PMSM Model Predictive Control With Field-Weakening Implementation

Zbynek Mynar, Libor Vesely, and Pavel Vaclavek, *Senior Member, IEEE*

***Abstract*—Permanent-magnet synchronous machine (PMSM) drives have become popular for motion control applications due to their performance and high torque-to-weight ratio. The complex task of PMSM control in high-performance applications is currently usually resolved with classical vector control. Modern control techniques such as model predictive control (MPC) can provide significant benefits over field-oriented control, especially with straightforward controller tuning and constraints handling. Unfortunately, these new algorithms usually suffer from problems with their computational complexity. In this paper, we present a PMSM control method based on an explicit MPC with a novel linearization and constraints handling method, allowing natural field weakening. The algorithm was designed with respect to computationally feasible implementation in the control hardware. The proposed control algorithm has been proved and successfully verified in both simulation and implementation on a real motor.**

***Index Terms*—Constraints, explicit model predictive control (MPC), field weakening, predictive control, synchronous machine.**

I. Introduction

THE demands placed upon the performance of industrial permanent-magnet synchronous machine (PMSM) controllers are more intensive than ever. PMSMs are heavily used not only in servo drives, but also as traction drives in modern vehicles and in many other applications [1], [2]. These applications benefit from their reliability and good power-to-weight ratio, but they are not as easy to control as the older dc drives. Modern applications also create demands going beyond simple speed control, requiring energy efficiency, high dynamics, and the ability to exploit drive capabilities to their limits. These requirements drive the invention of new algorithms and call for further investigation into their properties, their weaknesses, and strong points.

The classical approach to PMSM control is based on cascaded field-oriented vector control (FOC) [3], [4]. This structure uses several PI controllers with cascaded control loops for magnetic flux magnitude, torque control, speed and position control. FOC is relatively simple and it has been widely adopted by industry. It has proven itself over many years of successful use in various applications, but it also has its limits. The first problem is that controllers of individual control loops are run practically independently, which means that overall control performance optimization, as well as efficient constraint handling, is nearly impossible. Another problem is that the parameters of these controllers do not have a clear physical interpretation. This causes problems with controllers when fine tuning them for complex control objectives, e.g., a combination of drive dynamics and energy efficiency. Even though there are new modifications [5] of classical field-oriented control and direct torque control, the performance of these classical algorithms has not significantly improved.

One of the modern approaches to drive control is model predictive control (MPC). While the idea of MPC itself is relatively old, only the recent development of controllers with higher computational powers enables its implementation in systems with fast dynamics, such as electrical drives. An MPC controller includes a model of the plant that is used to predict the behavior of the system, and the related optimal state-space control problem is resolved with optimization methods. The great advantage of MPC is a proper constraints handling, as well as the possibility of defining control objectives in a very natural way.

The major obstacles in an MPC implementation are in its computational complexity and the still limited power of industrially applied control hardware. However, with the development of hardware and solvers, MPC is being implemented on ever faster plants [6]–[8]. This has also facilitated the appearance of the first implementations of MPC on a PMSM [9].

The biggest challenge in MPC implementation for ac electrical drive control is the nonlinear behavior of the drives [10]–[12]. Considering the computational power of the current industry-standard hardware, linear MPC is the best option to achieve feasible implementation of control systems with fast dynamics, such as electrical drives [13], [14]. Several articles have been published proposing ac drive linearization by assuming that nonlinear terms are measured disturbances [15]. While this approach allows for computationally efficient implementation of a linear explicit MPC [16], the measured disturbances are kept constant on the prediction horizon, limiting the achievable control performance.

Other approaches tend to retain the classical cascaded field-oriented vector control structure, in which MPC controllers

Manuscript received May 31, 2015; revised October 27, 2015 and February 17, 2016; accepted March 10, 2016. Date of publication April 25, 2016; date of current version July 08, 2016. This work was supported in part by the Technology Agency of the Czech Republic under the Project TE01020197 Center for Applied Cybernetics 3 and in part by ECSEL JU under the project 621429 EMC^2. This work was performed at the Central European Institute of Technology (CEITEC) with research infrastructure supported by the project CZ.1.05/1.1.00/02.0068 financed by the European Regional Development Fund.

Z. Mynar is with the Faculty of Electrical Engineering and Communication, Brno University of Technology, 616 00 Brno, Czech Republic (e-mail: xmynar03@stud.feec.vutbr.cz).

P. Vaclavek and L. Vesely are with the Central European Institute of Technology and with the Faculty of Electrical Engineering and Communication, Brno University of Technology, 612 00, Brno, Czech Republic (e-mail: pavel.vaclavek@ceitec.vutbr.cz; libor.vesely@ceitec.vutbr.cz).

Digital Object Identifier 10.1109/TIE.2016.2558165

replace either the current or the speed controllers [17]–[21]. A problem with this implementation is that the optimal control is located in two separate controllers and the system is not controlled as one overall entity. This type of implementation often struggles to handle the constraints properly and to achieve the overall control objectives. Another possibility is to design the algorithm based on a finite control set [22]–[24], although, unfortunately, the computational complexity is usually still very high and the tradeoff between the limited number of control values and the switching frequency can lead to significant current and torque ripples.

This paper presents a linear approach to the MPC of a PMSM in rotor coordinates (vector control), where a single linear multiple-input multiple-output predictive controller replaces both the current and speed controllers. The control algorithm is based on an explicit MPC, in which nonlinear terms are handled in such a way that allows it to perform optimization of the magnetic flux, including flux weakening in a high-speed region.

The intrinsic implementation of field weakening is one of the main advantages of the proposed algorithm. Most classical PMSM control schemes use an additional algorithm for magnetic flux magnitude optimization [25], [26]. Authors of MPC algorithms have already published results showing the possibility of direct implementation of field weakening as a natural feature of predictive control based on a finite control set [27], [28]. Unfortunately, these algorithms require a too high computational power and that is why they were tested only in simulation or in laboratory experiments with a very short, or even a one-step, prediction horizon. In contrast, this paper describes a computationally feasible MPC algorithm that provides a flux weakening control without any additional flux controller.

The algorithm proposed in this paper was implemented on a National Instruments CompactRIO (cRio) system, which combines a field-programmable gate array (FPGA) and a real-time (RT) controller. Some authors have already used an implementation of drive control on a pure FPGA [29]. The proposed design uses the benefits of the RT and FPGA combination to face the large memory demands of a linear explicit MPC implementation. The results of the successful experiments on a real PMSM are shown in the last part of this paper.

II. PMSM MPC

A. PMSM Model

The continuous-time model of a PMSM in a rotating frame consists of electrical dynamics and mechanical dynamics. The part describing the electrical dynamics can be expressed as

$$\begin{aligned}\frac{\mathrm{d}i_{\mathrm{sd}}}{dt} &= \frac{u_{\mathrm{sd}}}{L_d} + \frac{L_q}{L_d}P_p\omega_m i_{\mathrm{sq}} - \frac{R_s}{L_d}i_{\mathrm{sd}}\\ \frac{\mathrm{d}i_{\mathrm{sq}}}{dt} &= \frac{u_{\mathrm{sq}}}{L_q} - \frac{L_d}{L_q}P_p\omega_m i_{\mathrm{sd}} - \frac{R_s}{L_q}i_{\mathrm{sq}} - \frac{\Psi_{\mathrm{PM}}}{L_q}P_p\omega_m\\ \frac{\mathrm{d}\varphi_m}{dt} &= \omega_m \end{aligned}\tag{1}$$

where

$u_{\mathrm{sd}}, u_{\mathrm{sq}}$	stator voltage components in the rotating frame;
$i_{\mathrm{sd}}, i_{\mathrm{sq}}$	stator current components in the rotating frame;
R_s	stator winding resistance;
L_d, L_q	stator inductance components;
Ψ_{PM}	electromotive force (EMF) constant;
P_p	number of pole pairs;
ω_m	rotor mechanical angular speed.

The electromagnetic torque produced by a PMSM can be described as

$$T_{\mathrm{el}} = \frac{3}{2}P_p\left[\Psi_{\mathrm{PM}}i_{\mathrm{sq}} + (L_d - L_q)\, i_{\mathrm{sd}}i_{\mathrm{sq}}\right].\tag{2}$$

The mechanical dynamics of the PMSM model is then

$$\frac{\mathrm{d}\omega_m}{dt} = \frac{1}{J}\left(T_{\mathrm{el}} - T_l\right)\tag{3}$$

where J is the moment of inertia and T_l is the load torque. The Euler method was used to obtain a discrete time model. While there are more precise discretization techniques [23], [30], The Euler method provides a simpler model without introducing any unwanted additional nonlinear terms

$$\begin{aligned}i_{\mathrm{sd}}(k+1) = {}& \left(1 - T_s\frac{R_s}{L_d}\right)i_{\mathrm{sd}}(k) + T_sP_p\frac{L_q}{L_d}\omega_m(k)i_{\mathrm{sq}}(k)\\ &+\frac{T_s}{L_d}u_{\mathrm{sd}}(k)\end{aligned}\tag{4}$$

$$\begin{aligned}i_{\mathrm{sq}}(k+1) = {}& \left(1 - T_s\frac{R_s}{L_q}\right)i_{\mathrm{sq}}(k) - T_sP_p\frac{L_d}{L_q}\omega_m(k)i_{\mathrm{sd}}(k)\\ &- T_sP_p\frac{\Psi_{\mathrm{PM}}}{L_q}\omega_m(k) + \frac{T_s}{L_q}u_{\mathrm{sq}}(k)\end{aligned}\tag{5}$$

$$\begin{aligned}\omega_m(k+1) = {}& \omega_m(k) + \frac{T_s}{J}\left(\frac{3}{2}P_p\left[\Psi_{\mathrm{PM}}i_{\mathrm{sq}}(k)\right.\right.\\ &\left.\left.+ (L_d - L_q)\, i_{\mathrm{sd}}(k)i_{\mathrm{sq}}(k)\right] - T_l\right)\end{aligned}\tag{6}$$

where T_s is the sampling period.

