Koopman Operator Based Finite-Set Model Predictive Control for Electrical Drives

Sören Hanke*, Sebastian Peitz**, Oliver Wallscheid*, Stefan Klus***, Joachim Böcker*, and Michael Dellnitz**

*Department of Power Electronics and Electrical Drives, Paderborn University, Germany
**Department of Mathematics, Paderborn University, Germany
***Department of Mathematics and Computer Science, Freie Universität Berlin, Germany

Abstract

Predictive control of power electronic systems always requires a suitable model of the plant. Using typical physics-based white-box models, a trade-off between model complexity (i.e., accuracy) and computational burden has to be made. This is a challenging task since in general, the model order is directly linked to the number of system states. Even though white-box models show suitable performance in most cases, parasitic real-world effects often cannot be modeled satisfactorily with an expedient computational load. Hence, a Koopman operator based model reduction technique is presented which directly links the control action to the system’s outputs in a black-box fashion. The Koopman operator is a linear but infinite-dimensional operator describing the dynamics of observables of autonomous dynamical systems which can be nicely applied to the switching principle of power electronic devices. Following this data-driven approach, the model order and the number of system states are decoupled which allows us to consider more complex systems. Extensive experimental tests with an automotive-type permanent magnet synchronous motor fed by an IGBT 2-level inverter prove the feasibility of the proposed modeling technique in a finite-set model predictive control application.

1 Introduction

Linear feedback control is the most frequently used control strategy employed in power electronic and drive applications because of its simplicity and well-known design rules. However, those applications require to address nonlinear influences due to state and control action constraints, parameter changes (e.g., due to iron saturation) or in some cases even completely nonlinear control plants (e.g., LLC resonant converter). Hence, when designing linear feedback control loops, e.g., by utilizing classic PID elements, the control engineer has to manually tune the control parameters after a first analytical design step to ensure stability and acceptable performance. In addition, nonlinear control elements like anti-windup reset measures have to be added to the control loops. Consequently, most linear feedback control approaches are transformed step-wise into hand-tailored nonlinear control systems during the design process requiring highly experienced and application-specific engineering knowledge.

In contrast, model predictive control (MPC) techniques inherently allow to address nonlinear model plants and manifold system constraints by suitable definition of an optimal control problem on a receding horizon. In general, MPC calculates the control output by minimizing a cost function that describes the desired system behavior. The cost function evaluates the model-based predicted system output with a reference trajectory. For each sampling instant, the MPC calculates a control action sequence that minimizes the cost function, but only the first component of this sequence is applied to the system.

MPC theory was intensively investigated throughout the last century and first applied to complex chemical processes where standard linear feedback control delivered only unsatisfactory results [1] leading to first publications during the 1970s and 1980s [2]. First chemical processes considered for MPC had vast dominant time constants in the range of minutes or even hours and therefore, sufficiently large time intervals were available to solve the optimization problems with the limited computational power of that time. Due to the increasing computational performance of digital signal processing units in the last three decades, MPC techniques became also feasible for electrical power systems with typical time constants in the range of milliseconds or even microseconds [3, 4, 5] and for complex systems such as autonomous vehicles [6, 7].

For power electronic applications one has to distinguish between finite-control-set (FCS-MPC) and continuous-control-set (CCS-MPC) approaches [8]. In the latter, the control is an element of a continuous space and the computed control signals have to be forwarded to a modulator (e.g., space-vector- or pulse-width-modulation) to receive the desired switching sequence which can then be applied to the semiconductor’s driver circuits. The main advantages of CCS-MPC are a guaranteed fixed switching frequency and the usage of long prediction horizons since the controller turnaround-time is decoupled from the modulation carrier frequency. Furthermore, CCS-MPC allows to measure the relevant system outputs by regular-sampling, leading to reduced analog-digital-conversion requirements compared to FCS-MPC. FCS-MPC directly computes the switching signal sequence suitable for driving power electronic devices. On the one hand, FCS-MPC gives additional degrees of freedom to the control problem, e.g., to find loss-optimal pulse patterns or to manipulate the frequency-spectrum behavior but, on the other hand, the
controller turnaround-time has to be significantly smaller compared to CCS-MPC due to the missing modulation step. Furthermore, the underlying optimization problem is of combinatorial nature. Consequently, the FCS-MPC optimization routines have to be very fast and efficient which inherently requires streamlined plant models with minimal computational complexity. To achieve good control performance, both high model accuracy and long prediction horizons (i.e., lightweight plant models) are required. As these two model characteristics are obviously opposing goals, a trade-off decision has to be made.

