}{L_m'} & -\frac{R_r'}{L_m'} \end{bmatrix}, \; B = \begin{bmatrix} \frac{1}{L_s'} \\ 0 \end{bmatrix} \text{ and } C = \begin{bmatrix} 1 & 0 \end{bmatrix}$$

represent the nominal model, which is an LTI system, and

$$A_1 = \begin{bmatrix} -j & -j\frac{L_m'}{L_s'} \\ 0 & 0 \end{bmatrix} \text{ and } A_2 = \begin{bmatrix} 0 & j\frac{L_m'}{T_r L_s'} \\ 0 & -j\frac{1}{T_r} \end{bmatrix}$$

represent the effects of parameter variations in the linear system. These parameter variations symbolise the nonlinearities caused by $\delta_1 = \omega$ and $\delta_2 = i_{sq}/\hat{i}_{md}$, where all signals vary with time and $\hat{i}_{md} > 0 \; \forall t$. This particular choice of parameterisation has the advantage over the other obvious choice, $\delta_1 = \omega$ and $\delta_2 = \omega_r$, that ω will typically be close to ω_r; the parameterisation chosen above is a straightforward way to exploit this knowledge. The system (5) can then be written on an LFT form and can be meaningfully discretised, for instance according to the method presented in [1]. By considering the parameter variations as being caused by external effects, we are able to employ the LPV control synthesis that will be described in the following section. It should be noted, as already mentioned in the introduction, that the main reason why we discretise the model at this point is to be able to address limitations on the sample rate already in the synthesis phase, before the actual implementation.

3 LPV Controller Synthesis

In the synthesis, we consider the discrete-time system $M(k)$:

$$\begin{bmatrix} x_{k+1} \\ \hline z_{u,k} \\ z_{p,k} \\ y_k \end{bmatrix} = \left[\begin{array}{c|ccc} A & B_u & B_p & B \\ \hline C_u & D_{uu} & D_{up} & E_u \\ C_p & D_{pu} & D_{pp} & E_u \\ C & F_u & F_p & 0 \end{array}\right] \begin{bmatrix} x_k \\ \hline w_{u,k} \\ w_{p,k} \\ u_k \end{bmatrix} \quad (6)$$

with $x_k \in \mathbb{C}^n, u_k \in \mathbb{C}^m$ and $y_k \in \mathbb{C}^p$ representing states, inputs and outputs at sample instant k, respectively. All the matrices are assumed to be complex, constant and of appropriate dimensions. $w_{p,k} \in \mathbb{C}^{p_w}$ and $z_{p,k} \in \mathbb{C}^{p_z}$ are used to specify performance and $w_{u,k} \in \mathbb{C}^{u_w}$ and $z_{u,k} \in \mathbb{C}^{u_z}$ are channels which connect a set of residual gains collected in the mapping Δ_k with the nominal linear system as follows:

$$w_{u,k} = \Delta_k z_{u,k}. \quad (7)$$

Δ_k is a time-varying mapping that represents the nonlinearities in the system. We will assume that $\Delta \in \mathbf{\Delta}$, where $\mathbf{\Delta}$ is a compact, path-connected set containing 0, and that the interconnection between the nominal system model $M(k)$ and Δ is *well-posed*, that is, $I - \Delta D_{uu}$ is nonsingular for all $\Delta \in \mathbf{\Delta}$.

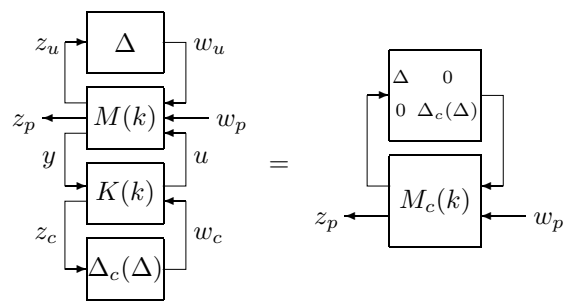

Figure 2: *The interconnection of the nominal system $M(k)$, the residual gains Δ, and the controller $K(k)$.*

We then consider the controller-system interconnection depicted in Figure 2. The controller is of the form

$$\begin{bmatrix} x_{c,k+1} \\ \hline u_k \\ z_{c,k} \end{bmatrix} = \left[\begin{array}{c|cc} A_c & B_{c1} & B_{c2} \\ \hline C_{c1} & D_{c11} & D_{c12} \\ C_{c2} & D_{c21} & D_{c22} \end{array}\right] \begin{bmatrix} x_{c,k} \\ \hline y_k \\ w_{c,k} \end{bmatrix} \quad (8)$$

with $w_{c,k} = \Delta_c(\Delta_k) z_{c,k}$ where Δ_c is a nonlinear function of Δ. If we interconnect the controller and the nominal system as depicted in the left part of Figure 2 we get the closed-loop LTI system $M_c(k)$ described by