B. MPC Problem Definition

For simplicity, we will consider the discrete time dynamical system model in the form

$$\begin{aligned}\boldsymbol{x}(k+1) &= \boldsymbol{A}\boldsymbol{x}(k) + \boldsymbol{B}\boldsymbol{u}(k)\\ \boldsymbol{y}(k) &= \boldsymbol{C}\boldsymbol{x}(k) + \boldsymbol{D}\boldsymbol{u}(k)\end{aligned}\tag{7}$$

where $\boldsymbol{x} \in \mathcal{X} \subset \mathbb{R}^n$ is the system state limited to constraints $\mathcal{X}$, $\boldsymbol{u} \in \mathcal{U} \subset \mathbb{R}^m$ are system inputs limited to constraints $\mathcal{U}$, and $\boldsymbol{y} \in \mathcal{Y} \subset \mathbb{R}^r$ are system outputs limited to constraints $\mathcal{Y}$. For this class of systems, the MPC can be considered to be a task of multiparametric programming [31]. The control goal can be prescribed by the cost function

$$\begin{aligned}J(k) = {}& \boldsymbol{x}(k+n_p)^{\mathrm{T}}\boldsymbol{P}\boldsymbol{x}(k+n_p)\\ &+\sum_{j=0}^{n_p-1}\left[\boldsymbol{x}(k+j)^{\mathrm{T}}\boldsymbol{Q}\boldsymbol{x}(k+j)\right.\\ &\left.+\boldsymbol{u}(k+j)^{\mathrm{T}}\boldsymbol{R}\boldsymbol{u}(k+j)\right]\end{aligned}\tag{8}$$

where n_p is the prediction horizon length and $\boldsymbol{P}, \boldsymbol{Q},$ and $\boldsymbol{R}$ are weighting matrices penalizing the final state, the current state, and the control action.

Matrix $\boldsymbol{R}$ is symmetrical, positive definite, matrices $\boldsymbol{P}$ and $\boldsymbol{Q}$ are symmetrical, positive semidefinite, which is necessary for the optimization problem to be convex and solved as a quadratic programming optimisation problem.

The optimal control action can be solved by minimizing the criterion (8) with respect to constraints

$$\begin{aligned} &\forall j = 1 \dots n_p \quad \boldsymbol{x}(k+j) \in \mathcal{X} \\ &\forall j = 0 \dots n_p \quad \boldsymbol{u}(k+j) \in \mathcal{U} \\ &\forall j = 0 \dots n_p - 1 \ \boldsymbol{x}\ (k+j+1) = \boldsymbol{A}\boldsymbol{x}(k+j) + \boldsymbol{B}\boldsymbol{u}(k+j) \\ &\forall n_u < j \le n_p \ \boldsymbol{u}\ (k+j) = \boldsymbol{K}\boldsymbol{x}(k+j) \end{aligned} \tag{9}$$

where n_u is the control horizon length and $\boldsymbol{K}$ is the feedback gains matrix, which is used to guarantee stability beyond the control horizon and can be selected in multiple ways [15]. This feedback matrix is a part of the control law definition only when the control horizon is shorter than the prediction horizon. This is not the case of the proposed and tested implementation, where $n_u = n_p$ (see Section III, in this case, all the control values are result of solution of the optimization problem) and it is included in (9) only for completeness. The system state values can be easily found using (7) as

$$\boldsymbol{x}(k+j) = \boldsymbol{A}^j \boldsymbol{x}(k) + \sum_{i=0}^{j-1} \boldsymbol{A}^i \boldsymbol{B}\boldsymbol{u}(k+j-1-i). \tag{10}$$

Using (10), the criterion (8) can be [32] written as

$$J(k) = \frac{1}{2}\boldsymbol{x}(k)^{\mathrm{T}}\boldsymbol{Y}\boldsymbol{x}(k) + \min_{\boldsymbol{U}} \left[\frac{1}{2}\boldsymbol{U}^{\mathrm{T}}\boldsymbol{H}\boldsymbol{U} + \boldsymbol{x}(k)^{\mathrm{T}}\boldsymbol{F}\boldsymbol{U}\right] \tag{11}$$

with constraints

$$\boldsymbol{G}\boldsymbol{U} \le \boldsymbol{W} + \boldsymbol{E}\boldsymbol{x}(k) \tag{12}$$

where $\boldsymbol{U} = \left[\boldsymbol{u}(k)^{\mathrm{T}}, \dots, \boldsymbol{u}(k+n_u-1)^{\mathrm{T}}\right]^{\mathrm{T}} \in \mathcal{R}^{\mathrm{mn_u}}$, while matrices $\boldsymbol{H}$, $\boldsymbol{F}$, $\boldsymbol{Y}$, $\boldsymbol{G}$, $\boldsymbol{W}$, and $\boldsymbol{E}$ can be computed from the original weighting matrices $\boldsymbol{Q}$, $\boldsymbol{R}$ and constraints (9) using the substitution (10). The problem (11), (12), is a *quadratic programming* optimization problem with known computationally efficient algorithms of the solution.

The aforementioned approach is limited to linear systems only, which brings implementation difficulties in the case of an ac machine with a nonlinear behavior. On the other hand, this method is very useful because of the possibility of reducing the real-time computational demands. The MPC, based on the problem definition (11), (12), can be resolved offline as the obtained control law is piecewise affine [32] and can be implemented as so-called *explicit control*.

C. Classic Approach to PMSM Explicit MPC

As stated in the previous section, the explicit MPC algorithm will be based on a model (7). The difference between L_d and L_q is usually small for surface permanent magnet machines (SPMSM), but also for many other PMSMs, and then, the torque equation (6) becomes linear, assuming

$$L_d \approx L_q = L. \tag{13}$$

The classical approach to remove the remaining nonlinearities in (4) and (5) is based on the idea that these nonlinear terms are considered to be measured disturbances [16]. Assuming the measured disturbances to be

$$\begin{aligned} \widehat{\omega_m i_{\mathrm{sd}}}(k) &= \omega_m(k) i_{\mathrm{sd}}(k) \\ \widehat{\omega_m i_{\mathrm{sq}}}(k) &= \omega_m(k) i_{\mathrm{sq}}(k) \end{aligned} \tag{14}$$

we can obtain the linearized current equations

$$\begin{aligned} i_{\mathrm{sd}}(k+1) = &\left(1 - T_s \frac{R_s}{L}\right) i_{\mathrm{sd}}(k) + T_s P_p \widehat{\omega_m i_{\mathrm{sq}}}(k) \\ &+ \frac{T_s}{L} u_{\mathrm{sd}}(k) \end{aligned} \tag{15}$$

$$\begin{aligned} i_{\mathrm{sq}}(k+1) = &\left(1 - T_s \frac{R_s}{L}\right) i_{\mathrm{sq}}(k) - T_s P_p \widehat{\omega_m i_{\mathrm{sd}}}(k) \\ &- T_s P_p \frac{\Psi_{\mathrm{PM}}}{L} \omega_m(k) + \frac{T_s}{L} u_{\mathrm{sq}}(k). \end{aligned} \tag{16}$$

The controlled system model is now linear, but there are still some issues preventing it from directly being used for an MPC design. The first problem is the unknown value of the load torque T_l. Here, it will be estimated using the load torque observer [33]–[35] and added into the model as a measured disturbance in the form of an equivalent load q-axis stator current

$$i_{\mathrm{sql}} = \frac{2T_l}{3P_p \Psi_{\mathrm{PM}}} = \frac{T_l}{k_t} \tag{17}$$

where k_t is the motor torque constant. This allows penalizing the difference between i_{sq} and its steady value, rather than zero, which improves the control performance under load. Another problem is that the stator voltages (system inputs) are not zero at the steady state. To keep the optimization problem convex, the penalization matrix $\boldsymbol{R}$ must be positive definite, which means a nonzero system input penalization. Both these problems can be solved by assuming that the controller output voltages are replaced by their differences $\Delta u_{\mathrm{sd}}, \Delta u_{\mathrm{sq}}$. The summation producing the actual control voltages is included into the controlled system model for controller design purposes

$$\boldsymbol{u}_{\mathrm{sdq}}(k) = \boldsymbol{u}_{\mathrm{sdq}}(k-1) + \Delta\boldsymbol{u}_{\mathrm{sdq}}(k) \tag{18}$$

where $\boldsymbol{u}_{\mathrm{sdq}}(k) = \left[\,u_{\mathrm{sd}}(k), \quad u_{\mathrm{sq}}(k)\,\right]^{\mathrm{T}}$ is the stator voltage and $\Delta\boldsymbol{u}_{\mathrm{sdq}}(k) = \left[\,\Delta u_{\mathrm{sd}}(k), \quad \Delta u_{\mathrm{sq}}(k)\,\right]^{\mathrm{T}}$ is the controller output. The main advantage is that it does not cause an increase of the system relative degree. It is important to keep the relative degree of the systems as low as possible, to be able to obtain a good performance without the necessity of a longer prediction horizon and a significantly increased computational complexity [28].