![Fig 1](image)

**Figure 1**: Field-oriented FCS-MPC topology for an IPMSM drive.

One possible way to circumvent this trade-off decision is by using black-box plant models. This decouples the model complexity from the number of system states. Following the goal of streamlining the plant model, the focus of this contribution is to apply a recently developed data-driven modeling approach based on the Koopman operator to the FCS-MPC problem as depicted in Fig. 1. Using this approach, dynamical systems are obtained directly for the system outputs. The interior-magnets permanent magnet synchronous motor (IPMSM) driven by a 2-level 3-phase voltage source inverter is used as a typical power electronics application for proving the general feasibility of the Koopman operator approach. For the sake of simplicity, only the current control loop is realized by an FCS-MPC approach in this first proof of concept while the reference values are calculated by a superimposed operation point strategy (OPS). The remainder of this paper is structured as follows: In Sec. 2, we will briefly present the FCS-MPC framework. The concept of the Koopman operator and its basic theoretical background will be given in Sec. 3. Extensive experimental tests are compared to CCS-MPC due to the missing modulation step. Furthermore, the underlying optimization problem is of combinatorial nature. Consequently, the FCS-MPC optimization routines have to be very fast and efficient which inherently requires streamlined plant models with minimal computational complexity. To achieve good control performance, both high model accuracy and long prediction horizons (i.e., lightweight plant models) are required. As these two model characteristics are obviously opposing goals, a trade-off decision has to be made.

Here, bold symbols denote vector and matrix quantities. For the drive system given in Fig. 1, the first-order Euler approximated discrete-time model in the rotor-flux oriented dq-system is given by:

\[
\begin{align*}
    \dot{i}_{dq,i+1} &= L_{dq}^{-1} \left[ Q(\Delta \varepsilon_{el,i}) (L_{dq} i_{dq,i} + T_s u_{dq,i}) - R_s T_s i_{dq,i} + (Q(\Delta \varepsilon_{el,i}) - I) \psi_p \right],
\end{align*}
\]

Above, \( x = i_{dq} = [i_d \ i_q]^\top \) is the stator current, \( T_s \) is the sampling time, \( R_s \) is the stator resistance, \( \psi_p = [\psi_{pm} \ 0]^\top \) is the permanent magnet flux linkage, \( I \) is the unity matrix and \( u_{dq} = [u_d \ u_q]^\top \) is the stator voltage, respectively. In addition, \( \Delta \varepsilon_{el,i} = \varepsilon_{el,i+1} - \varepsilon_{el,i} \) is the electrical rotor angle deviation between two sampling steps, \( Q \) is the rotation matrix

\[
Q(\varepsilon_{el}) = \begin{bmatrix} \cos(\varepsilon_{el}) & \sin(\varepsilon_{el}) \\ -\sin(\varepsilon_{el}) & \cos(\varepsilon_{el}) \end{bmatrix},
\]

and \( L_{dq} \) is the inductance matrix

\[
L_{dq} = \begin{bmatrix} L_d & 0 \\ 0 & L_q \end{bmatrix}.
\]

It should be noted that temperature influences, (cross-)saturation effects and iron losses are neglected and therefore, all motor parameters are considered constant. Linking (2) with (1), the control action \( u_i = [s_{a,i} \ s_{b,i} \ s_{c,i}]^\top \) is defined as:

\[
\begin{align*}
    u_{dq,i} &= Q(\varepsilon_{el,i}) \frac{2}{3} \begin{bmatrix} 1 & -\frac{1}{2} & -\frac{1}{2} \\ 0 & \frac{\sqrt{3}}{2} & -\frac{\sqrt{3}}{2} \end{bmatrix} \frac{u_{dc,i}}{2} u_i.
\end{align*}
\]

Here, \( s_{abc,i} = \{+1, -1\} \) are the switching commands for the inverter half-bridges and \( u_{dc,i} \) is the measured DC-link voltage.