$$\begin{bmatrix} \chi_{k+1} \\ \hline z_{u,k} \\ z_{c,k} \\ z_{p,k} \end{bmatrix} = \left[\begin{array}{c|ccc} \mathcal{A} & \mathcal{B}_u & \mathcal{B}_c & \mathcal{B}_p \\ \hline \mathcal{C}_u & \mathcal{D}_{uu} & \mathcal{D}_{uc} & \mathcal{D}_{up} \\ \mathcal{C}_c & \mathcal{D}_{cu} & \mathcal{D}_{cc} & \mathcal{D}_{cp} \\ \mathcal{C}_p & \mathcal{D}_{pu} & \mathcal{D}_{pc} & \mathcal{D}_{pp} \end{array}\right] \begin{bmatrix} \chi_k \\ \hline w_{u,k} \\ w_{c,k} \\ w_{p,k} \end{bmatrix} \quad (9)$$

subject to the parameter dependency

$$\begin{bmatrix} w_u \\ w_c \end{bmatrix} = \begin{bmatrix} \Delta_k & 0 \\ 0 & \Delta_c(\Delta_k) \end{bmatrix} \begin{bmatrix} z_u \\ z_c \end{bmatrix} \quad (10)$$

and with the state vector $\chi_k = \begin{bmatrix} x_k^* & x_{c,k}^* \end{bmatrix}^*$. Δ_c and the controller matrices must be chosen such that the interconnection with the system and controller is well-posed, i.e., $I - \left[\begin{smallmatrix} \Delta & 0 \\ 0 & \Delta_c \end{smallmatrix}\right] \left[\begin{smallmatrix} \mathcal{D}_{uu} & \mathcal{D}_{uc} \\ \mathcal{D}_{cu} & \mathcal{D}_{cc} \end{smallmatrix}\right]$ is nonsingular for all $\Delta \in \mathbf{\Delta}$. More explicitly, the gains of $M_c(k)$ in (9) are

given by

$$
\begin{aligned}
M_c &= \left[\begin{array}{c|cc|c} \mathcal{A} & \mathcal{B}_u & \mathcal{B}_c & \mathcal{B}_p \\ \hline \mathcal{C}_u & \mathcal{D}_{uu} & \mathcal{D}_{uc} & \mathcal{D}_{up} \\ \mathcal{C}_c & \mathcal{D}_{cu} & \mathcal{D}_{cc} & \mathcal{D}_{cp} \\ \hline \mathcal{C}_p & \mathcal{D}_{pu} & \mathcal{D}_{pc} & \mathcal{D}_{pp} \end{array}\right] \\
&= \left[\begin{array}{cc|cc|c} A & 0 & B_u & 0 & B_p \\ 0 & 0 & 0 & 0 & 0 \\ \hline C_u & 0 & D_{uu} & 0 & D_{up} \\ 0 & 0 & 0 & 0 & 0 \\ \hline C_p & 0 & D_{pu} & 0 & D_{pp} \end{array}\right] \\
&\quad + \left[\begin{array}{cc|c} 0 & B & 0 \\ I & 0 & 0 \\ \hline 0 & E_u & 0 \\ 0 & 0 & I \\ \hline 0 & E_p & 0 \end{array}\right] K \left[\begin{array}{cc|cc|c} 0 & I & 0 & 0 & 0 \\ C & 0 & F_u & 0 & F_p \\ 0 & 0 & 0 & I & 0 \end{array}\right] \\
&= \left[\begin{array}{c|c|c} \mathcal{A}_m & \mathcal{B}_{m1} & \mathcal{B}_{m2} \\ \hline \mathcal{C}_{m1} & \mathcal{D}_{m11} & \mathcal{D}_{m12} \\ \hline \mathcal{C}_{m2} & \mathcal{D}_{m21} & \mathcal{D}_{m22} \end{array}\right]
\end{aligned}
$$

where K is the matrix of controller gains given by (8). It can be shown [12] that the trajectories of (9) are identical to those of the nonlinear system

$$
\begin{aligned}
\chi_{k+1} &= \bar{\mathcal{A}}(\Delta_k)\chi_k + \bar{\mathcal{B}}(\Delta_k)w_{p,k} \\
z_{p,k} &= \bar{\mathcal{C}}(\Delta_k)\chi_k + \bar{\mathcal{D}}(\Delta_k)w_{p,k}
\end{aligned} \tag{11}
$$

