

Novel Lexicographic MPC for Loss Optimized Torque Control of Nonlinear PMSM

Christoph Schnurr¹, Sören Hohmann¹ and Johannes Kolb²

Abstract—In electrical drive applications torque tracking is conflicting with the subsidiary goal of loss minimization. In this Paper a lexicographic optimization is proposed to solve this conflict by stringent prioritization. A novel model predictive control (MPC) with a single objective function is presented which is proved to be equal to lexicographic optimization in steady state. The loss weighting factor does not affect the steady state tracking offset and no current set points are needed. Results are confirmed by a simulation with an experimental validated machine model. The torque of an anisotropic permanent magnet synchronous machine (PMSM) fed by a voltage source inverter (VSI) is controlled with minimal ohmic losses. The optimization under input constraints is done using the projected fast gradient method (PFGM).

I. INTRODUCTION

One key in designing a model predictive control (MPC) is the formulation of the objective function as a mathematical representation of the actual control objective. The scope of this paper is controlling the electromagnetic torque of a permanent magnet synchronous machine (PMSM) towards its reference value as fast as possible. Additionally, the steady state operation should be optimal in terms of power efficiency. The distinctiveness of this approach from the state of the art is the utilization of the lexicographic idea compared to a trade-off between tracking and loss minimization in the objective function allowing rigorous treatment of this multi-objective problem. We believe that a hierarchical approach is much closer to the desired control objective for the torque tracking. Straight forward application of the lexicographic optimization to the torque control results in two optimization problems. This is not desirable for real time implementations due to the small sampling times in drive control. It is shown, how to solve the lexicographic problem in the steady state in a single optimization step in order to ease the computation.

A. State of the Art

In general, MPC for electrical drives can be classified in continuous control set (CCS) MPC with a modulator and finite control set (FCS) MPC without a modulator. Newer results for applying FCS MPC to PMSM can be found in [1]–[7]. FCS MPC is used for applications where a constant current quality rather than a constant switching frequency is important [8]. An advantage of CCS MPC is the higher

switching frequency with the same sample time, due to the modulator. This is notably for computational intensive MPC approaches. That approach is presented in [9], where saturation effects are neglected and constraints simplified. For precise modeling of PMSM with high power density, the magnetic circuit cannot be described linear anymore. This would lead to inferior control performance compared to approaches that consider saturation effects. Also [10] utilizes CCS but does not inherently fulfill input constraints since it uses an explicit offline solution rather than online optimization.

The PFGM method [11] is shown in newer research [12], [13] and [14] to be capable for computations times in the lower and sub millisecond ranges. Also in drive application it has been already used. In [15] it is applied to solve the Model Predictive Pulse Pattern Control Problem online, where the switching times of the semiconductors (the pulse pattern) is optimized online. The usage for solving an MPC problem similar to this approach is presented in [8] and [16]. In the former one a current control of a AC/DC transmitter is designed. In the latter one the d-component of the current and the torque of a PMSM is controlled. However, the components of the flux vector in the dq-frame are used as states and no losses are considered.

As known from the basic machine theory, all steady states resulting the same torque lie on a hyperbola in the current plane. Thus, there is some degree of freedom in this control objective. The currents which cause the lowest losses are desired in steady state operation for torque tracking. Minimizing losses is the goal of various MPC approaches for PMSM control including e.g. [6], [17]–[19]. However, usually a trade-off is found by incorporating the two control objectives in the cost function as a weighted sum. This degenerates tracking performance since it slows down the tracking and also leads to a steady state error. To remedy that problem special methods for offset free tracking have to be used. State of the art of these are adding the disturbance dynamics to the prediction model and then use an observer to estimate the real disturbance [20], [21] or the delta input formulation [21], [22]. Approaches that addresses several control objectives are named multi-objective MPC. In [23] the objectives torque tracking and efficiency are combined for an induction machine using an optimization based weighting function generation algorithm. In [24] the multiple objectives for torque and flux tracking of an induction machine are solved with a ranking algorithm resulting in an equal trade-off for tracking of both variables.

¹Christoph Schnurr and Sören Hohmann are with the Karlsruhe Institute of Technology, Department of Electrical Engineering and Information Technology, Institute of Control Systems, 76131 Karlsruhe, Germany c.schnurr@kit.edu

²Johannes Kolb is with SHARE am KIT, Schaeffler Technologies AG & Co. KG, 76131 Karlsruhe, Germany johannes.kolb@schaeffler.com

B. Motivation

In MPC for electrical drives it is common to control the state vector to stationary optimized references (see e.g. [25] for MPC speed control). In this approach the torque tracking is formulated in the objective function. Therefore no current references are needed at all. This avoids the creation and usage of look-up tables for the mapping from the reference torque to the reference currents. From the state of the art we conclude, that the usage of PFGM as optimization algorithm for the MPC in electrical drives is a well suited. There is a lack of accuracy in the modelling of the machine and the constraints in many MPC approaches. The choice of the currents to be the states of the state space model is necessary here, since the ohmic losses and hence the currents are used in the objective function. However, this causes more effort in handling the nonlinear relation between flux and currents compared to using the flux components as states. That is because with the latter choice for the state vector a static nonlinear mapping can be used [8], [26]. The dissent between tracking and loss minimization in the objective function of MPC is not yet resolved satisfactorily because only trade-offs are found. It can be solved more elegantly by formulating an MPC with a lexicographic objective. This approach utilizes a stringent prioritization to answer this open question. The idea of using lexicographic optimization within MPC is traced back to [27] and [28]. To the authors best knowledge, lexicographic approaches have not yet been used for MPC in electrical drive torque control applications.

C. Contribution and Outline

A novel approach to solve a lexicographic optimization problem for steady states in a single objective function is presented. This is done by applying a transformation to the lower prioritized component of the objective function. The transformation is proved to be equal to the lexicographic standard approach in the unconstrained case, i.e., steady state operation. During transients, the presented approach reaches an optimum, that results from a weighted combination of the two control objectives. The new method is applied to design an MPC for torque control of a nonlinear PMSM. The MPC utilizes a PFGM optimization algorithm like [8]. However, it is now extended to loss consideration in the objective function as well as direct torque control in contrast to current control with look-up-tables. The approach is confirmed and further analyzed by simulation with an experimental validated machine model.

In section II the model of the PMSM which is the basis for controller synthesis is presented. Section III will show, how to solve the lexicographic problem for the steady state in a single optimization step. This method is then adapted for MPC and applied to the PMSM torque tracking problem in section IV. Simulations are shown in section V and important properties are highlighted. The paper closes with summarizing conclusions in section VI.