It is also necessary to augment the control set point into the state vector to be able to achieve tracking control [16]. Using the aforementioned assumptions, the drive model can be described as (7) with (19)–(22), shown at the bottom of the page, where $\omega_{\mathrm{mw}}(k)$ is the speed set point. Output matrix $\boldsymbol{C}$ is an identity matrix, as all the system states are assumed to be measurable or at least observable. Feed-forward matrix $\boldsymbol{D}$ is a zero matrix, as the system has a nonzero relative degree and there is no direct connection between the system inputs and outputs.

The last step in the MPC problem definition is the proper selection of the weighting matrices in the criterion (8). The model state weighting matrix can be selected as

$$\boldsymbol{P} = \boldsymbol{Q}$$

$$= \begin{bmatrix} q_{i_d} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & q_{i_q} & 0 & 0 & 0 & 0 & 0 & 0 & -q_{i_q} \\ 0 & 0 & q_\omega & 0 & 0 & 0 & 0 & -q_\omega & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -q_\omega & 0 & 0 & 0 & 0 & q_\omega & 0 \\ 0 & -q_{i_q} & 0 & 0 & 0 & 0 & 0 & 0 & q_{i_q} \end{bmatrix} \qquad (23)$$

where q_{i_d} and q_{i_q} are the stator current penalties, and q_ω is speed control error penalty. According to the requirements (8), the matrix $\boldsymbol{R}$ must be positive definite, even in the case that there is no technical reason for penalizing the control action increments $\Delta\boldsymbol{u}_{\mathrm{sdq}}$. The matrix $\boldsymbol{R}$ will be defined as

$$\boldsymbol{R} = \begin{bmatrix} r_{\Delta u_d} & 0 \\ 0 & r_{\Delta u_q} \end{bmatrix}. \qquad (24)$$

The explicit MPC controller can now be designed using various numerical tools, and one of the most convenient possibilities is use of the Multi-Parametric Toolbox (MPT) for the MATLAB-Simulink environment [15], [16], [36].

D. Proposed Approach to Model Linearization and Constraints Implementation

The usual way of linearizing PMSM machine model [see (15) and (16)] leads to loss of information that is critical for the field-weakening operation. Namely, the relationship between the stator current $\boldsymbol{i}_{\mathrm{sdq}}$, speed ω_m, and the coupling parts of the stator voltage. The problem is caused by the fact that the measured disturbances relating to the back electromotive force are held constant on the prediction horizon [37]. The method of implementing the MPC controller for a PMSM proposed in this paper incorporates linearization of the mathematical model with an approximation of this information, and thus, allows the field-weakening operation.

To eliminate the nonlinear mathematical operation of multiplying two states, let us assume a constant rotor speed $\omega_m = \Omega_{\mathrm{mi}}$. This leads to the current model described by

$$i_{\mathrm{sd}}(k+1) = \left(1 - T_s\frac{R_s}{L}\right) i_{\mathrm{sd}}(k) + T_s P_p \Omega_{\mathrm{mi}} i_{\mathrm{sq}}(k) + \frac{T_s}{L} u_{\mathrm{sd}}^{\mathrm{ctrl}}(k) \qquad (25)$$

$$i_{\mathrm{sq}}(k+1) = \left(1 - T_s\frac{R_s}{L}\right) i_{\mathrm{sq}}(k) - T_s P_p \Omega_{\mathrm{mi}} i_{\mathrm{sd}}(k) - T_s P_p \frac{\Psi_{\mathrm{PM}}}{L}\omega_m(k) + \frac{T_s}{L} u_{\mathrm{sq}}^{\mathrm{ctrl}}(k) \qquad (26)$$

where $\boldsymbol{u}_{\mathrm{sdq}}^{\mathrm{ctrl}} = \left[u_{\mathrm{sd}}^{\mathrm{ctrl}}, \quad u_{\mathrm{sq}}^{\mathrm{ctrl}}\right]^{\mathrm{T}}$ is the stator voltage calculated by the MPC controller. The coupling parts of the stator voltage

$$\boldsymbol{x}(k) = [i_{\mathrm{sd}}(k),\ i_{\mathrm{sq}}(k),\ \omega_m(k),\ u_{\mathrm{sd}}(k-1),\ u_{\mathrm{sq}}(k-1),\ \widehat{\omega_m i_{\mathrm{sd}}}(k),\ \widehat{\omega_m i_{\mathrm{sq}}}(k),\ \omega_{\mathrm{mw}}(k),\ i_{\mathrm{sqI}}(k)]^{\mathrm{T}} \qquad (19)$$

$$\boldsymbol{u}(k) = \left[\Delta u_{\mathrm{sd}}(k), \quad \Delta u_{\mathrm{sq}}(k)\right]^{\mathrm{T}} \qquad (20)$$

$$\boldsymbol{A} = \begin{bmatrix} \frac{L - T_s R_s}{L} & 0 & 0 & \frac{T_s}{L} & 0 & 0 & T_s P_p & 0 & 0 \\ 0 & \frac{L - T_s R_s}{L} & -\frac{T_s P_p \Psi_{\mathrm{PM}}}{L} & 0 & \frac{T_s}{L} & -T_s P_p & 0 & 0 & 0 \\ 0 & \frac{T_s k_t}{J} & 1 & 0 & 0 & 0 & 0 & 0 & -\frac{T_s k_t}{J} \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \qquad (21)$$

$$\boldsymbol{B} = \begin{bmatrix} \frac{T_s}{L} & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{T_s}{L} & 0 & 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix}^{\mathrm{T}} \qquad (22)$$

are now modeled as a function of the stator currents, which enables the controller to create a negative i_{sd} current during field weakening. The accuracy of this model depends on how close the selected speed constant Ω_m is to the actual speed ω_m. Since it is expected that the explicit controller will be used, it is highly undesirable to recalculate the controller based on a model with a new value of Ω_m every time the speed changes significantly. Multiple machine models, are therefore, implemented, for each $i \in \mathcal{I}$ region of speed ω_m, which begins at speed ω_{mi}, ends at speed $\omega_{\mathrm{mi}+1}$ and with its own speed constant $\Omega_{\mathrm{mi}} \in \langle \omega_{\mathrm{mi}}, \omega_{\mathrm{mi}+1} \rangle$. A minimum of two regions must be implemented, each with a Ω_{mi} constant of the opposite sign to ensure a proper speed reversibility.

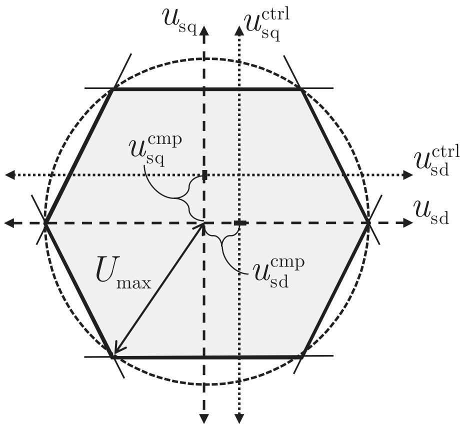
To implement a linear MPC controller for multiple systems of the same topology, but with different dynamics, for each specific part of the state space, a piecewise affine controller can be used [36]. This way, however, the computational demands rise rapidly with the number of speed regions $\mathcal{I}$. It might, therefore, be necessary to simply implement multiple controllers, for each given speed region i, and to switch between those based on the actual speed ω_m. With this method only the memory demands rise linearly with the number of speed regions $\mathcal{I}$. However, the predictions of the MPC controller cannot take into account the different dynamics in the different speed regions, which are other than that the controller was designed for. Since the prediction horizon is going to be chosen short for such a fast system, the resulting error is expected to be insignificant.