where

$$
\begin{bmatrix} \bar{\mathcal{A}}(\Delta_k) & \bar{\mathcal{B}}(\Delta_k) \\ \bar{\mathcal{C}}(\Delta_k) & \bar{\mathcal{D}}(\Delta_k) \end{bmatrix} = \begin{bmatrix} \mathcal{A}_m & \mathcal{B}_{m2} \\ \mathcal{C}_{m2} & \mathcal{D}_{m22} \end{bmatrix} +
$$
$$
\begin{bmatrix} \mathcal{B}_{m1} \\ \mathcal{D}_{m21} \end{bmatrix} \left[\begin{smallmatrix} \mathcal{D}_{uu} & \mathcal{D}_{uc} \\ \mathcal{D}_{cu} & \mathcal{D}_{cc} \end{smallmatrix}\right] \left(I - \left[\begin{smallmatrix} \Delta & 0 \\ 0 & \Delta_c \end{smallmatrix}\right] \left[\begin{smallmatrix} \mathcal{D}_{uu} & \mathcal{D}_{uc} \\ \mathcal{D}_{cu} & \mathcal{D}_{cc} \end{smallmatrix}\right]\right)^{-1} \begin{bmatrix} \mathcal{C}_{m1} & \mathcal{D}_{m12} \end{bmatrix}.
$$

The objective is, if possible, to find a gain-scheduled control law $K(k)$ and a scheduling function $\Delta_c(\Delta)$ such that the closed loop system (9) fulfills a *robust quadratic performance specification* (RQP), which is defined as follows.

- The interconnection of system and controller is well-posed.
- The unforced system is uniformly asymptotically stable, i.e., positive constants K and α exist such that $\|\chi_k\| \leq \|\chi_0\| K e^{-\alpha k}$ for $k \geq 0$ and all $\Delta \in \mathbf{\Delta}$ if $w_{p,k} \equiv 0$.
- The following performance specification holds for $\chi_0 = 0$:

$$
\exists \varepsilon > 0 : \sum_{k=0}^{\infty} \begin{bmatrix} w_{p,k} \\ z_{p,k} \end{bmatrix}^* P_p \begin{bmatrix} w_{p,k} \\ z_{p,k} \end{bmatrix} \leq -\varepsilon \sum_{k=0}^{\infty} w_{p,k}^* w_{p,k} \tag{12}
$$

  for some $P_p = \left[\begin{smallmatrix} Q_p & S_p \\ S_p^* & R_p \end{smallmatrix}\right], R_p \geq 0$, specified a priori.

As can be seen, this formulation is equivalent to the continuous-time formulation of the notion of RQP (see for instance [12]). The following result shows that the discrete-time version of the full-block S-procedure yields a synthesis procedure that will guarantee (discrete-time) RQP for (9).

Theorem 1 *Robust quadratic performance is achieved for the system (9)–(10) if one of the following two equivalent properties holds.*

1. *(9)–(10) is well-posed and there exists a Hermitian $\mathcal{X} > 0$ such that*

$$
\begin{bmatrix} * \\ * \\ \hline * \\ * \end{bmatrix}^* \left[\begin{array}{cc|cc} -\mathcal{X} & 0 & 0 & 0 \\ 0 & \mathcal{X} & 0 & 0 \\ \hline 0 & 0 & Q_p & S_p \\ 0 & 0 & S_p^* & R_p \end{array}\right] \left[\begin{array}{cc} I & 0 \\ \bar{\mathcal{A}}(\Delta) & \bar{\mathcal{B}}(\Delta) \\ \hline 0 & I \\ \bar{\mathcal{C}}(\Delta) & \bar{\mathcal{D}}(\Delta) \end{array}\right] < 0 \tag{13}
$$

   for all $\Delta \in \mathbf{\Delta}$.

2. *There exists a Hermitian multiplier*

$$
P_e = \left[\begin{array}{c|c} P & P_{eo} \\ \hline P_{eo}^* & P_{ed} \end{array}\right] \tag{14}
$$

   which fulfills the matrix inequality

$$
\left[\begin{array}{cc} \Delta & 0 \\ 0 & \Delta_c(\Delta) \\ \hline I & 0 \\ 0 & I \end{array}\right]^* P_e \left[\begin{array}{cc} \Delta & 0 \\ 0 & \Delta_c(\Delta) \\ \hline I & 0 \\ 0 & I \end{array}\right] > 0 \tag{15}
$$

   for all $\Delta \in \mathbf{\Delta}$ and a Lyapunov matrix $\mathcal{X} > 0$ such that

$$
\tau^* \left[\begin{array}{cc|c|c|cc} -\mathcal{X} & 0 & 0 & 0 & 0 & 0 \\ 0 & \mathcal{X} & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & P & P_{eo} & 0 & 0 \\ \hline 0 & 0 & P_{eo}^* & P_{ed} & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & Q_p & S_p \\ 0 & 0 & 0 & 0 & S_p^* & R_p \end{array}\right] \tau < 0 \tag{16}
$$