Finally, the optimization task of the FCS-MPC for the given drive application is defined as

\[
\min_{u_i} J = \sum_{i=1}^{n_p} (u_{d,i} - u_{d,i})^2 + (u_{q,i} - u_{q,i})^2,
\]

with \( n_p \) being the number of prediction steps. To solve (4), an exhaustive search among all possible switching sequences from \( i = 1, \ldots, n_p \) will be performed. More efficient optimization algorithms, like branch-and-bound, can improve the MPC performance because more prediction steps can be performed with the same computational effort – however, the focus of this contribution is on the modeling part and therefore, the exhaustive search ensures comparability between different internal FCS-MPC models.

## 3 Koopman Operator Based Model Reduction

In the past decade, a new approach for the construction of reduced order models (ROMs) has emerged. It is based
on the Koopman operator \( \mathcal{K} \) which is a linear but infinite-dimensional operator describing the dynamics of observables of autonomous dynamical systems \([13,14]\). If we fix the input \( u_i = \bar{u} \) for \( i = 0,1,\ldots \), the system \([1]\) becomes an autonomous system:

\[
x_{i+1} = \Phi_{\bar{u}}(x_i), \quad i = 0,1,\ldots ,
\]

where the notation \( \Phi_{\bar{u}} \) indicates that a constant input \( \bar{u} \) is applied to system \([1]\).

We now introduce the real-valued observable \( f \) of the system. Then the Koopman operator \( \mathcal{K}_{\bar{u}} \) (corresponding to the constant control \( \bar{u} \)) describes the evolution of the observable \( y = f(x) \) and is defined by

\[
(\mathcal{K}_f(x)) = f(\Phi_{\bar{u}}(x)),
\]

which means that this way, we obtain a dynamical system for the observable \( f(x) \): Since the Koopman operator acts on observations of the dynamics, the computation is data based and we hence do not require knowledge of the underlying equations. Consequently, it can also be used to construct a surrogate model from sensor data.

\[\text{Dyn. Sys.} \quad f \quad \Phi_{\bar{u}} \quad \text{Koopman} \quad f \quad \Psi \quad \text{EDMD} \quad f \quad \Phi_{\bar{u}} \quad \text{Koopman} \quad P \quad \text{EDMD} \quad f \quad \Phi_{\bar{u}} \quad \text{Koopman} \quad f \]

\[\begin{array}{c}
x_i \\
\Phi_{\bar{u}} \\
x_{i+1}
\end{array} \quad \begin{array}{c}
f(x_i) \\
\mathcal{K}_{\bar{u}}(x_i) \\
f(x_{i+1})
\end{array} \quad \begin{array}{c}
\Psi \\
\mathcal{K}_{\bar{u}} \\
\Psi
\end{array} \quad \begin{array}{c}
z_i \\
K_{\bar{u}} \\
z_{i+1}
\end{array} \quad \begin{array}{c}
P \\
P
\end{array} \quad \begin{array}{c}
z_{i+1}
\end{array}
\]

Figure 2: Relation between the system dynamics \( \Phi_{\bar{u}} \), the corresponding Koopman operator \( \mathcal{K}_{\bar{u}} \) and its finite-dimensional representation \( \mathcal{K}_{\bar{u}} \) computed via EDMD.

The most popular approach to construct a finite-dimensional approximation of the Koopman operator is via Dynamic Mode Decomposition (DMD) \([15]\) or Extended Dynamic Mode Decomposition (EDMD) \([10]\).

While in DMD the Koopman operator is approximated for the observations themselves, in EDMD they are expressed in terms of arbitrary basis functions (e.g., monomials, Hermite polynomials or radial basis functions). For a given set of basis functions \( \{\psi_1,\psi_2,\ldots,\psi_k\} \) (the so-called dictionary), we define \( \psi \) by

\[
\psi(y) = [\psi_1(y) \ \psi_2(y) \ \ldots \ \psi_k(y)]^T ,
\]

where \( y = f(x) \). If \( \psi(y) = y \), we obtain DMD as a special case of EDMD. We assume that we have either measurement or simulation data, written in matrix form as

\[
Y = [y_1 \ y_2 \ \ldots \ y_m] \quad \text{and} \quad \tilde{Y} = [\tilde{y}_1 \ \tilde{y}_2 \ \ldots \ \tilde{y}_m]
\]