To further reduce the inaccuracy in our model caused by the difference between the actual speed and the currently used speed constant Ω_{mi}, the compensation voltage $\boldsymbol{u}_{\mathrm{sdq}}^{\mathrm{cmp}} = \left[u_{\mathrm{sd}}^{\mathrm{cmp}}, \quad u_{\mathrm{sq}}^{\mathrm{cmp}}\right]^{\mathrm{T}}$ is added to the stator voltage calculated by the controller $\boldsymbol{u}_{\mathrm{sdq}}^{\mathrm{ctrl}}$. The actual voltage applied to the stator of the machine is then going to be

$$\begin{aligned} u_{\mathrm{sd}}(k) &= -LP_p\left(\omega_m(k) - \Omega_{\mathrm{mi}}\right) i_{\mathrm{sq}}(k) + u_{\mathrm{sd}}^{\mathrm{ctrl}}(k) \\ &= u_{\mathrm{sd}}^{\mathrm{cmp}}(k) + u_{\mathrm{sd}}^{\mathrm{ctrl}}(k) \qquad (27) \\ u_{\mathrm{sq}}(k) &= LP_p\left(\omega_m(k) - \Omega_{\mathrm{mi}}\right) i_{\mathrm{sd}}(k) + u_{\mathrm{sq}}^{\mathrm{ctrl}}(k) \\ &= u_{\mathrm{sq}}^{\mathrm{cmp}}(k) + u_{\mathrm{sq}}^{\mathrm{ctrl}}(k). \qquad (28) \end{aligned}$$

The compensation voltage $\boldsymbol{u}_{\mathrm{sdq}}^{\mathrm{cmp}}$ was obtained as the result of the subtraction of the proposed model equations (25) and (26) from the ideal current model (15) and (16). Adding the compensation voltage to the controller output removes the inaccuracy of the proposed model (25) and (26), caused by inequality of the actual speed ω_m and the speed constant Ω_{mi} for the current sample k. However, since the compensation voltage is added outside the controller, it cannot help to the correct predictions made by the controller.

Since the stator voltage calculated by the controller $\boldsymbol{u}_{\mathrm{sdq}}^{\mathrm{ctrl}}$ is no longer equal to the actual stator voltage $\boldsymbol{u}_{\mathrm{sdq}}$, the controller cannot keep the voltage within its constraints. The solution to this problem is to include the compensation voltage $\boldsymbol{u}_{\mathrm{sdq}}^{\mathrm{cmp}}$ into the model, in the form of a measured disturbance, constant over the prediction horizon. Voltage constraints can then be implemented over the sum of the compensation and the controller calculated voltage. In other words, the stator voltage constraints will approximate the circle given by



u_{sq}
$u_{\mathrm{sq}}^{\mathrm{ctrl}}$
$u_{\mathrm{sq}}^{\mathrm{cmp}}$
$u_{\mathrm{sd}}^{\mathrm{ctrl}}$
u_{sd}
U_{max}
$u_{\mathrm{sd}}^{\mathrm{cmp}}$

Fig. 1. Voltage constraints, including the compensation voltage.

$$U_{\mathrm{max}} \geq \sqrt{\left(u_{\mathrm{sd}}^{\mathrm{cmp}}(k) + u_{\mathrm{sd}}^{\mathrm{ctrl}}(k)\right)^2 + \left(u_{\mathrm{sq}}^{\mathrm{cmp}}(k) + u_{\mathrm{sq}}^{\mathrm{ctrl}}(k)\right)^2} \qquad (29)$$

where U_{max} is the maximal stator voltage, that can be generated using the space vector modulation (SVM) technique. For example, the hexagon-shaped voltage inscribed to the circle in (29), as shown in Fig. 1, will be expressed as (30) and (31)–(33) as shown at bottom of the next page

$$\begin{bmatrix} -1 & \frac{-1}{\sqrt{3}} & -1 & \frac{-1}{\sqrt{3}} \\ -1 & \frac{1}{\sqrt{3}} & -1 & \frac{1}{\sqrt{3}} \\ 0 & \frac{2}{\sqrt{3}} & 0 & \frac{2}{\sqrt{3}} \\ 0 & \frac{-2}{\sqrt{3}} & 0 & \frac{-2}{\sqrt{3}} \\ 1 & \frac{-1}{\sqrt{3}} & 1 & \frac{-1}{\sqrt{3}} \\ 1 & \frac{1}{\sqrt{3}} & 1 & \frac{1}{\sqrt{3}} \end{bmatrix} \begin{bmatrix} u_{\mathrm{sd}}^{\mathrm{cmp}}(k) \\ u_{\mathrm{sq}}^{\mathrm{cmp}}(k) \\ u_{\mathrm{sd}}^{\mathrm{ctrl}}(k-1) \\ u_{\mathrm{sq}}^{\mathrm{ctrl}}(k-1) \end{bmatrix} \leq \begin{bmatrix} U_{\mathrm{max}} \\ U_{\mathrm{max}} \\ U_{\mathrm{max}} \\ U_{\mathrm{max}} \\ U_{\mathrm{max}} \\ U_{\mathrm{max}} \end{bmatrix}. \qquad (30)$$

The final mathematical model incorporating the proposed linearization method leads to a system (7) with the state vector (31), input vector (32), and state and input matrix (33). Both the state weighting matrix $\boldsymbol{Q}$ and the matrix $\boldsymbol{R}$ will remain the same as (23) and (24). Similarly to the voltage constraints, the current constraints can be implemented as a hexagon inscribed to the circle with radius I_{max}, according to equation

$$\begin{bmatrix} \frac{-1}{\sqrt{3}} & \frac{1}{\sqrt{3}} & \frac{2}{\sqrt{3}} & \frac{-2}{\sqrt{3}} & \frac{-1}{\sqrt{3}} & \frac{1}{\sqrt{3}} \\ -1 & -1 & 0 & 0 & 1 & 1 \end{bmatrix}^{\mathrm{T}} \begin{bmatrix} i_{\mathrm{sd}}(k) \\ i_{\mathrm{sq}}(k) \end{bmatrix} \leq \begin{bmatrix} I_{\mathrm{max}} \\ \vdots \\ I_{\mathrm{max}} \end{bmatrix}. \qquad (34)$$

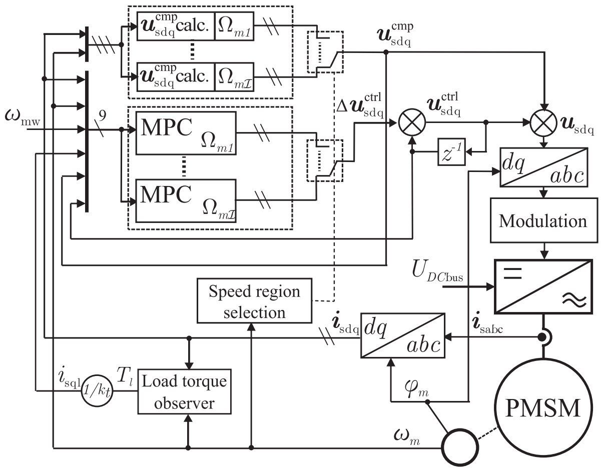


Fig. 2. Block diagram of the proposed control algorithm.

The explicit MPC design is applied to this system as described in Section II-C. A block diagram of the proposed controller, realized using multiple MPC controllers, is shown in Fig. 2.

III. Implementation and Testing

The proposed MPC algorithm has been verified in two steps. First, the MPC controller parameters that would assure the system stability and good performance, were found using MATLAB/Simulink 2014b 64-bit and the MPT version 3.0.20. GUROBI version 6.0.0 was used as a solver for the quadratic programming optimization problem. The standard FOC method, including a field-weakening algorithm with a PI controller, was simulated as well for comparison. The second step was implementation on a National Instruments cRIO system to prove the performance during the control of a real PMSM drive. The load torque observer was implemented in the form of a sliding mode estimator [35]. The parameters of PMSM are listed in Table I.

TABLE I
Parameters of PMSM

Parameter	Value	Unit
R_s	0.38	Ω
$L = L_d = L_q$	535	μH
Ψ_{PM}	0.02594	Wb
P_p	3	-
$U_{\max}$	8.6	V
$I_{\max}$	2	A
$\omega_{m\ \mathrm{nom}}$	286	rad/s
J	65.10^{-6}	kg·m^2

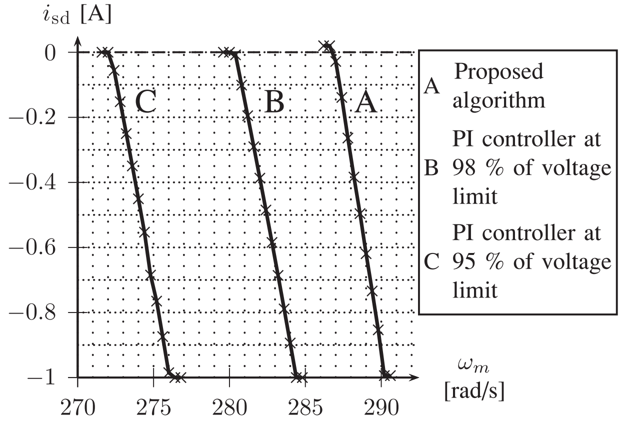


Fig. 3. Comparison of i_{sd} during the field-weakening operation.