   where

$$
\tau = \left[\begin{array}{cccc} I & 0 & 0 & 0 \\ \mathcal{A} & \mathcal{B}_u & \mathcal{B}_c & \mathcal{B}_p \\ \hline 0 & I & 0 & 0 \\ \mathcal{C}_u & \mathcal{D}_{uu} & \mathcal{D}_{uc} & \mathcal{D}_{up} \\ \hline 0 & 0 & I & 0 \\ \mathcal{C}_c & \mathcal{D}_{cu} & \mathcal{D}_{cc} & \mathcal{D}_{cp} \\ \hline 0 & 0 & 0 & I \\ \mathcal{C}_p & \mathcal{D}_{pu} & \mathcal{D}_{pc} & \mathcal{D}_{pp} \end{array}\right]. \tag{17}
$$

Proof: Inspection reveals that the only difference between the continuous-time case and the discrete-time

case is the upper left block in the central factors in (16) and (13). The equivalence between 1) and 2) hence follows from a direct application of the full-block S-procedure, Theorem 8 in [12].

We thus just need to show that requirement 1) yields RQP. Let $w_{p,k} \equiv 0$ in (11) and choose $\mathcal{V}_k = \chi_k^* \mathcal{X} \chi_k$ as a Lyapunov candidate for the unforced system. The difference from sample to sample is $\mathcal{V}_{k+1} - \mathcal{V}_k = \chi_k^* \bar{\mathcal{A}}(\Delta)^* \mathcal{X} \bar{\mathcal{A}}(\Delta) \chi_k - \chi_k^* \mathcal{X} \chi_k$, which implies that the system is uniformly exponentially stable if $\bar{\mathcal{A}}(\Delta)^* \mathcal{X} \bar{\mathcal{A}}(\Delta) < \mathcal{X}$. But this is immediately deduced from the upper left block in (13), which can be written as $\bar{\mathcal{A}}(\Delta)^* \mathcal{X} \bar{\mathcal{A}}(\Delta) - \mathcal{X} + \bar{\mathcal{D}}(\Delta)^* R_p \bar{\mathcal{D}}(\Delta) < 0$. Since $R_p \geq 0$ it is seen that if $\mathcal{X}$ renders (13) satisfied, the unforced system is uniformly exponentially stable.

Furthermore, due to continuity and strictness of (13), we can add a small perturbation $G = \left[\begin{smallmatrix} 0 & 0 \\ 0 & \varepsilon I \end{smallmatrix}\right]$ to the left-hand side of the inequality without rendering it unsatisfied. Multiplying from the left and right with $\xi_k = \begin{bmatrix} \chi_k^* & w_{p,k}^* \end{bmatrix}^*$ then gives

$$\xi_k^* \begin{bmatrix} \bar{\mathcal{A}}(\Delta)^* \mathcal{X} \bar{\mathcal{A}}(\Delta) - \mathcal{X} & \bar{\mathcal{A}}(\Delta)^* \mathcal{X} \bar{\mathcal{B}}(\Delta) \\ \bar{\mathcal{B}}(\Delta)^* \mathcal{X} \bar{\mathcal{A}}(\Delta) & \bar{\mathcal{B}}(\Delta)^* \mathcal{X} \bar{\mathcal{B}}(\Delta) \end{bmatrix} \xi_k +$$
$$\xi_k^* \begin{bmatrix} 0 & I \\ \bar{\mathcal{C}}(\Delta) & \bar{\mathcal{D}}(\Delta) \end{bmatrix}^* P_p \begin{bmatrix} 0 & I \\ \bar{\mathcal{C}}(\Delta) & \bar{\mathcal{D}}(\Delta) \end{bmatrix} \xi_k + \xi_k^* G \xi_k \leq 0$$

which reduces to

$$(\chi_{k+1} - \chi_k)^* \mathcal{X} (\chi_{k+1} - \chi_k) + \begin{bmatrix} w_{p,k} \\ z_{p,k} \end{bmatrix}^* P_p \begin{bmatrix} w_{p,k} \\ z_{p,k} \end{bmatrix} + \varepsilon w_{p,k}^* w_{p,k} \leq 0$$

Summing from $k = 0$ to $k = \infty$ with $\chi_0 = 0$ and $\lim_{k \to \infty} \chi_k = 0$ then yields (12), and hence requirement 1) implies RQP. ■

It is observed that, as in the continuous-time case, (16) is a Linear Matrix Inequality [1] (LMI) in the unknowns $\mathcal{X}$ and P_e. The discrete-time controller synthesis thus continues completely analoguously with the continuous-time synthesis in [12]. It is not possible to describe the entire synthesis here, but, referring to [12], it is simply necessary perform the substitution

$$\begin{bmatrix} 0 & X \\ X & 0 \end{bmatrix} \rightarrow \begin{bmatrix} -X & 0 \\ 0 & X \end{bmatrix}$$
$$\begin{bmatrix} 0 & Y \\ Y & 0 \end{bmatrix} \rightarrow \begin{bmatrix} -Y & 0 \\ 0 & Y \end{bmatrix}$$

in Equations (25) and (26). In short the synthesis progresses as follows:

[1] LMIs can be solved efficiently using standard software tools. Refer to e.g. [5] for more information on LMIs in general.