II. MODELING OF THE PMSM

Since the MPC utilizes a modulator, the complete modeling and control scheme is shown in the rotor oriented reference frame (dq-frame).

A. Machine Equations

As motivated before, the nonlinear magnetics have to be considered. Therefore, for the proposed controller, the functions describing the relation between the magnetic flux $\psi_{dq}(\mathbf{i}_{dq}(t))$ and the current $\mathbf{i}_{dq}(t) = [i_d(t) \ i_q(t)]^T$

$$\psi_{dq}(\mathbf{i}_{dq}(t)) = \begin{bmatrix} \psi_d(i_d(t), i_q(t)) \\ \psi_q(i_d(t), i_q(t)) \end{bmatrix} \quad (1a)$$

in the dq-frame has to be known. Typically (see e.g. [29]), they are considered in the form of look-up-tables depending on the currents in the dq-frame. From now on time dependencies of the current vector $\mathbf{i}_{dq}(t)$, the voltage vector $\mathbf{u}_{dq}(t) = [u_d(t) \ u_q(t)]^T$ and the following signals and their components are suppressed. Additionally, the arguments of the flux function $\psi_{dq}(\mathbf{i}_{dq}(t))$ and their elements are dropped for better readability. The System is described as per unit system. This leads to the voltage equations

$$u_d = r_s i_d + \dot{\psi}_d - \omega_{el} \psi_q, \quad (2a)$$

$$u_q = r_s i_q + \dot{\psi}_q + \omega_{el} \psi_d \quad (2b)$$

describing the dynamics of the PMSM with the normalized stator resistance r_s and the electrical angular velocity ω_{el} . It is assumed, that ω_{el} is constant. This is appropriate in many applications, where the mechanical time constant is much bigger than the electrical one. The number of pole pairs and the constant $2/3$ resulting from the amplitude invariant Park Transformation are included in the normalization constant. Hence, the torque m can be calculated as

$$m = \psi_d i_q - \psi_q i_d. \quad (3)$$

B. Affine State Space Model

Starting with the machine equations (2), the model is customized to allow MPC operation. To receive a convex optimization function an affine state space model is deduced.

1) *Linearization:* Equations (1) and (2) can be combined to a nonlinear system of differential equations. Using the currents as states $\mathbf{x} = [i_d \ i_q]^T \in \mathbb{R}^2$, voltages as inputs $\mathbf{u} = [u_d \ u_q]^T \in \mathbb{R}^2$ and the torque as output $m \in \mathbb{R}$ a nonlinear state space description depending on look-up-tables of the flux results. After applying the total differential, one gets an affine state space description for the d- and q-current. We assume that in the neighborhood of the current state, which is reachable within the prediction horizon of the MPC, it is sufficient to use a linearization of the system dynamics. This was qualitatively validated by simulations, since a receding linearization [30] with multiple operating points over the prediction horizon shows no major impact on model accuracy. Then a first-order Taylor approximation of the equations for $\dot{\mathbf{x}}$ can be done. For a fixed operating

point $\mathbf{z}^T = [\mathbf{x}^T \quad \mathbf{u}^T] \in \mathbb{R}^4$ this results in the affine system

$$\dot{\mathbf{x}} = \mathbf{A}_c(\mathbf{z}) \mathbf{x} + \mathbf{B}_c(\mathbf{z}) \mathbf{u} + \mathbf{G}_c(\mathbf{z}) \quad (4a)$$

$$m = \mathbf{C}_c(\mathbf{z}) \mathbf{x} + F_c(\mathbf{z}) \quad (4b)$$

with the matrices $\mathbf{A}_c(\mathbf{z}) \in \mathbb{R}^{2 \times 2}$, $\mathbf{B}_c(\mathbf{z}) \in \mathbb{R}^{2 \times 2}$, $\mathbf{G}_c(\mathbf{z}) \in \mathbb{R}^2$, $\mathbf{C}_c(\mathbf{z}) \in \mathbb{R}^{1 \times 2}$ and the scalar $F_c(\mathbf{z}) \in \mathbb{R}$. The equations for the matrix elements are rather complex but can be determined as explicit equations depending on the current operating point \mathbf{z} offline. The affine system model (4) is the basis for the discretization step.

2) *Discretization*: In accordance with [31] and [32] simulations show that a discretization with Euler Approximation is not reasonable for higher angular velocities ω_{el} . The system was discretized with a third order Picard Iteration [33]. This leads to the affine discrete state space system

$$\mathbf{x}_{k+1} = \mathbf{A}(\mathbf{z}_k) \mathbf{x}_k + \mathbf{B}(\mathbf{z}_k) \mathbf{u}_k + \mathbf{G}(\mathbf{z}_k) \quad (5a)$$

$$m_k = \mathbf{C}(\mathbf{z}_k) \mathbf{x}_k + F(\mathbf{z}_k). \quad (5b)$$

For every prediction step of the MPC this affine machine model is recalculated by employing the current operating point \mathbf{z}_k and the current angular velocity ω_{el} . From now on the operating point \mathbf{z}_k dependency of the matrices are left out for better readability as well as the index k representing the current time instance of the signals.

III. LEXICOGRAPHIC OPTIMIZATION FOR STEADY STATE OPERATION

Given are an objective vector

$$\mathbf{J}(\mathbf{x}) = [J_1(\mathbf{x}) \quad J_2(\mathbf{x}) \quad \dots \quad J_k(\mathbf{x})]^T \quad (6)$$

consisting of $k \in \mathbb{N}$ different scalar objective functions, a feasible set \mathcal{X} for the decision vector $\mathbf{x} \in \mathbb{R}^n$ and the image of the feasible set $\mathcal{J} = \mathbf{J}(\mathcal{X})$. From [34] follows:

Definition 1 (lexicographic optimum): An objective vector \mathbf{J}^* is a lexicographic optimum iff

$$\emptyset = \left\{ \mathbf{J} \in \mathcal{J} \mid J_i(\mathbf{x}) < J_i^*, i = \min_{j \in \mathcal{I}_k} \{j \mid J_j(\mathbf{x}) \neq J_j^*\} \right\},$$

where $J_j^* = \min_{\mathbf{x} \in \mathcal{X}} J_j(\mathbf{x})$ and $\mathcal{I}_k = \{1, \dots, k\}$.

Definition 2 (lexicographic minimizer): The lexicographic minimizer \mathbf{x}^* is defined as the argument of the objective function at the lexicographic optimum: $\mathbf{J}^* = \mathbf{J}(\mathbf{x}^*)$.