where \( \tilde{y}_i = f(\Phi_{\bar{u}}(x_i)) \). The data can be obtained from one long trajectory in which case we have \( \tilde{y}_i = y_{i+1} \). Alternatively, many short simulations may be assembled in

the matrices \( Y \) and \( \tilde{Y} \). For EDMD, the data matrices are embedded into the typically higher-dimensional feature space by

\[
\Psi_Y = [\psi(y_1) \ \psi(y_2) \ \ldots \ \psi(y_m)],
\]

\[
\Psi_{\tilde{Y}} = [\psi(\tilde{y}_1) \ \psi(\tilde{y}_2) \ \ldots \ \psi(\tilde{y}_m)].
\]

With these data matrices, we then compute the matrix \( \mathcal{K}_{\bar{u}} \in \mathbb{R}^{k \times k} \), where \( k \) depends on the dimension of the dictionary:

\[
\mathcal{K}^\top_{\bar{u}} = \Psi_{\tilde{Y}} \Psi_Y = (\Psi_{\tilde{Y}} \Psi_Y)^\top ,
\]

see \([17]\) for details. The matrix \( \mathcal{K}_{\bar{u}} \) can be viewed as a finite-dimensional approximation of the Koopman operator. Instead of the more common approach, where a decomposition into eigenvalues, eigenfunctions, and modes is applied to analyze the system dynamics, we follow \([10]\) and – introducing \( z = \psi(f(x)) \) – compute updates for the observable \( f(x) \) using \( \mathcal{K}_{\bar{u}} \) directly:

\[
z_{i+1} = \mathcal{K}_{\bar{u}} z_i , \quad i = 0,1,\ldots
\]

From here, we can obtain \( f(x_{i+1}) \) using the projection matrix \( P \), cf. Fig. 2 where the relation between the dynamical system \( \Phi_{\bar{u}} \), the related Koopman operator \( \mathcal{K}_{\bar{u}} \), and the EDMD approximation \( \mathcal{K}_{\bar{u}} \) is visualized. Note that \( z = f(x) \) if we use DMD instead of EDMD.

More recently, various attempts have been made to use ROMs based on the Koopman operator for both open and closed loop control problems \([18,19,23,21]\). In these approaches, the Koopman operator is either approximated for an augmented state (consisting of the actual state and the control) in order to deal with the non-autonomous control system or an affine control dependency is assumed. An alternative approach which will be utilized in this article is to replace the control system \([1]\) by a set of autonomous systems \([5]\) with constant control inputs \( \bar{u}_1,\ldots,\bar{u}_n \). This way, the optimal control problem is either transformed into a switching time problem \([10]\) or into a bilinear control problem \([22]\). By utilizing a recent convergence result for EDMD \([23]\), optimality can be guaranteed provided that the objective function can be expressed in terms of the observable.

Since the finite-set MPC problem introduced in Sec. 2 is already a problem of switching type, the approach from \([10]\) can be applied directly. The only adaptation necessary is to replace the original dynamics \([2]\) by Koopman operator based ROMs \([7]\). In the given example from Fig. 1 the 2-level inverter has \( 2^3 = 8 \) switching states, but the two zero-voltage vectors lead to the same autonomous system behavior. Even though inverter power losses may vary depending on which of the two zero-voltage inputs is applied, this will be neglected for the sake of simplicity. Hence, only seven Koopman operator based ROMs have to be computed which reduces the online computational burden for the FCS-MPC due to fewer switching sequences.

In an offline phase, we collect data for all seven system states and construct the matrices \( \mathcal{K}_{\bar{u}_i} \) to \( \mathcal{K}_{\bar{u}_j} \). Since real-time applicability is crucial, we use DMD instead of EDMD such that the matrices have a very low dimension. The relevant observations are the currents \( i_d \) and \( i_q \).
and the electrical rotor angle \( \varepsilon_{el} \). We assume nearly constant rotational speed over the prediction horizon, i.e. the rotor angle is increasing continuously such that the resulting Koopman operator is unbounded, which violates the requirements for the convergence properties and yields unsatisfactory results \([23]\). To this end, we observe the sine and cosine of the angle which results in the observation

\[
y = \begin{bmatrix} i_d & i_q & \sin(\varepsilon_{el}) & \cos(\varepsilon_{el}) \end{bmatrix}^T.
\]

The corresponding ROM hence has dimension four.