The solution to an LMI problem provides P_e and $\mathcal{X}$. Since (17) is a linear function of the controller matrices (A_c, B_c, C_c, D_c), this means that (16) becomes a *quadratic matrix inequality* (QMI) in (A_c, B_c, C_c, D_c). A direct solution for the QMI problem (16) can for instance be found in [12] or [16]. The scheduling function will typically be on the form

$$\Delta_c(\Delta) = N_- V_-(\Delta)^* \times \left(\left[\begin{smallmatrix} \Delta \\ I \end{smallmatrix}\right]^* P \left[\begin{smallmatrix} \Delta \\ I \end{smallmatrix}\right] - V_-(\Delta) N_- V_-(\Delta)^* \right)^{-1} V_+(\Delta). \quad (18)$$

where V_- and V_+ are linear functions of Δ. To sum up, the synthesis consists of solving an LMI problem and then finding the controller matrices and scheduling function by direct solutions to algebraic problems.

Notice that we have allowed for complex signals and systems whereas [12] explicitly assumes real values. However the results can be applied to complex systems without problems [16].

4 Controller Synthesis

In this section we apply the synthesis method outlined in the previous section to the discrete-time motor model. The model (5) was employed, using the following parameter values previously identified from the actual motor setup:

$$A_0 = \begin{bmatrix} -320.7 & 140.0 \\ 10.5 & -10.5 \end{bmatrix}, \quad B = \begin{bmatrix} 42.0 \\ 0 \end{bmatrix}, \quad C = \begin{bmatrix} 1 & 0 \end{bmatrix}$$

and

$$A_1 = \begin{bmatrix} -j & -13.3j \\ 0 & 0 \end{bmatrix}, \quad A_2 = \begin{bmatrix} 0 & 140.0j \\ 0 & -10.5j \end{bmatrix}.$$

The contributions of the parameter variations to the state equation could be described by $B_u w_u = \left[\begin{smallmatrix} A_1 & A_2 \end{smallmatrix}\right] \left[\begin{smallmatrix} \delta_1 I \\ \delta_2 I \end{smallmatrix}\right] z_u$. However, since A_1 and A_2 both have rank 1, we may write

$$A_1 = U_1 \Sigma_1 V_1^* = \begin{bmatrix} u_1^1 & u_2^1 \end{bmatrix} \begin{bmatrix} \sigma_1 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} (v_1^1)^* \\ (v_2^1)^* \end{bmatrix}$$

and

$$A_2 = U_2 \Sigma_2 V_2^* = \begin{bmatrix} u_1^2 & u_2^2 \end{bmatrix} \begin{bmatrix} \sigma_2 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} (v_1^2)^* \\ (v_2^2)^* \end{bmatrix}$$

and let $B_u = \begin{bmatrix} u_1^1 \sigma_1 & u_1^2 \sigma_2 \end{bmatrix}$. It then follows that, with $z_u = C_u x, C_u = \begin{bmatrix} v_1^1 & v_1^2 \end{bmatrix}^*$, the parameter variation can be written as $\delta_1 A_1 x + \delta_2 A_2 x = B_u \left[\begin{smallmatrix} \delta_1 & 0 \\ 0 & \delta_2 \end{smallmatrix}\right] z_u$, which is advantageous in terms of implementation. The parameter variation channel $z_u \rightarrow w_u$ was hence defined as follows:

$$w_u = \Delta z_u = \begin{bmatrix} \omega & 0 \\ 0 & i_{sq}/\hat{i}_{md} \end{bmatrix} z_u.$$

The performance and noise/reference channels were denoted z_p and w_p. The performance output

$$z_p = \begin{bmatrix} i_s - i_{s,ref} \\ \sigma_u u_s \end{bmatrix}$$

consisted of the control error and the control signal weighted by a factor σ_u. The performance input (or noise channel)

$$w_p = \begin{bmatrix} i_{s,ref} \\ \nu_m \end{bmatrix}$$

consisted of the stator current reference and measurement noise. Finally, the measurement y was defined as the control error corrupted by random measurement noise $\nu_m \in [-\sigma_\nu; \sigma_\nu]$, i.e.