A. Results of Simulations

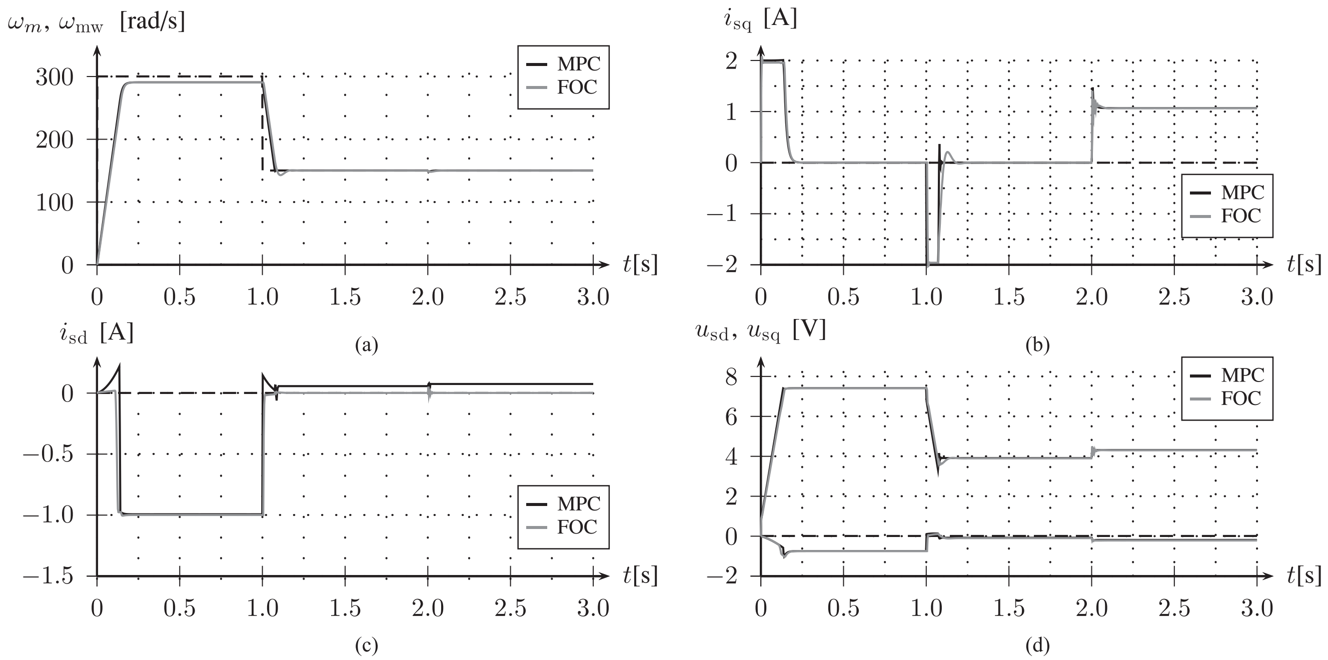
The parameters for the design of the MPC controller were experimentally set as

$$\begin{array}{llllll} T_s & = & 300\mu\mathrm{s}, & q_{i_q} & = & 0.5 \\ n_p & = & 4, & q_\omega & = & 1000 \\ n_u & = & 4, & r_{\Delta u_d} & = & 100 \\ q_{i_d} & = & 0.2, & r_{\Delta u_q} & = & 100. \end{array}$$

$$\boldsymbol{x}(k) = \left[i_{\mathrm{sd}}(k),\ i_{\mathrm{sq}}(k),\ \omega_m(k),\ u_{\mathrm{sd}}^{\mathrm{cmp}}(k),\ u_{\mathrm{sq}}^{\mathrm{cmp}}(k),\ u_{\mathrm{sd}}^{\mathrm{ctrl}}(k-1),\ u_{\mathrm{sq}}^{\mathrm{ctrl}}(k-1),\ \omega_{\mathrm{mw}}(k),\ i_{\mathrm{sql}}(k)\right]^{\mathrm{T}} \tag{31}$$

$$\boldsymbol{u}(k) = \left[\Delta u_{\mathrm{sd}}^{\mathrm{ctrl}}(k),\ \Delta u_{\mathrm{sq}}^{\mathrm{ctrl}}(k)\right]^{\mathrm{T}} \tag{32}$$

$$\underbrace{\boldsymbol{A}_i = \begin{bmatrix} 1-T_s\dfrac{R_s}{L} & T_sP_p\Omega_{m_i} & 0 & 0 & 0 & \dfrac{T_s}{L} & 0 & 0 & 0 \\ -T_sP_p\Omega_{m_i} & 1-T_s\dfrac{R_s}{L} & -T_sP_p\dfrac{\Psi_{\mathrm{PM}}}{L} & 0 & 0 & 0 & \dfrac{T_s}{L} & 0 & 0 \\ 0 & \dfrac{T_sk_t}{J} & 1 & 0 & 0 & 0 & 0 & 0 & -\dfrac{T_sk_t}{J} \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}}_{\forall\omega_m\in\langle\omega_{\mathrm{m}\,i},\ \omega_{\mathrm{m}\,i+1})},\quad \boldsymbol{B} = \begin{bmatrix} \dfrac{T_s}{L} & 0 \\ 0 & \dfrac{T_s}{L} \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \tag{33}$$



(a) (b) (c) (d)

Fig. 4. Simulation comparison of FOC with a PI controller-based field weakening and the proposed MPC algorithm.

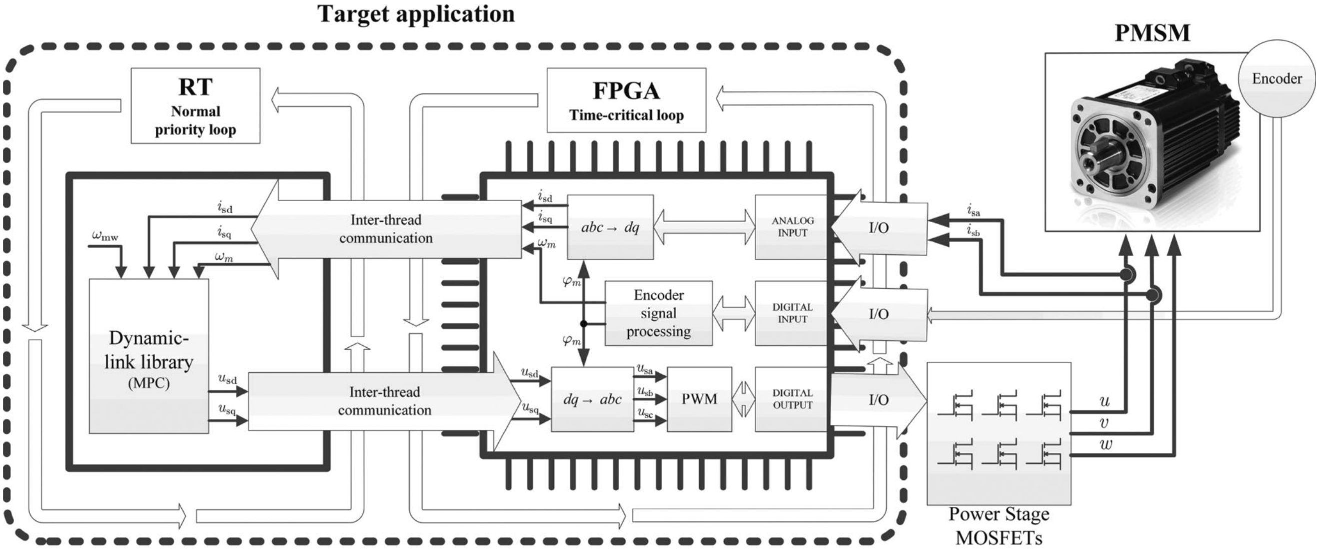


Fig. 5. Target application.

A total of two MPC controllers were implemented with the speed constants selected as $\Omega_{m1} = 200$ rad/s and $\Omega_{m2} = -200$ rad/s. A zero speed was selected as the border for selection between these two controllers.

Voltage constraints were implemented as a hexagon inscribed to a circle with radius $U_{\max}$. The current constraints were modified so that $|i_{\mathrm{sd}}| \leq 1$ A. All system state and input variables were normalized to optimize the number of regions of the explicit MPC controller.

In the simulation, the FOC with a field-weakening algorithm was compared with the proposed MPC algorithm. The results are shown in Fig. 4(a)–4(d). The field-weakening capability of both algorithms was shown when the required speed ω_{mw} is first set to an unreachable value of 300 rad/s. The load torque $T_l = 0.125$ Nm was introduced after 2 s. Thanks to inclusion of the load torque information into the machine model for the MPC, both algorithms show an equal performance under load. The moment of inertia was set to $J = 443 \times 10^{-6}$ kg·m^2 in all the simulations.

The field-weakening algorithm with a PI controller is widely used in practical applications [38]. The drawback of this algorithm (as with most field-weakening algorithms) is that the field weakening must start before the actual voltage limitation is met. The i_{sd} is then unnecessarily large. This is, however, not the case

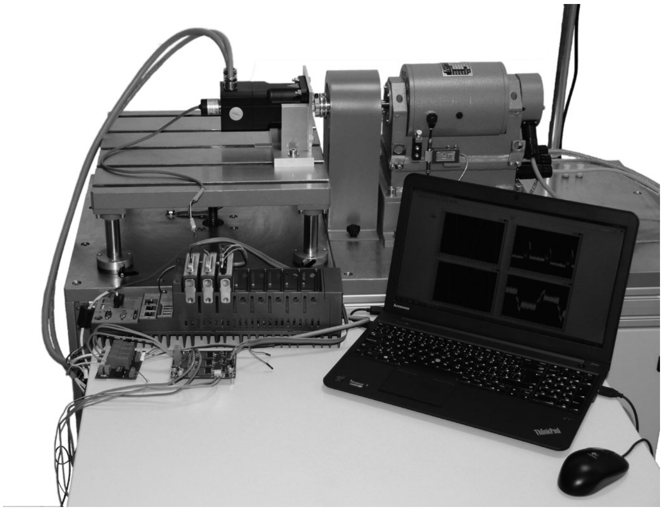


Fig. 6. Test bench.