A straight forward reformulation of definition 1 leads to a hierarchical constrained optimization problem sequence [28]:

$$\begin{aligned} \mathbf{x}^* &\in \{ \mathbf{x} \in \mathcal{X} \mid J_i(\mathbf{x}) \leq \hat{J}_i^*, j \in \mathcal{I}_k \} \\ \hat{J}_i^* &= \min_{\mathbf{x} \in \mathcal{X}} \left(J_i(\mathbf{x}) \mid J_j(\mathbf{x}) \leq \hat{J}_j^*, j \in \mathcal{I}_{i-1} \right) \\ \hat{J}_1^* &= \min_{\mathbf{x} \in \mathcal{X}} (J_1(\mathbf{x})). \end{aligned} \quad (7)$$

Hence, the lexicographic minimizer for two objectives could be calculated by the solution of two consecutive constrained optimization problems. However, this is not practicable for real time implementations as they appear in drive control

applications. The novelty of this approach is a more efficient solution for this problem for the steady state operation, i.e., no state constraints $\mathbf{x} \in \mathcal{X}$ are active. The proposed scheme reformulates the optimization problem, such that it can be solved in single optimization step. Let now be

$$J_i(\mathbf{x}) = \mathbf{x}^T \mathbf{H}_i \mathbf{x} + \mathbf{h}_i^T \mathbf{x} \quad (8)$$

the form of the objective functions. Then the following facts can be stated [28]:

Remark 1: Assume that at least one \mathbf{H}_i is positive definite, then \mathbf{x}^* is unique.

Remark 2: However, all J_j with j greater than the smallest i with positive definite \mathbf{H}_i can be omitted, since they do not affect \mathbf{x}^* .

Remark 3: Every function of the type

$$J(\mathbf{x}) = \|\mathbf{c}(\mathbf{x})\|_2 \quad \text{with } \mathbf{c}(\mathbf{x}) = [c_1(\mathbf{x}) \dots c_m(\mathbf{x})]^T \quad (9)$$

where $\|\cdot\|_2$ denotes the 2-norm and $m < n$ with $\mathbf{x} \in \mathbb{R}^n$, can be reformulated in the form of (8) by neglecting the constant terms as usual in optimization. Since the result of a 2-norm is always non-negative the function is convex, but with $m < n$ Matrix \mathbf{H}_i will be not positive definite.

It is assumed, that only reachable set points are fed to the MPC. However, this is no severe restriction, since it is common to limit the values of the reference to the maximum rating in steady state. The following theorem gives a solution for the lexicographic minimizer for stationary operation in a single optimization step:

Theorem 1 (unconstrained lexicographic minimizer):

Given two convex functions: $J_1(\mathbf{x})$ in the form of (8) but deduced from (9) and $J_2(\mathbf{x})$ in the form of (8) with \mathbf{H}_2 positive definite, as well as a matrix \mathbf{R} satisfying

$$\mathbf{R}^{-1} = \mathbf{R}^T, \quad (10)$$

$$\mathbf{a}^T \mathbf{R} \mathbf{b} = 0 \quad (11)$$

for any linear dependent vectors $\mathbf{a}, \mathbf{b} \in \mathbb{R}^n$, the objective vector $\mathbf{J}(\mathbf{x}) = [J_1(\mathbf{x}) \quad J_2(\mathbf{x})]^T \in \mathbb{R}^2$ has the unconstrained lexicographic minimizer

$$\mathbf{x}^* = \arg \min_{\mathbf{x}} \left(J_1(\mathbf{x}) + \sum_{i=1}^m \left(\frac{\partial J_2(\mathbf{x})}{\partial (\mathbf{R} \nabla c_i(\mathbf{x}))} \right)^2 \right). \quad (12)$$

To illustrate the idea and to motivate the use of the rotational matrix \mathbf{R} in Fig. 1 the components of (12) are drawn for an example with $\mathbf{x} \in \mathbb{R}^2$ and $m = 1$. Fig. 1 also guides the proof of theorem 1, but first the following is introduced:

Definition 3: \mathcal{X}_y is called optimal set of objective function $J_y(\mathbf{x})$, iff

$$\mathcal{X}_y = \{ \mathbf{x} \mid J_y(\mathbf{x}) = \min_{\mathbf{x}} J_y(\mathbf{x}) \}.$$

Lemma 1: For the objective functions $J_a(\mathbf{x})$, $J_b(\mathbf{x})$ and $J_c(\mathbf{x}) = J_a(\mathbf{x}) + J_b(\mathbf{x})$ provided that $\mathcal{X}_a \cap \mathcal{X}_b \neq \emptyset$ it is obvious, that

$$\mathcal{X}_c = \mathcal{X}_a \cap \mathcal{X}_b.$$

Proof: [Sketch of proof of theorem 1] From the aforementioned assumption of only feasible reference values

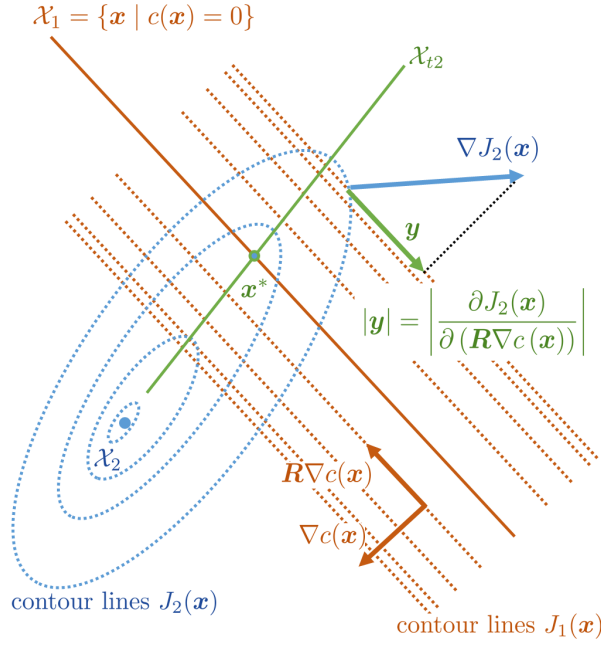


Fig. 1. Visualization of (12) for $x \in \mathbb{R}^2$ and $m = 1$

follows, that in steady state the solution of the optimization problem lies within the input constraint set area. Hence, the constraint $x \in \mathcal{X}$ in definition 1 can be dropped. The lexicographic minimizer of an objective vector $\mathbf{J}(\mathbf{x}) = [J_1(\mathbf{x}) \ J_2(\mathbf{x})]^T \in \mathbb{R}^2$ is given by

$$J_1^* = \min (J_1(\mathbf{x}))$$

$$\mathbf{x}^* = \arg \min_{\mathbf{x}} (J_2(\mathbf{x}) \mid J_1(\mathbf{x}) - J_1^* = 0).$$