$$y = i_s - i_{s,ref} + \nu_m.$$

The following nominal system could thus be constructed:

$$\begin{bmatrix} \dot{x} \\ \hline z_u \\ z_p \\ y \end{bmatrix} = \left[\begin{array}{c|ccc} A_0 & B_u & 0 & B \\ \hline C_u & 0 & 0 & 0 \\ C_p & 0 & D_{pp} & E_p \\ C & 0 & F_p & 0 \end{array}\right] \begin{bmatrix} x \\ \hline w_u \\ w_p \\ u \end{bmatrix}.$$

The matrices C_p, D_{pp} and E_p were used to define the weightings of the state, noise, reference, and control signal contributions to the performance and measurement output, respectively. Correspondingly, F_p accounted for the weightings of the noise and reference contribution to the measurement output. We chose $\sigma_u = 10^{-6}$ and $\sigma_\nu = 10^{-8}$.

The control error part of the performance channel was augmented by a first-order filter that allowed frequency tuning of the controller; the pole was placed in $s = -100$. This system was then discretised with a sampling frequency of $600Hz$, using a bilinear transformation as described in [1], yielding the discrete-time nominal system (6). As discussed earlier, this sampling frequency was imposed by the hardware setup. The eigenvalues of the discretised system matrix were located at $z = 0.5815, z = 0.9903$ and $z = 0.9990$.

The next step was to solve the LMIs mentioned at the end of Section 3 in order to compute a controller. The performance specification

$$P_p = \begin{bmatrix} -\gamma I & 0 \\ 0 & \frac{1}{\gamma} I \end{bmatrix} \tag{19}$$

providing a bound γ on the induced 2-norm

$$\sup_{w_p \neq 0} \frac{\|z_p\|_2}{\|w_p\|_2} \leq \gamma$$

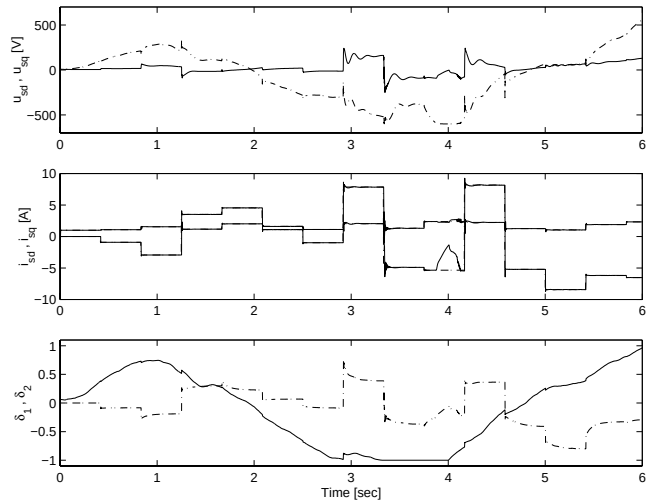

Figure 3: *LPV current control, simulation. The top figure shows the real and imaginary components, u_{sd} (full line) and u_{sq} (dash-dotted), of the control voltage generated by the controller. The middle figures show the real and imaginary components, i_{sd} and i_{sq}, of the controlled currents, plotted with full lines (—) along with their reference signals, plotted with dash-dotted lines (− · −). The bottom plot shows δ_1 (—) and δ_2 (− · −) scaled to the interval $[-1\ ;\ 1]$. As can be seen, the currents track the reference closely, except when the control voltage saturates (at around 4 sec).*

was chosen and a bisectional search for the smallest γ for which the LMIs were feasible could then be performed. We allowed $\delta_1 = \omega$ and $\delta_2 = \Im\{i_s\}/\Re\{\hat{i}_m\}$ to obtain values in the intervals $[-800\ ;\ 800]$ and $[-10\ ;\ 10]$, respectively. Under these circumstances, a performance of $\gamma = 0.0011$ could be achieved and a controller on the form (8) could be obtained by solving the QMI (16). When solving the synthesis QMI, it was found that the controller order could be reduced by one, yielding a second-order controller with eigenvalues of A_c in $z = 0.9990$ and $z = 0.8202$.

5 Simulations

In the simulations the reference sequence was chosen as a series of steps, each with a duration of 250 samples. For each step, the reference for i_{sq} was allowed to take random values in the interval $[-10\ ;\ 10]$, while the reference for i_{sd} was chosen from the interval $[1\ ;\ 3]$. The system was disturbed by a load torque m_L, which was a sequence of uniformly distributed white noise filtered through a first-order filter with a time constant of 1/2 second. Subject to these external signals, the nonlinear model generated the δ_1 and δ_2 sequences based on which the controller scheduling function was calcu-