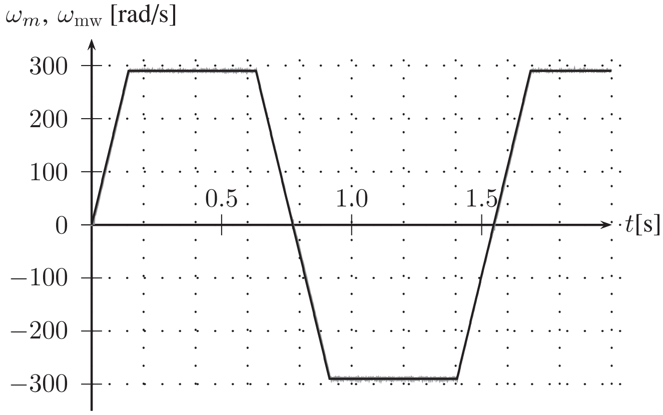


Fig. 7. Speed response—speed reversal (ω_m—gray; ω_{mw}—black).

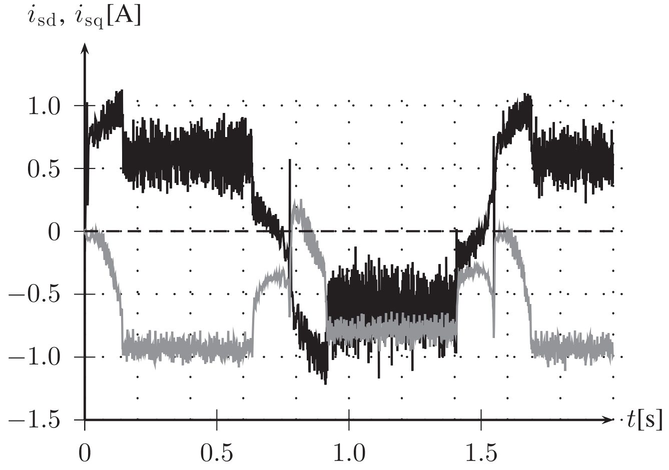


Fig. 8. Stator current—speed reversal (i_{sd}—gray; i_{sq}—black).

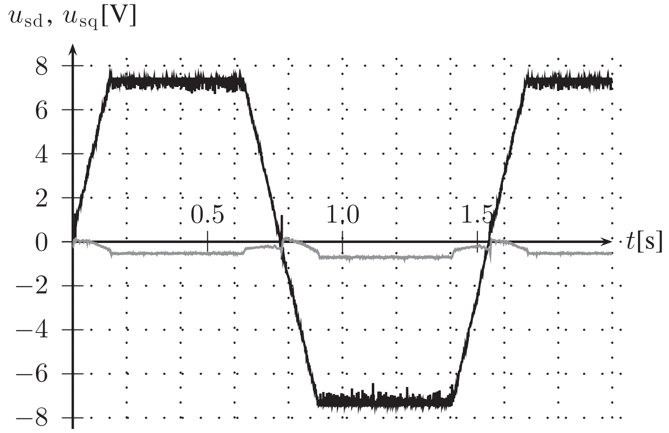


Fig. 9. Stator voltage—speed reversal (u_{sd}—gray; u_{sq}—black).

of the proposed MPC algorithm, because the field-weakening ability is a natural consequence of the machine model and control strategy. The simulation of this situation is shown in Fig. 3, where the proposed MPC algorithm (A) utilizes the smallest i_{sd} current to reach desired speed compared to the field-weakening method with a PI controller (B and C). In other words, the field weakening with proposed algorithm (A) starts at speed higher by 6 rad/s, respectively, 15 rad/s, than in the case of the PI controller algorithm (B and C). Please note, that since the algorithm was proposed for an SPMSM and the i_{sd} does not produce torque, the size of i_{sd} is the only criterion evaluated here. It can be seen in the Fig. 4(c) that the current i_{sd} is not kept exactly at the zero level during normal operation with no field weakening. This is caused by the value of q_{i_d} constant, which must be small enough to allow field weakening, which means, that the i_{sd} plays smaller role in the cost function. The inaccuracy of predictions based on model (25) and (26) contribute to this issue as well.

B. Experimental Results

The target NI cRIO-9082 platform contains a deterministic RT controller and an FPGA, the Spartan-6 LX150. The RT controller is based on a 1.33-GHz dual-core i7 processor running the RT operating system (OS) Phar Lap 13.1. The MPC algorithm was implemented using C-code and it was compiled in LabWindows/CVI as a Dynamic-link library (DLL). The DLL was called within the RT OS. The MPC controller for one speed constant Ω_{mi} contained 527 regions and generated a DLL of size 846 kB.

The architecture of the target application is shown in Fig. 5. The target application was divided into a time-critical task and a normal priority task. Better determinism is reached in the FPGA; therefore, the parts of the program with time-critical priority are executed in the FPGA. This time-critical loop contains the encoder signal processing, generation of the pulsewidth modulation, dead-time compensation, communication with the RT, and communication with the I/O modules. It was not possible to implement the MPC algorithm in the FPGA, thanks to the limited memory capacity. On the other hand, there was an effort to minimize the computation time of the MPC algorithm. Therefore, the FPGA also executes the calculation of the Clarke's and Park's transformations, the inverse Park's transformation and the SVM.

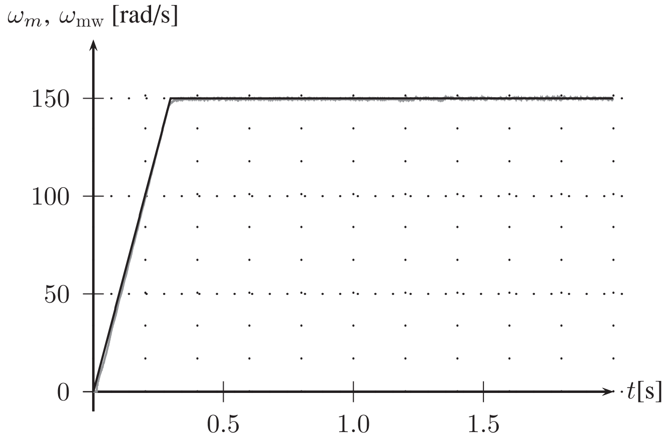


Fig. 10. Speed response—response to the load torque (ω_m—gray; $\omega_{\rm mw}$—black).

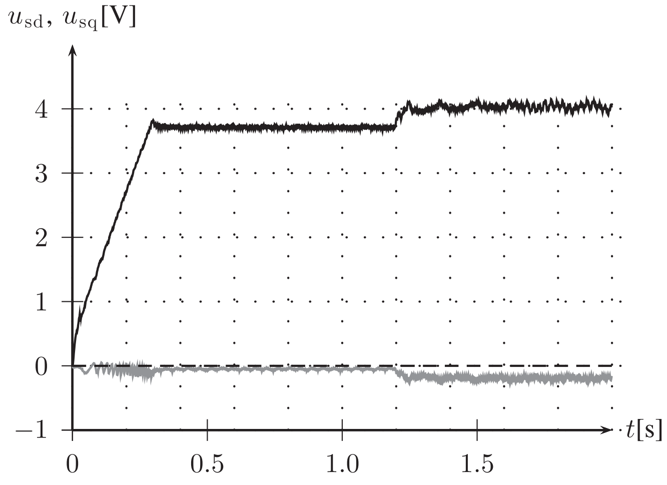


Fig. 12. Stator voltage—response to the load torque ($u_{\rm sd}$—gray; $u_{\rm sq}$—black).

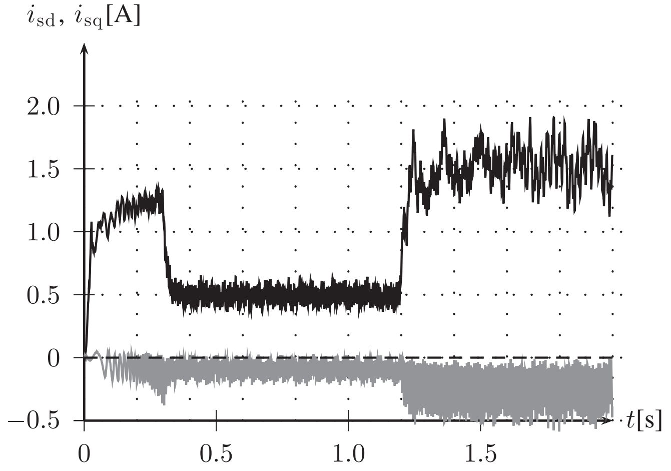


Fig. 11. Stator current—response to the load torque ($i_{\rm sd}$—gray; $i_{\rm sq}$—black).