Applying the Karush-Kuhn-Tucker conditions to the equality constraint $g(\mathbf{x}) = J_1(\mathbf{x}) - J_1^*$ results in the equivalent problem

$$\mathcal{X}_{t2} = \left\{ \mathbf{x} \mid \forall i \ \nabla J_2(\mathbf{x}) (\mathbf{R} \nabla c_i(\mathbf{x}))^T = 0 \right\} \quad (13)$$

$$\mathbf{x}^* = \{ \mathbf{x} \mid \mathbf{x} \in \mathcal{X}_1 \cap \mathcal{X}_{t2} \}. \quad (14)$$

By geometrically reasoning it follows from the assumptions on the definiteness of J_1 and J_2 , that \mathcal{X}_1 and \mathcal{X}_{t2} are linearly independent: The linear subspace of all points with a constant gradient direction of the strictly convex function J_2 can never be orthogonal to that gradient direction. This leads to $\mathcal{X}_1 \cap \mathcal{X}_{t2} \neq \emptyset$. From remark 1 follows, that lemma 1 can be applied to calculate the unique \mathbf{x}^* using theorem 1. ■

IV. CONTROLLER SYNTHESIS

The two control objectives are torque tracking and loss minimization and described by J_1 and J_2 respectively. They are prioritized in their numeric order. The controller will solve this lexicographic optimization problem for steady state operation. The control objective is evaluated over the prediction horizon in the cost function of the MPC. In the following section IV-A the cost function is examined solely for a single instant of time for ease of understanding. In

section IV-B the cost function is expanded over the prediction horizon.

A. Problem Formulation

1) *Control Objective:* The torque $m(\mathbf{x})$ of the PMSM should track the reference m_r . Hence, it is natural to penalize the quadratic control deviation in the cost function:

$$J_1(\mathbf{x}) = \Delta m(\mathbf{x})^2 = (m(\mathbf{x}) - m_r)^2. \quad (15)$$

This is the first and most important control objective. The subsidiary control objective is to minimize the copper losses $P(i_{dq}) = (i_d^2 + i_q^2)r_s$, which obviously is a convex function. Hence,

$$J_2(\mathbf{x}) = \mathbf{x}^T \mathbf{I} \mathbf{x} \quad (16)$$

can be used to represent the loss minimization. The resistance can be replaced by the identity matrix \mathbf{I} , since it does not affect the optimizer.

2) *Constraints:* In addition, the MPC only calculates feasible voltage vectors. The voltage constraint for a standard two-level three phase VSI in the voltage plane is a hexagon in the dq-frame rotating with ω_{el} , thus the orientation depends on the rotor angle θ . It is described by the constraint $u \in \mathcal{U}(\theta)$. Furthermore the minimization of the objective functions (15) and (16) is subject to the system model (5).

3) *Lexicographic Optimization for PMSM:* The lexicographic approach is applied to a loss optimized torque control of a PMSM: Calculate the lexicographic minimum in the steady state of the objective vector

$$\mathbf{J}(\mathbf{x}) = [\Delta m(\mathbf{x})^2 \ \mathbf{x}^T \mathbf{I} \mathbf{x}]^T, \quad (17)$$

subject to (5) and $u \in \mathcal{U}(\theta)$. With J_1 and J_2 defined by (15) and (16), respectively. Note: The given problem has a unique minimizer and has no redundant objectives (straight forward application of remarks 1 and 2).

B. MPC Design

1) *Model Adaption for MPC:* First, the model from (5) is stacked [21]. So far, the dependencies from point in time k were omitted, but to make things clear they are used temporarily now. The vectors $\mathbf{x}_s(k) \in \mathbb{R}^{2n_p}$, $\mathbf{u}_s(k) \in \mathbb{R}^{2n_p}$ and $\mathbf{m}_s(k) \in \mathbb{R}^{n_p}$ encapsulate the discrete time values over a prediction horizon of length n_p as shown for the states:

$$\mathbf{x}_s(k)^T = [\mathbf{x}(k)^T \ \dots \ \mathbf{x}(k+n_p-1)^T]. \quad (18)$$

From now on all variables indexed with s are determined by the above scheme. This leads to the stacked model

$$\mathbf{x}_s(k+1) = \mathbf{A}_s \mathbf{x}(k) + \mathbf{B}_s \mathbf{u}_s(k) + \mathbf{G}_s \quad (19a)$$

$$\mathbf{m}_s(k, \mathbf{x}) = \mathbf{C}_s \mathbf{x}_s(k) + \mathbf{F}_s \quad (19b)$$

with the matrices $\mathbf{A}_s \in \mathbb{R}^{2n_p \times 2}$, $\mathbf{B}_s \in \mathbb{R}^{2n_p \times 2n_p}$, $\mathbf{G}_s \in \mathbb{R}^{2n_p}$, $\mathbf{C}_s \in \mathbb{R}^{n_p \times 2n_p}$ and $\mathbf{F}_s \in \mathbb{R}^{n_p}$. The matrices are calculated by repeatedly applying (5) over the prediction horizon n_p .

2) *Objective Function Construction and Optimization with Constraints*: The stacked variables can now be used to formulate the first control objective in the cost function in a single matrix equation $J_{1,s}(k, \mathbf{x}) = \|\Delta \mathbf{m}_s(k, \mathbf{x})\|_2^2$. For better readability two abuses of notation are made: time step dependency is omitted again and the objectives are now expressed in terms of the inputs \mathbf{u}_s instead of the states \mathbf{x}_s . Note: the states depend on the inputs: $\mathbf{x}_s(\mathbf{u}_s)$, which is obviously the case in discrete models but not accounted by notation. Using the substitution approach [21] the objectives are expressed solely depending on the stacked inputs \mathbf{u}_s :

$$J_{1,s}(\mathbf{u}_s) = \mathbf{u}_s^T \mathbf{H}_{1,s} \mathbf{u}_s + \mathbf{h}_{1,s}^T \mathbf{u}_s, \quad (20)$$

$$J_{2,s}(\mathbf{u}_s) = \mathbf{u}_s^T \mathbf{H}_{2,s} \mathbf{u}_s + \mathbf{h}_{2,s}^T \mathbf{u}_s. \quad (21)$$

The substitution procedure removes the additional equality constraint imposed by the system model (19) from the optimization problem. The optimization problem is only constrained by the inequality constraint induced by the inputs. The resulting feasible set is a simple set with a cheap projection operator (see [35] and the reference therein for details). The substitution procedure is an important step in modifying the objective function in order to use projective algorithms like the PFGM.