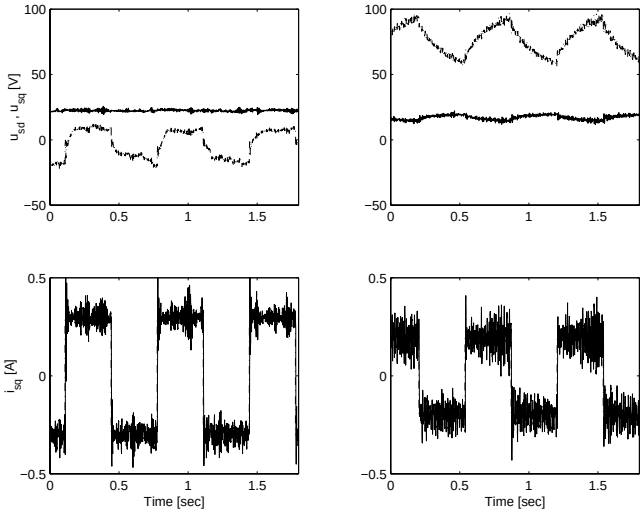

Figure 4: *LPV current control, experimental results. The top figures show the real and imaginary components, u_{sd} (full line) and u_{sq} (dash-dotted), of the control voltage generated by the implemented LPV controller. The lower figures show the controlled current i_{sq}. The reference signals are shown with dash-dotted lines ($- \cdot -$), while the measurements are shown with full lines (—). The left figures are without load, while the right figures are recorded with a load torque of $m_L = 4Nm$.*

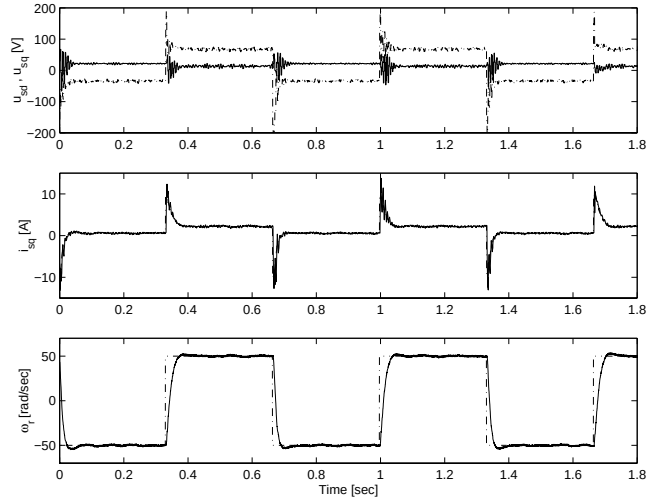

Figure 5: *LPV current control in cascade with rotational speed controller, experimental results. The top figure shows the real and imaginary components, u_{sd} and u_{sq}, of the control voltage generated by the implemented LPV controller. The middle and lower figures show the controlled current i_{sq} and the rotational speed ω_r, respectively. The reference signals are shown with dash-dotted lines ($- \cdot -$), while the measurements are shown with full lines (—). The current reference signal was generated by an outer loop speed controller.*

lated. Motivated by limitations of the hardware of the experimental setup, the control voltage u_s was made to saturate at $600V$.

Figure 3 shows a simulation of the current control system. It is seen that the control loop achieves good tracking, in accordance with the performance value achieved for all values of the parameter variations, except when the control signal saturates. The parameter variations are shown in the bottom plot in Figure 3, scaled to the interval $[-1\ ;\ 1]$. It is noted that the generated stator voltage compensates for the parameter variations throughout the allowed range. This scheduling is crucial to successful control; switching off the scheduling signal leads to instability in certain regions.

6 Practical Experiments

The controller presented above was implemented in C without modifications on a standard PC. The power device is a voltage-sourced inverter controlled directly from the PC. The induction motor is a $1.5kW$, two pole-pair motor with a rated torque of $10Nm$. The first two experiments were open-loop current control experiments, where the aim was to keep the magnetising current constant and make the imaginary part of the stator current follow a series of steps. The first experiment was conducted without load, while in the second experiment the motor shaft was subjected to a load torque of $4Nm$. The results are shown in Figure 4, where it is observed that the current tracks the reference steps adequately well. Looking at the stator voltage, it is noted that the imaginary part of the voltage is significantly different between the two experiments. This is due to the two different disturbance load torques, which cause the scheduling controller to yield significantly different control signals. Some variation can be noted in the real part of the voltage as well, caused by cross couplings between the stator and magnetising currents.

In the third experiment the shaft speed loop was closed using an outer PI-controller. In this case the stator current reference signals were thus generated by the PI-controller, and the LPV controller had to track these signals. The results of this experiment is shown in Figure 5. As can be seen, the control loop performs satisfactorily.

7 Concluding Remarks

In this paper a recently developed procedure for LPV controller synthesis, the so-called full-block S-

procedure, has been applied to stator current control of an induction motor, which is a highly nonlinear dynamical system. This method required the model to be written on a linear fractional form, where the nonlinearities entered the model as parameter variations. Due to hardware limitations on the sample rate, it was chosen to discretise the system model. It was demonstrated how the controller synthesis can be formulated in discrete time.