Two experiments were performed to test the proposed algorithm. The first experiment was focused on proving the performance at a high angular speed (over the nominal speed of the PMSM) and demonstrating the field-weakening operation. A trapezoid with a maximum 290 rad/s and minimum −290 rad/s was used as the reference speed signal. The initial speed reference was set to zero. The speed response is shown in Fig. 7, the stator currents can be seen in Fig. 8 and the stator voltage is presented in Fig. 9. The noise in the stator current response was caused by the noise in current measurement and inaccuracies in the machine model (nonmodeled dynamics and nonlinearities) as well as by relatively high value of penalty q_ω, and therefore, high gain of speed control. It was found that high value of q_ω is necessary for proper field-weakening operation and for speed control error reduction. The stator current $i_{\rm sq}$ reached the positive value immediately after the start of the experiment, and influenced the magnitude of the generated torque. The voltage limitation was reached near the nominal speed. Then, the stator current $i_{\rm sd}$ decreased to a negative value, and thus, reduced the back EMF. As the desired speed of 290 rad/s was not reachable through a normal motor operation, field weakening had to be performed.

In the second experiment, a dynamometer was connected to the PMSM as a load (see Fig. 6) and the response to the load torque change was tested. The total moment of inertia was increased to $J = 443 \times 10^{-6}$kg·m^2. The speed control error penalty was changed to $q_\omega = 5000$ for a faster angular speed response. The speed reference was set to a constant value of 150 rad/s during the whole experiment and the initial speed reference was set to zero. Fig. 10 shows the speed response, the stator currents are presented in Fig. 11, and the stator voltages are in Fig. 12. At time 1.2 s, the step change of the load torque to 0.125 Nm was introduced. The steepness of the increase in the load torque was set to 2 Nm/s. The stator current $i_{\rm sq}$ increased to compensate the load torque.

IV. Conclusion

The linear MPC possesses several beneficial properties that classical motor control strategies do not offer. It is especially the ease of the control strategy definition and the ability to deal with complex constraints of system inputs, outputs, and states in simple manner. Since the explicit MPC allowed to reduce computation demands, the linear MPC became viable option for motor control as well. The classic approach to MPC of PMSM described in Section II-C does not allow field weakening, although it is natural behavior resulting from the system model and defined control strategy. The proposed algorithm employs different model linearization method to remove this limitation.

The results of simulations shown in Section III-A show that the performance of the proposed MPC algorithm is better or at least comparable to FOC.

The results of the experiments have proved that the proposed MPC algorithm is able to control a real PMSM drive under laboratory conditions. The first experiment proved the field-weakening operation capability of the proposed algorithm. The second experiment verified the stator current constraints. The actual stator voltage and current constraints are coupled through the stator resistance and accuracy of the SVM. The accuracy of the SVM is usually constant, but not the stator resistance. This dependence might cause problems when the motor winding temperature rises and the stator resistance is different. The main disadvantage of this approach is the high computational complexity and the memory requirements of the MPC algorithm.

A major advantage is the possibility of a natural definition of the control objectives, as well as proper constraints handling, allowing automatic realization of the field weakening necessary to achieve a higher speed. The significant advantage is also that the field weakening actually occurs with a lower *d*-axis current than is possible with the popular field-weakening algorithm based on a PI controller. It was demonstrated that implementation of a predictive control of a PMSM is feasible in a laboratory environment using hardware that is close to real life industrial applications. The research will continue to implement the proposed approach using industrial motor control processors and for interior PMSMs as well.

References

[1] F. Zhang, Z. Meng, and J. Chen. Study on control technology of double rotor PMSM in underwater vehicle. in *Proc. Int. Conf. Applied Superconduct. Electromagnet. Devices*, Sep. 2009, pp. 295–298. [Online]. Available: http://dx.doi.org/10.1109/ASEMD.2009.5306636, doi: 10.1109/ASEMD.2009.5306636.

[2] G. Sree Lakshmi, S. Kamakshaiah, and T. R. Das. Closed loop PI control of PMSM for hybrid electric vehicle using three level diode clamped inverter for optimal efficiency. in *Proc. Int. Conf. Energy Efficient Technol. Sustain.*, Apr. 2013, pp. 754–759. [Online]. Available: http://dx.doi.org/10.1109/ICEETS.2013.6533479, doi: 10.1109/ICEETS.2013.6533479.

[3] S. Yu, Z. Yang, S. Liu, and K. Zheng. Analysis and implementation of digitalized vector control for PMSM with switching control. in *Proc. 3rd Int. Symp. Systems Control Aeronaut. Astronaut.*, Jun. 2010, pp. 912–917. [Online]. Available: http://dx.doi.org/10.1109/ISSCAA.2010.5633049, doi: 10.1109/ISSCAA.2010.5633049.

[4] T. Sun, C. Liu, N. Lu, D. Gao, and S. Xu. Design of PMSM vector control system based on TMS320F2812 DSP. in *Proc. 7th Int. Power Electronc. Motion Control Conf.*, Jun. 2012, pp. 2602–2606. [Online]. Available: http://dx.doi.org/10.1109/IPEMC.2012.6259270, doi: 10.1109/IPEMC.2012.6259270.

[5] F. Niu, B. Wang, A. Babel, K. Li, and E. Strangas, "Comparative evaluation of direct torque control strategies for permanent magnet synchronous machines," *IEEE Trans. Power Electron.*, vol. 31, no. 2, pp. 1408 –1424, Feb. 2016. [Online]. Available: http://dx.doi.org/10.1109/TPEL.2015.2421321, doi: 10.1109/TPEL.2015.2421321.

[6] M. A. Perez, J. Rodriguez, E. J. Fuentes, and F. Kammerer, "Predictive control of AC-AC modular multilevel converters," *IEEE Trans. Ind. Electron.*, vol. 59, no. 7, pp. 2832–2839, Jul. 2012. [Online]. Available: http://dx.doi.org/10.1109/TIE.2011.2159349, doi: 10.1109/TIE.2011.2159349.

[7] P. Cortes, J. Rodriguez, C. Silva, and A. Flores, "Delay compensation in model predictive current control of a three-phase inverter," *IEEE Trans. Ind. Electron.*, vol. 59, no. 2, pp. 1323–1325, Feb. 2012. [Online]. Available: http://dx.doi.org/10.1109/TIE.2011.2157284, doi: 10.1109/TIE.2011.2157284.

[8] J. D. Barros, J. F. A. Silva, and E. G. A. Jesus, "Fast-Predictive optimal control of NPC multilevel converters," *IEEE Trans. Ind. Electron.*, vol. 60, no. 2, pp. 619–627, Feb. 2013. [Online]. Available: http://dx.doi.org/10.1109/TIE.2012.2206352, doi: 10.1109/TIE.2012.2206352.

[9] C. A. Rojas, J. Rodriguez, F. Villarroel, 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. [Online]. Available: http://dx.doi.org/10.1109/TIE.2012.2206344, doi: 10.1109/TIE.2012.2206344.

[10] K. Drobnic, M. Nemec, D. Nedeljkovic, and V. Ambrozic, "Predictive direct control applied to AC drives and active power filter," *IEEE Trans. Ind. Electron.*, vol. 56, no. 6, pp. 1884–1893, Jun. 2009. [Online]. Available: http://dx.doi.org/10.1109/TIE.2009.2015749, doi: 10.1109/TIE.2009.2015749.

[11] H. Miranda, P. Cortes, J. Yuz, and J. Rodriguez, "Predictive torque control of induction machines based on state-space models," *IEEE Trans. Ind. Electron.*, vol. 56, no. 6, pp. 1916–1924, Jun. 2009. [Online]. Available: http://dx.doi.org/10.1109/TIE.2009.2014904, doi: 10.1109/TIE.2009.2014904.

[12] C. Xia, Y. Wang, and T. Shi, "Implementation of Finite-State model predictive control for commutation torque ripple Minimization of permanent-magnet brushless DC motor," *IEEE Trans. Ind. Electron.*, vol. 60, no. 3, pp. 896–905, Mar. 2013. [Online]. Available: http://dx.doi.org/10.1109TIE.2012.2189536, doi: 10.1109/TIE.2012.2189536.

[13] S. Richter, S. Mariethoz, and M. Morari, "High-speed online MPC based on a fast gradient method applied to power converter control," in *Proc. Amer. Control Conf.*, 2010, pp. 4737–4743.

[14] S. Mariethoz, A. Domahidi, and M. Morari. Sensorless explicit model predictive control of permanent magnet synchronous motors. in *Proc. IEEE Int. Elect. Machine. Drives Conf.*, May. 2009, pp. 1250–1257. [Online]. Available: http://dx.doi.org/10.1109/IEMDC.2009.5075363, doi: 10.1109/IEMDC.2009.5075363.

[15] A. Bemporad, M. Morari, V. Dua, and E. Pistikopoulos. The explicit solution of model predictive control via multiparametric quadratic programming. in *Proc. American Control Conf.*, vol. 2, 2000, pp. 872–876. [Online]. Available: http://dx.doi.org/10.1109/ACC.2000.876624, doi: 10.1109/ACC.2000.876624.