Now, the conditions for applying theorem 1 to $J_{1,s}$ and $J_{2,s}$ are revised. The statements in [36] on convexity preserving operations can be adopted to vector compositions of affine mappings:

Lemma 2: Suppose the function $f_c(\mathbf{x})$ is built by the composition of $f_a : \mathbb{R} \mapsto \mathbb{R}^k$ and the affine function $f_b : \mathbb{R}^n \mapsto \mathbb{R}^k$ defined by $f_c(\mathbf{x}) = f_a(f_b(\mathbf{x}))$. If $f_a(\mathbf{x})$ is strictly convex, function $f_c(\mathbf{x})$ is strictly convex.

Remark 4: $J_{1,s}$ and $J_{2,s}$ are still in the form of (8). Lemma 2 can be directly applied to show that $H_{2,s}$ in (21) is positive definite. In (20) we have $H_{1,s} = B_s^T C_s^T C_s B_s$ with $C_s \in \mathbb{R}^{n_p \times 2n_p}$. Hence, $H_{1,s}$ is proven to be not positive definite. Together with remarks 1 and 2 it follows, that the objective vector has no redundant objectives and that the lexicographic minimizer is unique.

Remark 5: If \mathbf{R}_s is a block diagonal matrix consisting of n_p blocks of \mathbf{R} , \mathbf{R}_s satisfies (10) and (11) and \mathbf{R}_s^T also does.

Remark 6: $J_{2,s}$ in (21) is strictly convex and $J_{2,s}$ in (21) leads to a linear manifold $\mathcal{X}_{1,s}$. Geometrically reasoning shows, that $\mathcal{X}_{1,s}$ and $\mathcal{X}_{t2,s}$ are linearly independent. With $\mathcal{X}_{1,s}$ given by definition 3 and $\mathcal{X}_{t2,s}$ defined by accordingly applying (13) to the stacked objectives $J_{1,s}$ and $J_{2,s}$.

Remark 7: Torque reference values are assumed to be feasible, that means they must be reachable. Thus, the minimizer lies within the feasible set and the voltage constraint can be dropped in steady state operation.

Remarks 4 to 7 show, that theorem 1 can be applied to calculate the lexicographic minimizer in steady state for the stacked objectives $J_{1,s}$ and $J_{2,s}$. This leads to

$$J_s(\mathbf{u}_s) = J_{1,s}(\mathbf{u}_s) + \lambda \|\mathbf{C}_s \mathbf{B}_s \mathbf{R}_s (2\mathbf{H}_{2,s} \mathbf{u}_s + \mathbf{H}_{2,s})\|_2^2. \quad (22)$$

Theorem 1 holds in steady state, but it does not hold when the constraint $u \in \mathcal{U}(\theta)$ is active. In the latter case (i.e., in transients) the objective function (22) turns out to be a weighted sum of the squared tracking error and the losses. Hence, the weighting factor λ is introduced and can be used to adjust the dynamic behavior. For a larger λ the second objective $J_{2,1}$ is given more weight. This leads to smaller inputs which slows down dynamics.

The resulting optimization problem is:

$$\mathbf{u}_s^* = \arg \min J_s(\mathbf{u}_s) \quad (23)$$

$$\text{s.t. } \mathbf{u}_s \in \mathcal{U}_s(\theta_s)$$

with the objective from (22) and $\mathcal{U}_s(\theta_s)$ being $\mathcal{U}(\theta)$ expanded over the prediction horizon. Up to here the approach was generally applicable to system models in the form of (5) and two prioritized objectives fulfilling the assumptions in theorem 1.

C. Overall Controller Scheme

For PMSM torque tracking (23) can be solved very efficiently using the PFGM optimization algorithm. The reason for that is the simple projection on the symmetrical polytopic constraint set $\mathcal{U}_s(\theta_s)$ also used in [37]. For faster computation times early termination and warm starting is used, see [8] for details. In Fig. 2 an overview of the resulting controller structure is given. As previously explained, the reference value m_r has to be limited to the maximum rating in steady state operation. This is done in the *Limiter* block. Effects like part-to-part variability and degradation with aging have to be considered for setting the correct limits, however this task is not in the scope of this contribution.

In general, applications may not provide knowledge of future reference values, thus $\hat{\mathbf{m}}_s$ is solely the n_p times repetition of \hat{m}_r and feeds as the stacked reference value for the calculation of $\Delta \mathbf{m}_s$. The *Prediction* block for θ_s is straight forward, since ω_{el} is known and assumed to be constant. The steps to calculate the matrices of the operation point depended affine state space model from section II-B are subsumed in the block *Matrix Generation*. The procedure presented in section IV-B is contained in the blocks *Objective Function*, *Constraint Generation* and *PFGM*. Note that the receding horizon strategy is applied. Hence, in Fig. 2 the *PFGM* block only applies the first entry of \mathbf{u}_s^* to the plant and the others are discarded.

D. Classification of the Presented Approach

To classify the proposed method a short summarizing overview of the properties is given now. It is an MPC scheme based on the receding horizon policy applied to a piecewise linear state space system with the machine currents as states in the dq-frame. Hence, the approach is classified as an CCS MPC with online optimization that can handle effects like cross coupling, saturation and magnetic anisotropy. The controlled variable is the electromagnetic torque of the machine, thus no current set values are needed. The rotating hexagonal voltage constrained in the dq-frame, owing the topology of the voltage source inverter (VSI), is

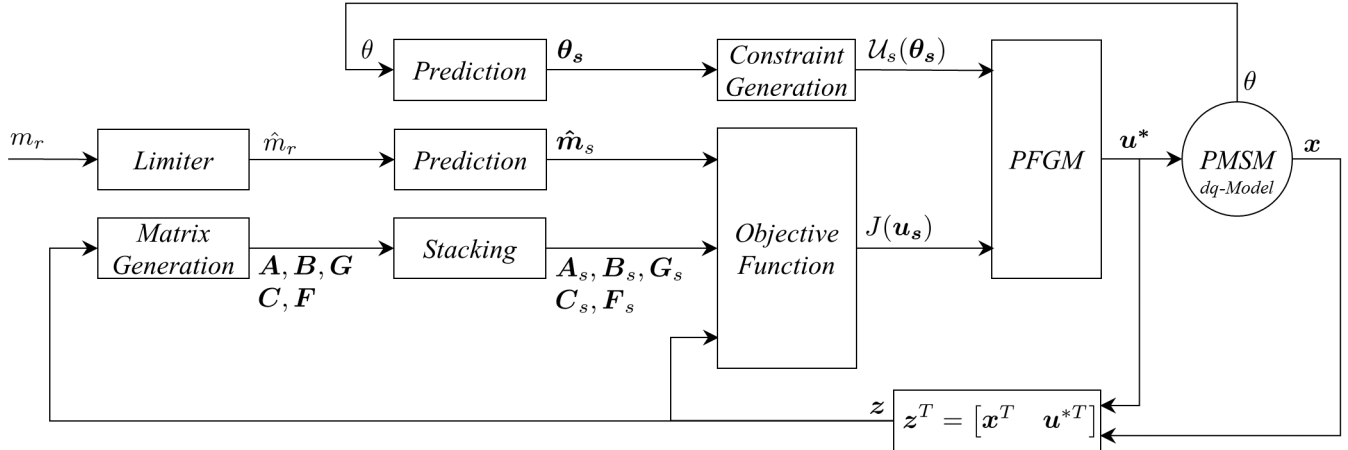


Fig. 2. Simulation overview

implemented without further simplifications. The MPC uses the PFGM for optimization of the objective function which considers ohmic losses as well as the torque tracking error over the prediction horizon.