A controller was constructed such that its dynamics depended on a scheduling function calculated from the parameter variation part of the model. The synthesis of the controller and scheduling function was then achieved by solving a set of linear matrix inequalities constructed from the model parameters. It was found via simulations that the gain-scheduled closed-loop system fulfilled a robust quadratic performance specification throughout the operating range, but that it would become unstable if the scheduling was switched off. The gain-scheduling is thus an integral part of the controller. The main contribution of this paper was then to show that a systematic non-conservative control design with more than one scheduling parameter could be implemented on a real-life system with fast dynamics.

Finally, a suggestion for further research could be a deeper analysis of the robustness properties. The rotor resistance is known to vary greatly with temperature. Additionally, it is often desirable to avoid the use of a speed sensor. Both these subjects can be treated as robustness issues. Although a few simple tests indicate good robustness properties of the presented current controller, a more thorough analysis would be interesting. Alternatively, robustness to specified uncertainties can be incorporated in the synthesis, but this would be a non-convex problem, making the design much less straight-forward than the one suggested here.

Another interesting subject would be the design of an LPV control design for the entire control system from speed reference to stator voltage.

References

[1] P. Apkarian, "On the Discretization of LMI-synthesized Linear Parameter-Varying Controllers," *Automatica* Vol. 33, 4:655–661, 1997

[2] P. Apkarian, P. Gahinet, "A Convex Characterization of Gain-Scheduled H_∞ Controllers," *IEEE Transactions on Automatic Control*, Vol. 40, 5:853–864, 1995

[3] P. Apkarian, P. Gahinet, G. Becker, "Self-scheduled H_∞ control of linear parameter-varying systems: a design example" *Automatica*, Vol. 31, 9:1251–1262, 1995

[4] F. Blaabjerg, J. K. Pedersen, M. P. Kazmierkowski, "DSP-based Current Regulated PWM Inverter-fed Induction Motor Drive Without Speed Sensor," *Proc. of the IEEE International Symposium on Industrial Electronics*, 659–664, 1996

[5] S. Boyd, L. El Ghaoui, E. Feron, V. Balakrishnan, "Linear Matrix Inequalities in System and Control Theory," Philadelphia, PN: SIAM, 1994

[6] J. W. Choi, S. K. Sul, "Generalized solution of minimum time current control in three-phase balanced systems," *IEEE Transactions on Industrial Electronics* Vol. 45, 5:738–744, 1998

[7] J. Jung, S. Lim, K. Nam, "PI-type Decoupling Control Scheme for Highspeed Operation of Induction Motors," *Proc. of the 28th Annual IEEE Power Electronics Specialists Conference*, 1082–1085, 1997

[8] R. F. F. Koning, C. T. Chou, M. H. G. Verhaegen, J. B. Klaassens, J. R. Uittenbogaart, "A Novel Approach on Parameter Identification for Inverter Driven Induction Machines," *IEEE Transactions on Control Systems Technology* Vol. 8, 6:873–882, 2000

[9] A. Packard, "Gain Scheduling via Linear Fractional Transformations," *Systems and Control Letters* Vol. 22, 79–92, 1994

[10] H. Rasmussen, P. Vadstrup, H. Børsting, "Nonlinear Field Oriented Control of Induction Motors using the Backstepping Method," *Proc. of the Sixth International Conference on Control, Automation, Robotics and Vision*, 2000

[11] W. Rugh, J. S. Shamma, "Research on Gain Scheduling," *Automatica* Vol. 36, 1401–1425, 2000

[12] C. W. Scherer, "LPV Control and Full Block Multipliers," *Automatica* Vol. 37, 361–375 2001

[13] J. S. Shamma, M. Athans, "Gain Scheduling: Potential Hazards and Possible Remedies," *IEEE Control Systems Magazine* Vol. 12, 101–107, 1992

[14] L. G. Shiau, J. L. Lin, "Stability of Sliding-mode Current Control for High Performance Induction Motor Position Drives," *IEE Proceedings on Electric Power Applications* Vol. 148, 1:69–75, 2001

[15] K. K. Shyu, H. J. Shieh, "Variable Structure Current Control for Induction Motor Drives by Space Voltage Vector PWM," *IEEE Transactions on Control Systems Technology* Vol. 42, 6:572–578, 1995

[16] K. Trangbaek, "Linear Parameter Varying Control of Induction Motors," Ph.D. thesis, Aalborg University, 2001

[17] S. M. Yang, C. H. Lee, "A Current Control Technique for Voltage-fed Induction Motor Drives," *Proc. of the 25th Annual Conference IEEE Industrial Electronics Society*, 1380–1385, 1999