For the following experimental validation, the observations have been generated according to Fig. 3. The standard white-box motor model from \([2]\) has been utilized in a simple Simulink-based closed-loop control simulation to generate data linking the control action \( u \) to the observation \( y \). Using this data, the matrices \( K_{\pi_2} \) to \( K_{\pi_1} \) are computed via \((6)\). Following this simulation based offline training process, it becomes clear that the computed Koopman matrix can perform, at best, at the same level compared to the original model \([2]\) in an FCS-MPC application. In addition, it should be pointed out that the Koopman operator was trained only carried out for one fixed motor speed of \( n = 1000 \text{min}^{-1} \). Operating the Koopman operator based MPC at other speeds will directly result in a systematic modeling error. This issue will be discussed in Sec. 4.2. Thus, the primary objectives of this publication are to evaluate the approximation accuracy of the Koopman operator based ROM as well as its computational load compared to the baseline model and also its general feasibility in a MPC context.

\begin{figure}[h]
\centering
\includegraphics[width=0.8\textwidth]{koopman.png}
\caption{Koopman operator based ROM training and FCS-MPC application.}
\end{figure}

\section{Experimental Evaluation}

The proposed Koopman operator based FCS-MPC has been implemented on a laboratory test bench equipped with a dSPACE DS1006 rapid-control-prototyping system. The drive system under test consists of a 55 kW IPMSM manufactured by Brusa (HSM1-6.17.12-C01) and a two-level IGBT inverter from Semikron (SKiiP 1242GB120-4D). Moreover, a speed-controlled load machine is coupled with the DUT via a torque meter. The most important motor and test bench parameters are summarized in Tab. 1.

\begin{table}[h]
\centering
\caption{IPMSM and control Parameters}
\begin{tabular}{|c|c|c|}
\hline
Parameter & Value & Unit \\
\hline
Stator resistance & \( R \) & 18 m\Omega \\
Inductance in d-direction & \( L_d \) & 370 \mu H \\
Inductance in q-direction & \( L_q \) & 1200 \mu H \\
Permanent magnet flux & \( \Psi_{pm} \) & 66 mVs \\
Pole pair number & \( p \) & 3 \\
DC-link voltage & \( u_{dc} \) & 300 V \\
Mechanical speed & \( n \) & 1000 \text{min}^{-1} \\
FOC: Fixed switching frequency & \( f_{sw} \) & 10 kHz \\
FOC: Symmetrical optimum & \( a \) & 6 \\
FOC: Regular sampling cycle time & \( T_{sw, FOC} \) & 100 \mu s \\
MPC: Controller cycle time & \( T_{sw, MPC} \) & 50 \mu s \\
MPC: Max. switching frequency & \( f_{sw} \) & 10 kHz \\
MPC: Prediction horizon & \( n_p \) & 3 \\
\hline
\end{tabular}
\end{table}

At the test bench, three different control approaches have been implemented and tested:

- Field-oriented control (FOC) with PWM
- FCS-MPC with Koopman operator based ROM
- FCS-MPC with standard white-box motor model \([2]\)

The FOC was analytically tuned according to the well-known symmetrical optimum \([24]\), and it is triggered by the PWM interrupt to enable regular sampling and a fixed switching frequency. Both MPC variants compensate for the one step delay due to the digital implementation of the controllers by adding a prediction step before starting the exhaustive search with \( n_p = 3 \) steps. The FCS-MPC approach in conjunction with the selected controller cycle time of 50 \mu s results in an upper limit of the switching frequency of 10 kHz.

The turnaround times of the implemented controllers on the used hardware setup are summarized in Tab. 2. Among other test-bench specific computational overhead, the time required for the analog to digital conversions and the PLL are included. It can be seen that the Koopman based execution times are in the same range as the simple white-box model based MPC. It can be concluded that the simplest white-box motor model has the same computational complexity as the Koopman based model. When additional effects, like (cross-)saturation or iron losses, should be modeled within the white-box approach, this will directly result in an increased computational load, whereas the Koopman based black box model can be trained with the same matrix dimensions and therefore preserve the computational complexity.