V. SIMULATION STUDY

A. Simulation Setup

The PMSM model used for the simulation is experimentally validated, as it was also used in [29]. This machine model is implemented in MATLAB Simscape and simulated with a variable-step solver using a relative tolerance of 0.2%. The VSI is modeled as an ideal voltage source. The MPC is implemented in MATLAB Simulink and runs with a constant sampling time of $125 \mu\text{s}$ which corresponds to a switching frequency of the VSI of 8 kHz. All simulations are done in the dq-frame, thus the set voltage from the controller is directly applied as input to the machine model.

1) *Properties of the Simulated Drive:* The VSI is assumed to be a two-level inverter with a DC link voltage of 400 V. The model of the PMSM, that is designed for electric vehicles, has parameters as given in table I. The machine has interior permanent magnets and hence has an anisotropic magnetic circuit which causes a reluctance torque. Simulations shown here are carried out at constant angular velocity of the drive. The angular velocity is at nominal value (i.e., 5000 1/min).

TABLE I
MACHINE PROPERTIES

Nom. power	P_n	90.32 kW
Nom. speed	Ω_n	5000 min^{-1}
Number of pole pairs	p	3
Resistance	R_s	0.0284 Ω
PM flux linkage	Ψ_{PM}	0.1019 Wb

2) *Controller Setup:* The PI controller with a static mapping from reference torque to reference currents (classical MTPA and MTPV strategy) used for comparison is the same as in [38]. The prediction horizon of the MPC is set to $n_p = 2$. For the sake of completeness the termination conditions of the PFGM used for the simulation are stated, despite it is not the focus of this publication. The termination conditions of the PFGM used for the simulation are: The minimum step size in the decision variable, i.e. the inputs, is set to 0.0017 pu, which equals 3 Volts absolute. The maximum amount of iterations for the PFGM is set to 5. This setup has the aim to reach short computation times while preserving good performance.

B. Torque Dynamics and Comparison to Conventional PI Control

In Fig. 3 the response of the system to a step change in the reference torque can be seen for increasing values of λ from (a) to (c). Fig. 3 shows the desired behavior and confirms the idea and their theoretical realization. The increase in parameter $\lambda = 80$ does not deteriorate the tracking offset in steady state. Thus, it can be used as a tuning parameter for the transition speed of the torque. However, the settling of the torque in Fig. 3c shows slow oscillations with small amplitudes. The small offset that is observed stems from friction and iron losses of the PMSM model. Additionally, a faster response compared to PI current control is achieved. Of course PI control could be tuned to reach the reference faster but this will lead to higher overshoots. However, they can occur in the proposed MPC, but are rather traced back to computational inaccuracies of the optimization caused by the early termination technique.

C. Current Dynamics

Fig. 4 shows the elements of the normalized state vector $x^T = [i_d \ i_q]$ over time during the torque reference step shown in Fig. 3b. It is obvious, that the lexicographic MPC

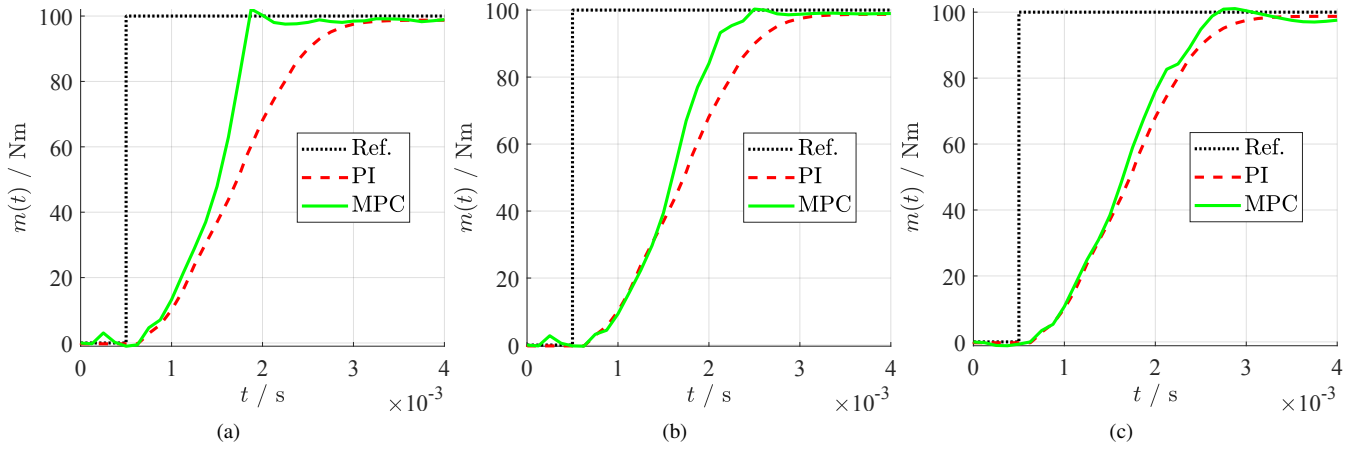


Fig. 3. Simulation results of the controlled variable m for different settings of λ : The PI controller is the same in all the graphs and allows the comparison between the different settings of λ . In (a) the fastest possible transition with $\lambda = 0.1$ is reached. Whereas (c) with $\lambda = 80$ shows the setting nearly producing same behavior as the PI controller. Setting $\lambda = 10$ leads to an intermediate dynamic shown in figure (b).

approach differs from the PI controller. Although the corresponding torque trajectory is aperiodic, the state trajectories in Fig. 4 oscillate around the state trajectory produced by PI control. The states move along a hyperbola in the state space that produces a constant torque as they approach their terminal value, which can be seen in Fig. 5. This makes it possible to achieve extreme fast torque responses without overshoots, since the oscillations in the currents are along these hyperbolas. It is observed, that for smaller λ the oscillations in the states increase, while they decrease for bigger ones. This is explicable by theory, since a bigger λ leads to an increased weighting of the losses during transients, which will also penalize current oscillations. A rise in torque dynamics leads to higher current oscillations.