[16] S. Bolognani, S. Bolognani, L. Peretti, and M. Zigliotto, "Design and implementation of model predictive control for electrical motor drives," *IEEE Trans. Ind. Electron.*, vol. 56, no. 6, pp. 1925–1936, Jun. 2009. [Online]. Available: http://dx.doi.org/10.1109/TIE.2008.2007547, doi: 10.1109/TIE.2008.2007547.

[17] S. Kouro, P. Cortes, R. Vargas, U. Ammann, and J. Rodriguez, "Model predictive control? A simple and powerful method to control power converters," *IEEE Trans. Ind. Electron.*, vol. 56, no. 6, pp. 1826–1838, Jun. 2009. [Online]. Available: http://dx.doi.org/10.1109/TIE.2008.2008349, doi: 10.1109/TIE.2008.2008349.

[18] C.-K. Lin, J.-t. Yu, L.-C. Fu, T.-H. Liu, and C.-F. Hsiao. An improved predictive current control for interior permanent magnet synchronous motor drives based on current difference detection. in *Proc. IEEE/ASME Int. Conf. Advanced Intelligent Mechatron.*, 2012, pp. 988–993. [Online]. Available: http://dx.doi.org/10.1109/AIM.2012.6265870, doi: 10.1109/AIM.2012.6265870.

[19] C.-C. Hua, C.-W. Wu, and C.-W. Chuang, "A digital predictive current control with improved sampled inductor current for cascaded inverters," *IEEE Trans. Ind. Electron.*, vol. 56, no. 5, pp. 1718–1726, May. 2009. [Online]. Available: http://dx.doi.org/10.1109/TIE.2009.2012415, doi: 10.1109/TIE.2009.2012415.

[20] J. Guzinski and H. Abu-Rub, "Speed sensorless induction motor drive with predictive current controller," *IEEE Trans. Ind. Electron.*, vol. 60, no. 2, pp. 699–709, Feb. 2013. [Online]. Available: http://dx.doi.org/10.1109/TIE.2012.2205359, doi: 10.1109/TIE.2012.2205359.

[21] T. Turker, U. Buyukkeles, and A. Bakan, "A robust predictive current controller for PMSM drives," *IEEE Trans. Ind. Electron.* [Online]. Available: http://dx.doi.org/10.1109/TIE.2016.2521338, doi: 10.1109/TIE.2016.2521338.

[22] 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, Sep. 2015. [Online]. Available: http://dx.doi.org/10.1109/TIE.2015.2410767, doi: 10.1109/TIE.2015.2410767.

[23] P. Vaclavek and P. Blaha, "Enhanced discrete time model for AC induction machine model predictive control," in *Proc. 38th Annu. Conf. IEEE Indust. Electron. Soc.*, Oct. 2012, pp. 5043–5048. [Online]. Available: http://dx.doi.org/10.1109/IECON.2012.6389564, doi: 10.1109/IECON.2012.6389564.

[24] A. Formentini, A. Trentin, M. Marchesoni, P. Zanchetta, and P. Wheeler, "Speed finite control set model predictive control of a PMSM fed by matrix converter," *IEEE Trans. Ind. Electron.*, vol. 62, no. 11, pp. 6786–6796, Nov. 2015. [Online]. Available: http://dx.doi.org/10.1109/TIE.2015.2442526, doi: 10.1109/TIE.2015.2442526.

[25] Y. Zhang, L. Xu, M. K. Gven, S. Chi, and M. Illindala, "Experimental verification of deep field weakening operation of a 50-kW IPM machine by using single current regulator," *IEEE Trans. Ind. Appl.*, vol. 47, no. 1, pp. 128–133, Jan./Feb. 2011. [Online]. Available: http://dx.doi.org/10.1109/TIA.2010.2091478, doi: 10.1109/TIA.2010.2091478.

[26] P. Vaclavek and P. Blaha, "Interior permanent magnet synchronous machine field weakening control Strategy—The analytical solution," in *Proc. SICE Annu. Conf.*, 2008, pp. 753–757.

[27] M. Preindl and S. Bolognani, "Model predictive direct torque control with finite control set for PMSM drive systems, Part 2: Field weakening operation," *IEEE Trans. Ind. Informat.*, vol. 9, no. 2, pp. 648–657, May. 2013. [Online]. Available: http://dx.doi.org/10.1109/TII.2012.2220353, doi: 10.1109/TII.2012.2220353.

[28] P. Vaclavek and P. Blaha. PMSM model discretization for model predictive control algorithms. in *Proc. IEEE/SICE Int. Symp. System Integr.*, Dec. 2013, pp. 13–18. [Online]. Available: http://dx.doi.org/10.1109/SII.2013.6776649, doi: 10.1109/SII.2013.6776649.

[29] Z. Ma, S. Saeidi, and R. Kennel, "FPGA implementation of model predictive control with constant switching frequency for PMSM drives," *IEEE Trans. Ind. Informat.*, vol. 10, no. 4, pp. 2055–2063, Nov. 2014. [Online]. Available: http://dx.doi.org/10.1109/TII.2014.2344432, doi: 10.1109/TII.2014.2344432.

[30] T. Nguyen-Van and N. Hori, "Discretization of nonautonomous nonlinear systems based on continualization of an exact discrete-time model," *J. Dynamic Syst., Meas. Control*, vol. 136, no. 2, p. 021004, Nov. 2013. [Online]. Available: http://dx.doi.org/10.1115/1.4025711, doi: 10.1115/1.4025711.

[31] M. Kvasnica, *Real-Time Model Predictive Control via Multi-Parametric Programming: Theory and Tools*. VDM Verlag, 2009.

[32] A. Bemporad, M. Morari, V. Dua, and E. Pistikopoulos, "The explicit linear quadratic regulator for constrained systems," *Automatica*, vol. 38, no. 1, pp. 3–20, Jan. 2002. [Online]. Available: http://dx.doi.org/10.1016/S0005-1098(01)00174-1, doi: 10.1016/S0005-1098(01)00174-1.

[33] R. Errouissi, M. Ouhrouche, W.-h. Chen, and A. M. Trzynadlowski, "Robust nonlinear predictive controller for permanent-magnet synchronous motors with an optimized cost function," *IEEE Trans. Ind. Electron.*, vol. 59, no. 7, pp. 2849–2858, Jul. 2012. [Online]. Available: http://dx.doi.org/10.1109/TIE.2011.2157276, doi: 10.1109/TIE.2011.2157276.

[34] P. Vaclavek, P. Blaha, and I. Herman, "AC drive observability analysis," *IEEE Trans. Ind. Electron.*, vol. 60, no. 8, pp. 3047–3059, Aug. 2013. [Online]. Available: http://dx.doi.org/10.1109/TIE.2012.2203775, doi: 10.1109/TIE.2012.2203775.

[35] C. Zhang, L. Jia, and H. Jing. Load torque observer based sliding mode control method for permanent magnet synchronous motor. in *Proc. 25th Chinese Control Decision Conf.*, pp. 550–555, 2013. [Online]. Available: http://dx.doi.org/978-1-4673-5534-6/13, doi: 978-1-4673-5534-6/13.

[36] M. Kvasnica, P. Grieder, M. Baotic, and M. Morari, "Multi-parametric toolbox (MPT)," in *Hybrid Systems: Computation and Control* (Lecture Notes in Computer Science), R. Alur and G. J. Pappas, Eds. Berlin, Germany: Springer, 2004, vol. 2993, pp. 121–124.

[37] P. Vaclavek and P. Blaha. Field weakening implementation in AC induction machine predictive control. in *Proc. IEEE 9th Int. Conf. Power Electron. Drive Sys.*, Dec. 2011, pp. 171–176. [Online]. Available: http://dx.doi.org/10.1109/PEDS.2011.6147242, doi: 10.1109/PEDS.2011.6147242.

[38] R. Nalepa and T. Orlowska-Kowalska, "Optimum trajectory control of the current vector of a nonsalient-pole PMSM in the field-weakening region," *IEEE Trans. Ind. Electron.*, vol. 59, no. 7, pp. 2867–2876, Jul. 2012. [online]. Available: http://dx.doi.org/10.1109/TIE.2011.2116755, doi: 10.1109/TIE.2011.2116755.

Zbynek Mynar received the M.Sc. degree in cybernetics from Brno University of Technology, Brno, Czech Republic, in 2014, where he is currently working toward the Ph.D. degree.

His research interests include sensorless control of synchronous reluctance motors and predictive control of synchronous motors. He is also currently a System Application Engineer with NXP Semiconductor N.V., Roznov pod Radhostem, Czech Republic.

Libor Vesely received the M.Sc. and Ph.D. degrees in cybernetics from Brno University of Technology, Brno, Czech Republic, in 2006 and 2012, respectively.

His research interests include speed-sensorless control of permanent-magnet synchronous motors, state estimation, system modeling, and parameters estimation. He is currently a Junior Researcher with the Central European Institute of Technology, Brno University of Technology.

Pavel Vaclavek (M'04–SM'12) received the M.Sc. and Ph.D. degrees in cybernetics from Brno University of Technology, Brno, Czech Republic, in 1993 and 2001, respectively, where he also received the M.Sc. degree in industrial management in 1998.

His research interests include speed-sensorless control of induction machines, system modeling, and parameters estimation. He is a Research Group Leader with the Central European Institute of Technology, Brno University of Technology.