\begin{table}[h]
\centering
\caption{Turnaround times of the control methods}
\begin{tabular}{|c|c|c|}
\hline
Control Method & Mean Value & Standard Deviation \\
\hline
FOC & 27.6 \mu s & 0.42 \mu s \\
Koopman based FCS-MPC & 28.5 \mu s & 0.40 \mu s \\
Standard model FCS-MPC & 26.5 \mu s & 0.39 \mu s \\
\hline
\end{tabular}
\end{table}

\subsection{4.1 Evaluation at nominal speed}

For the comparison of the three considered control approaches, the step response as well as the behavior in
steady state at the nominal fixed rotational speed of \( n = 1000 \text{ min}^{-1} \) were examined. The responses to small current reference steps are shown in Figs. 4, 5 and 6, respectively. The reference \( i_d^* \) is changed to \(-25\) A first. Afterwards \( i_q^* \) is set to \( 25\) A. For the shown responses, the currents sampled by the respective controller are used.

Both MPC variants are able to comply with the reference change. Compared to the FOC, the settling times are significantly faster for the FCS-MPC approaches. Also, the decoupling between the two current axes is slightly better when using MPC compared to FOC. Due to a lower MPC switching frequency, the ripple of the currents during steady state operation is higher. The average switching frequencies for different steady state operations are summarized in Tab. 3.

Figs. 7, 8 and 9 show the responses to large current reference steps. Here, the currents are changed from zero to \( i_d^* = -169\) A and \( i_q^* = 169\) A which corresponds to the nominal current of the motor. With the FOC, the \( d\)-current reaches the steady state operation at about \( 2\) ms. In less than \( 1\) ms the set point is reached with both MPCs. The rise time of the \( q\)-current is nearly identical in all cases. After \( 1\) ms both MPCs operate in steady state. Beginning with the step, the FOC operates for \( 0.5\) ms within the limitation of the actuating variables. Following the rise, an overshoot can be observed. The steady state is reached after \( 2\) to \( 3\) ms. During the step responses of the MPCs there is hardly any mutual influencing between the two currents. Using the FOC, a significant influence on the \( d\)-current during the rise of the \( q\)-current can be noticed.

The complete compensation of that effect takes almost

| Table 3: Average switching frequencies at rotational speed \( n = 1000 \text{ min}^{-1} \). |
|-----------------------------------------------|
| \( i_d = -25\) A, \( i_q = -169\) A, \( i_d = 25\) A, \( i_q = 169\) A |
| FOC | 10.0 kHz | 10.0 kHz |
| Koopman-MPC | 3.7 kHz | 4.1 kHz |
| standard-MPC | 3.2 kHz | 4.1 kHz |
The currents of one phase during operation with the nominal current operating points are shown in Figs. 10, 11 and 12 to determine the current ripple and harmonics. The currents are sampled at a rate of 20 MHz by means of an external transient recorder for all three control approaches. The corresponding discrete Fourier transformations (DFT) are given in Figs. 13, 14 and 15; please note the different scaling of the ordinate. The current ripple with the FOC is lower than with the MPCs, which directly follows from the constant FOC switching frequency of 10 kHz. Both MPCs use a significantly lower average switching frequency of only 4.1 kHz. For the FOC, current harmonics at multiples of the switching frequency emerge in the spectra and the resulting total harmonic distortion (THD) amounts to 2.4%. In contrast, the THD with the Koopman based MPC amounts to 15.6% and to 15.0% for the standard model based MPC. Here, the higher order harmonics are located mainly in a broader spectrum around the mean switching frequency.

4.2 Evaluation at deviating speeds

The Koopman ROM was constructed from data obtained at a constant mechanical speed of \( n = 1000 \, \text{min}^{-1} \). We now investigate the robustness of the ROM performance and consider deviating speeds. For comparison, the white-box standard model based FCS-MPC is utilized. Since the differences to the FOC approach are similar to the previous section, they will not be discussed here again.

In Figs. 16 and 17, the large signal step response behavior for both MPC approaches is shown for a mechanical speed of \( n = 100 \, \text{min}^{-1} \). Moreover, the corresponding current spectra during steady state at nominal motor current are shown in Figs. 18 and 19. It can be seen that both modeling approaches deliver nearly equal performances in the FCS-MPC framework.