The prediction horizon of $n_p = 2$ results in low computational demand, but leads already to good results as shown

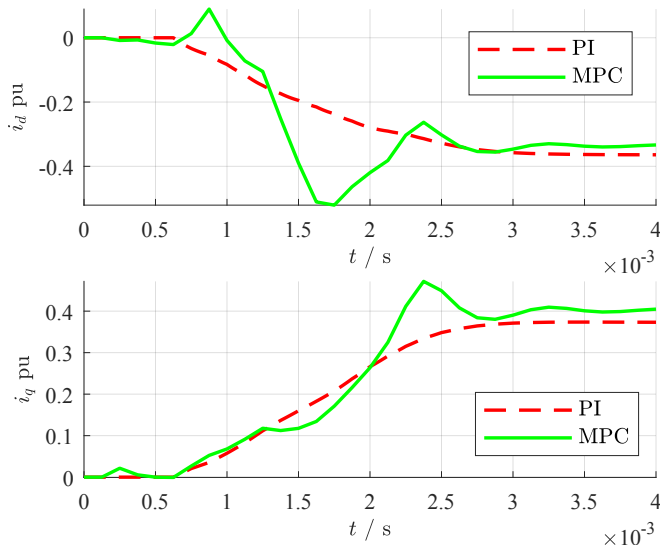


Fig. 4. Currents over time compared for the MPC with $\lambda = 10$ and the PI controller

before. As a result a maximum computation time of 27 ms during the transition in Fig. 3a could be reached with a desktop computer with CPU clock of 2.5 GHz and 8 GB RAM running at 800 MHz.

VI. SUMMARY AND CONCLUSION

Simulations show, that in drive applications there is a benefit in tracking torques compared to control strategies which track reference currents. Only with an approach directly minimizing the torque tracking error rather than the current tracking error this degree of freedom can be utilized for faster dynamics. This also holds for an MPC tracking states rather than the torque because the remaining degree of freedom is not left to the MPC and the underlying optimization step. However, torque tracking is a control objective with an ambiguous solution, i.e., all states lying on the constant torque hyperbola are satisfying this demand. To resolve this issue and for energy efficiency it is reasonable to incorporate losses in the objective function. This is the first time a lexicographic approach was used for torque tracking of electrical drives to solve this multi-objective problem. Furthermore a computationally efficient new method was presented to find the lexicographic minimizer in a single

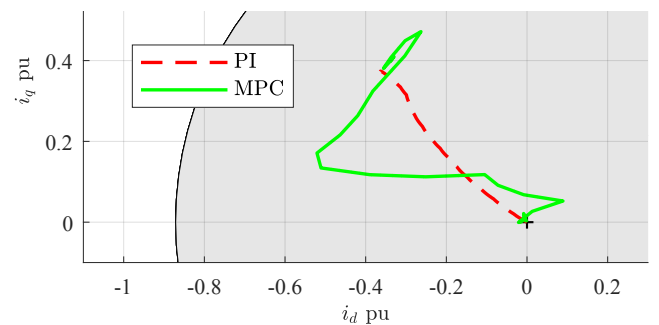


Fig. 5. Currents in the state space plane for the MPC with $\lambda = 10$ and the PI controller

optimization step for steady state operation. The proposed approach does not increase the tracking error in steady state by rising the loss weight. The dynamic of the resulting MPC can be set by a single parameter. State of the art methods for offset free tracking like disturbance observer or the delta input formulation can be used additionally if it is required by the application owing to model uncertainties or unknown disturbances. However, the proposed method will enhance the dynamic.

REFERENCES

- [1] M. Leuer and J. Böcker, "Real-time implementation of an online model predictive control for ipmsm using parallel computing on fpga," in *2014 International Power Electronics Conference (IPEC-Hiroshima 2014 - ECCE ASIA)*, May 2014, pp. 346–350.
- [2] M. Leuer and J. Böcker, "Voltage utilization in model predictive control for IPMSM," in *IEEE International Conference on Power Electronics, Drives and Energy Systems (PEDES)*, 2014, pp. 1–6.
- [3] M. Leuer, A. Rütting, and J. Böcker, "Efficiency-optimized model predictive torque control for ipmsm," in *2014 IEEE International Energy Conference (ENERGYCON)*, May 2014, pp. 9–13.
- [4] S. Saeidi and R. Kennel, "Finite set model predictive control with imposed target dynamic," in *PCIM Europe 2014; International Exhibition and Conference for Power Electronics, Intelligent Motion, Renewable Energy and Energy Management*, May 2014, pp. 1–8.
- [5] M. Preindl, "Robust control invariant sets and lyapunov-based MPC for IPM synchronous motor drives," *IEEE Trans. Ind. Electron.*, vol. 63, no. 6, pp. 3925–3933, June 2016.
- [6] M. Leuer and J. Böcker, "Switching strategy for direct model predictive control in power converter and drive applications with high switching frequency," in *2015 6th International Conference on Automation, Robotics and Applications (ICARA)*, Feb 2015, pp. 569–573.
- [7] W. Xie, X. Wang, F. Wang, W. Xu, R. Kennel, D. Gerling, and R. Lorenz, "Finite-control-set model predictive torque control with a deadbeat solution for pmsm drives," *IEEE Trans. Ind. Electron.*, vol. 62, no. 9, pp. 5402–5410, Sept 2015.
- [8] M. Preindl, S. Bolognani, and C. Danielson, "Model predictive torque control with PWM using fast gradient method," in *IEEE Applied Power Electronics Conference and Exposition (APEC)*, no. 28, March 2013, pp. 2590–2597.
- [9] G. Cimini, D. Bernardini, A. Bemporad, and S. Levijoki, "Online model predictive torque control for permanent magnet synchronous motors," in *IEEE Int. Conf. on Industrial Technology (ICIT)*, March 2015, pp. 2308–2313.
- [10] R. Errouissi, A. Al-Durra, S. Muyeen, and S. Leng, "Continuous-time model predictive control of a permanent magnet synchronous motor drive with disturbance decoupling," *IET Electric Power Applications*, vol. 11(5), p. 697 706, 2017.
- [11] Y. E. Nesterov, "Introductory lectures on convex optimization: A basic course," *Kluwer Academic Publishers*, 2003, for Citation see Nesterov-1998-LecPP for PDF.
- [12] J. Jerez, P. Goulart, S. Richter, G. Constantinides, E. Kerrigan, and M. Morari, "Embedded online optimization for model predictive control at megahertz rates," *Automatic Control, IEEE Transactions on*, vol. 59, no. 12, pp. 3238–3251, Dec 2014.
- [13] P. Patrinos and A. Bemporad, "An accelerated dual gradient-projection algorithm for embedded linear model predictive control," *IEEE Transactions on Automatic Control*, vol. 59, no. 1, pp. 18–33, 2014.
- [14] R. A. Braker and L. Y. Pao, "An application of the fast gradient method to model predictive control of an atomic force microscope x-y stage," in *2017 IEEE Conference on Control Technology and Applications (CCTA)*, Aug 2017, pp. 111–116.
- [15] H. Peyrl, J. Liu, and T. Geyer, "An fpga implementation of the fast gradient method for solving the model predictive pulse pattern control problem," *2013 IEEE International Symposium on Sensorless Control for Electrical Drives and Predictive Control of Electrical Drives and Power Electronics (SLED/PRECEDE)*, pp. 1–6, 2013.
- [16] M. Preindl, "Novel model predictive control of a pm synchronous motor drive," Ph.D. dissertation, University of Padua, 2013.
- [17] J.-F. Stumper, A. Dötlinger, J. Jung, and R. Kennel, "Predictive control of a permanent magnet synchronous machine based on real-time dynamic optimization," in *European Conference on Power Electronics and Applications*, 2011.
- [18] J.-F. Stumper, A. Dötlinger, and R. Kennel, "Classical model predictive control of a permanent magnet synchronous motor," *European Power Electronics and Drives Journal*, vol. 22, no. 3, pp. 24–31, 2012.
- [19] P. Vaclavek and P. Blaha, "Pmsm model discretization for model predictive control algorithms," in *System Integration (SII), 2013 IEEE/SICE International Symposium on*, Dec 2013, pp. 13–18.
- [20] U. Maeder and M. Morari, "Offset-free reference tracking with model predictive control," *Automatica*, vol. 46, no. 9, pp. 1469 – 1476, 2010.
- [21] F. Borrelli, A. Bemporad, and M. Morari, "Predictive control for linear and hybrid systems," *Cambridge University Press*, 2017.
- [22] L. Wang, *Model Predictive Control System Design and Implementation Using MATLAB*. Springer Publishing Company, Incorporated, 2009.
- [23] A. Hanif, Q. Ahmed, A. I. Bhatti, and G. Rizzoni, "Genetic algorithms optimized multi-objective controller for an induction machine based electrified powertrain," in *2017 IEEE Conference on Control Technology and Applications (CCTA)*, Aug 2017, pp. 853–858.
- [24] C. A. Rojas, J. Rodriguez, F. Villarreal, J. R. Espinoza, C. A. Silva, and M. Trincado, "Predictive torque and flux control without weighting factors," *IEEE Trans. Ind. Electron.*, vol. 60, no. 2, pp. 681–690, Feb 2013.
- [25] S. Carpiuc and C. Lazar, "Energy-efficient model predictive speed control of permanent magnet synchronous machine based automotive traction drives," in *IEEE Vehicle Power and Propulsion Conference (VPPC)*, Oct 2014, pp. 1–6.
- [26] M. Preindl, "Robust control invariant sets and lyapunov-based MPC for IPM synchronous motor drives," *IEEE Trans. Ind. Electron.*, vol. 63, pp. 3925–3933, 2016.
- [27] M. L. Tyler and M. Morari, "Propositional logic in control and monitoring problems," *Automatica*, vol. 35, no. 4, pp. 565 – 582, 1999, lexico.
- [28] E. C. Kerrigan and J. M. Maciejowski, "Designing model predictive controllers with prioritised constraints and objectives," in *IEEE International Symp. on Comput. Aided Control Syst. Design*, 2002, pp. 33–38.
- [29] J. Richter, T. Gemaßner, and M. Doppelbauer, "Predictive current control of saturated cross-coupled permanent magnet synchronous machines," in *International Symposium on Power Electronics, Electrical Drives, Automation and Motion*, June 2014, pp. 830–835.
- [30] A. b. Rahideh and M. Shaheed, "Constrained output feedback model predictive control for nonlinear systems," *Control Engineering Practice*, vol. 20, no. 4, pp. 431–443, 2012.
- [31] C. A. Rojas, J. I. Yuz, M. Aguirre, and J. Rodriguez, "A comparison of discrete-time models for model predictive control of induction motor drives," in *IEEE Int. Conf. on Industrial Technology*, 2015, pp. 568–573.
- [32] M. Graf, L. Otava, and L. Buchta, "Simple linearization approach for MPC design for small PMSM with field weakening performance," in *Conf. on Programmable Devices and Embedded Systems (PDES)*, vol. 13, 2015, pp. 159 – 164.
- [33] J. Rodríguez-Millán, A. Patete, and C. González, "Picard discretization of nonlinear systems: Symbolic or numeric implementation?" *Lecture Notes in Computer Science (LNCS)*, vol. 4739, pp. 121–129, 2007.
- [34] M. M. Mäkelä and Y. Nikulin, "On cone characterizations of strong andlexicographic optimality inconvex multiobjective optimization," *Journal of Optimization Theory and Applications*, vol. 143, no. 3, p. 519, May 2009.
- [35] S. Richter, C. Jones, and M. Morari, "Computational complexity certification for real-time MPC with input constraints based on the fast gradient method," *IEEE Trans. Autom. Control*, vol. 57, no. 6, pp. 1391–1403, 2012.
- [36] S. P. Boyd and L. Vandenberghe, *Convex optimization*. Cambridge [u.a.]: Cambridge Univ. Press, 2004, hier auch später erschienene, unveränderte Nachdrucke.
- [37] S. Richter, S. Mariéthoz, and M. Morari, "High-speed online MPC based on a fast gradient method applied to power converter control," in *American Control Conference*, June 2010, pp. 4737–4743.
- [38] T. Gemaßner, M. Schnarrenberger, and H. Späth, "Simple strategy of overmodulation in control of interior permanent magnet synchronous machines for improving efficiency in automotive applications," *PCIM Europe*, pp. 231–238, 2013.

This is the accepted version of the following article: C. Schnurr, S. Hohmann, J. Kolb „Novel lexicographic MPC for loss optimized torque control of nonlinear PMSM“, 2018 IEEE Conference on Control Technology and Applications (CCTA), pp. 690-697, 2018.

The article has been published in final form at <https://doi.org/10.1109/CCTA.2018.8511455>.

© 2018 IEEE