Furthermore, the control behavior is also evaluated at a higher speed of \( n = 2500 \, \text{min}^{-1} \). The large signal step responses are depicted in Figs. 20 and 21 while the current spectra at steady state are given in Figs. 22 and 23. Again, the control performance of both plant modeling ap-
Figure 16: Koopman-MPC: large signal response at \( n = 100 \text{ min}^{-1} \), \( i_d \) (blue), \( i_q \) (red), setpoints (green).

Figure 17: Standard-MPC: large signal response at \( n = 100 \text{ min}^{-1} \), \( i_d \) (blue), \( i_q \) (red), setpoints (green).

Figure 18: Koopman-MPC: DFT of phase current at \( n = 100 \text{ min}^{-1} \) and nominal current.

Figure 19: Standard-MPC: DFT of phase current at \( n = 100 \text{ min}^{-1} \) and nominal current.

Figure 20: Koopman-MPC: large signal response at \( n = 2500 \text{ min}^{-1} \), \( i_d \) (blue), \( i_q \) (red), setpoints (green).

Figure 21: Standard-MPC: large signal response at \( n = 2500 \text{ min}^{-1} \), \( i_d \) (blue), \( i_q \) (red), setpoints (green).

Figure 22: Koopman-MPC: DFT of phase current at \( n = 2500 \text{ min}^{-1} \) and nominal current.

Figure 23: Standard-MPC: DFT of phase current at \( n = 2500 \text{ min}^{-1} \) and nominal current.

Problems is almost identical. However, it can be observed that there is a small steady state error in both cases. This is caused by sampling issues due to non-regular sampling approaches and general modeling errors of the baseline white-box motor model (which was also used for the Koopman ROM training). The latter effect is typical for MPC-frameworks at higher electrical fundamental frequencies and can be compensated by disturbance observers.

5 Conclusion and Outlook

Extensive experimental tests show that the Koopman operator based FCS-MPC achieves the same performance
as a white-box model based approach – also for operating points which were not contained in the training data, which highlights the robustness of the presented black box plant model. Even when tested at significantly different operation regimes compared to the training setup, the presented method shows satisfying results. To the authors’ best knowledge, this is the first real-world application of the Koopman operator based ROM to a control problem in the field of power electronics. The presented results are very promising, however, it should be stressed that this contribution is only a first proof of concept since the Koopman matrix was trained offline by using simulation generated test data from the reference white-box motor model.

There are plenty of future research questions in terms of using a Koopman operator based model for MPC. For example, the data collection can be realized using measured motor data from the test bench. By doing so, the Koopman ROM will inherently take real-world effects into account like (cross-)saturation, iron losses effects or flux linkage harmonics. This could lead to improved MPC performance without increasing the computational load due to a more complex plant model. Moreover, the training process and thus the ROM can be adapted online using streaming data [24]. Besides that, the motor’s torque can be incorporated as an additional observation for the training and the FCS-MPC can be extended to directly control the torque in an open-loop manner. In a larger scope, the presented modeling method can be also applied to more complex power electronics applications like motor drives including sine-filter between inverter and motor or DC-DC LLC resonant converters where important parasitic effects are hardly to model in white-box approaches.

Acknowledgment

This work was partially funded by the German Research Foundation (DFG) under the reference number BO 2535/11-1.

References

[1] M. Morari and J. H. Lee. Model Predictive Control: Past, Present and Future. Computers and Chemical Engineering, 23:667 – 682, 1999.
[2] C. R. Cutler and B. L. Ramaker. Dynamic Matrix Control – A Computer Control Algorithm. In Proc. Joint Automatic Control Conf., 1980.
[3] P. Cortes, M. P. Kazmierkowski, R. M. Kennel, D. E. Quevedo, and J. Rodriguez. Predictive Control in Power Electronics and Drives. IEEE Transactions on Industrial Electronics, 55, iss. 12:4312 – 4324, 2008.
[4] S. Kouro, M. A. Perez, J. Rodriguez, A. M. Llor, and H. A. Young. Model Predictive Control: MPC’s Role in the Evolution of Power Electronics. IEEE Industrial Electronics Magazine, 9(4):8–21, 2015.
[5] S. Vazquez, J. I. Leon, L. G. Franquelo, J. Rodriguez, H. A. Young, A. Magneux, and P. Zanchetta. Model Predictive Control: A Review of Its Applications in Power Electronics. IEEE Industrial Electronics Magazine, 8(1):16–31, 2014.
[6] J. Eckstein, S. Peitz, K. Schäfer, P. Friedel, U. Köhler, M. Hessel-von Molo, S. Ober-Blöbaum, and M. Dellnitz. A Comparison of two Predictive Approaches to Control the Longitudinal Dynamics of Electric Vehicles. Procedia Technology, 26:465 – 472, 2016.
[7] S. Peitz, K. Schafer, S. Ober-Blöbaum, J. Eckstein, U. Köhler, and M. Dellnitz. A Multiobjective MPC Approach for Autonomous Driven Electric Vehicles. IFAC PapersOnLine, 50(1):8674 – 8679, 2017.
[8] S. Vazquez, J. Rodriguez, M. Rivera, L. G. Franquelo, and M. Norambuena. Model Predictive Control for Power Converters and Drives: Advances and Trends. IEEE Transactions on Industrial Electronics, 64(2):935 – 947, 2017.
[9] T. Geyer and D. E. Quevedo. Performance of Multistep Finite Control Set Model Predictive Control for Power Electronics. IEEE Transactions on Power Electronics, 30(3):1633–1644, 2015.
[10] S. Peitz and S. Klus. Koopman operator-based model reduction for switched-system control of PDEs. arXiv:1710.06759, 2017.
[11] W. Peters, O. Wallscheid, and J. Böcker. A Precise Open-Loop Torque Control for an Interior Permanent Magnet Synchronous Motor Considering Iron Losses. In Annual Conference of the IEEE Industrial Electronics Society (IECON), 2012.
[12] A. Specht, S. Ober-Blöbaum, O. Wallscheid, C. Romaus, and J. Böcker. Discrete-Time Model of an IPMSM Based on Variational Integrators. In IEEE International Electric Machines and Drives Conference, 2013.
[13] B. O. Koopman. Hamiltonian Systems and Transformations in Hilbert Space. Proceedings of the National Academy of Sciences, 17(5):315–318, 1931.
[14] C. W. Rowley, I. Mezić, S. Bagheri, P. Schlatter, and D. S. Henningson. Spectral Analysis of Nonlinear Flows. Journal of Fluid Mechanics, 641:115–127, 2009.
[15] P. J. Schmid. Dynamic Mode Decomposition of Numerical and Experimental Data. Journal of Fluid Mechanics, 656:5–28, 2010.
[16] M. O. Williams, I. G. Kevrekidis, and C. W. Rowley. A Data-Driven Approximation of the Koopman Operator: Extending Dynamic Mode Decomposition. Journal of Nonlinear Science, 25(6):1307–1346, 2015.
[17] S. Klus, P. Koltai, and C. Schütte. On the Numerical Approximation of the Perron-Frobenius and Koopman Operator. Journal of Computational Dynamics, 3(1):51–79, 2016.
[18] J. L. Proctor, S. L. Brunton, and J. N. Kutz. Dynamic Mode Decomposition With Control. SIAM Journal on Applied Dynamical Systems, 15(1):142–161, 2015.
[19] S. L. Brunton, B. W. Brunton, J. L. Proctor, and J. N. Kutz. Koopman Invariant Subspaces and Finite Linear Representations of Nonlinear Dynamical Systems for Control. PLoS ONE, 11(2):1–19, 2016.
[20] M. Korda and I. Mezič. Linear Predictors for Nonlinear Dynamical Systems: Koopman Operator Meets Model Predictive Control. arXiv:1611.05537, 2016.
[21] E. Kaiser, J. N. Kutz, and S. L. Brunton. Data-driven Discovery of Koopman Eigenfunctions for Control. arXiv:1707.01114, 2017.
[22] S. Peitz. Controlling nonlinear PDEs using low-dimensional bilinear approximations obtained from data. arXiv:1801.06419, 2018.
[23] M. Korda and I. Mezič. On Convergence of Extended Dynamic Mode Decomposition to the Koopman Operator. Journal of Nonlinear Science, 28(2):687–710, 2018.
[24] J. Böcker. Lecture Notes: Mechatronics and Electrical Drives. Paderborn University, 2017.
[25] M. S. Hemati, M. O. Williams, and C. W. Rowley. Dynamic Mode Decomposition for Large and Streaming Datasets. Physics of Fluids, 26(111701):1–6, 2014.