ETH zürich

ETH Library

Data-Driven Continuous-Set Predictive Current Control for Synchronous Motor Drives

Journal Article

Author(s):
Carlet, Paolo Gherardo; Favato, Andrea; Bolognani, Saverio; Dörfler, Florian

Publication date:
2022-06

Permanent link:
https://doi.org/10.3929/ethz-b-000535679

Rights / license:
In Copyright - Non-Commercial Use Permitted

Originally published in:
IEEE Transactions on Power Electronics 37(6), https://doi.org/10.1109/TPEL.2022.3142244

This page was generated automatically upon download from the ETH Zurich Research Collection.
For more information, please consult the Terms of use.

Published on the IEEE Transactions on Power Electronics (Volume: 37, Issue: 6, June 2022).
http://doi.org/10.1109/TPEL.2022.3142244

Data-Driven Continuous-Set Predictive Current Control for Synchronous Motor Drives

Paolo Gherardo Carlet[1], Andrea Favato[1], Saverio Bolognani[2], Florian Dörfler[2]
[1] *Electric Drives Laboratory, Dept. of Industrial Engineering*, *University of Padova*, Padova (Italy)
[2] *Automatic Control Laboratory*, *ETH Zurich*, 8092 Zurich (Switzerland)
paologherardo.carlet@unipd.it, andrea.favato@phd.unipd.it, bsaverio@ethz.ch, dorfler@ethz.ch

***Abstract*—Optimization-based control strategies are an affirmed research topic in the area of electric motors drives. These methods typically rely on an accurate parametric representation of the motor equations. In this paper, we present the transition from model-based towards data-driven optimal control strategies. We start from the model predictive control paradigm which uses the voltage balance model of the motor. Second, we discuss the prediction error method, where a state-space model is identified from data, without a parametrization. Moving toward data-driven controls, we present the Subspace Predictive Control, where a reduced model is constructed based on the singular value decomposition of raw data. The final step is represented by a complete data-driven approach, named data-enabled predictive control, in which raw data is not encoded into a model but directly used in the controller. The theory behind these techniques is reviewed and applied for the first time to the design of the current controller of synchronous permanent magnet motor drives. Design guidelines are provided to practitioners for the proposed application and a way to address offset-free tracking is discussed. Experimental results demonstrate the feasibility of the real-time implementation and provide comparisons between model-based and data-driven controls.**

***Index Terms*—Data driven control, model predictive control (MPC), permanent magnet synchronous motor (PMSM), prediction error method (PEM), subspace predictive control (SPC), data-enabled predictive control (DeePC)**

I. Introduction

The interest in data analysis is constantly growing, supported by an unprecedented availability of computational power and memory storage, as well as advances in optimization, statistics and machine learning. This leads to an increasing attention towards data-enabled methods in all branches of science and engineering. This revolution has a significant impact on the control engineering too. *Data-driven control* design consists in synthesizing a controller using the data collected on the real system, without defining and identifying a parametric model for the plant [1]. This is in contrast with model-based approaches, which rely on plant modeling and identification procedures. The epitome of this model-based paradigm is arguably the model predictive control (MPC), which has been applied to power electronics control tasks for two decades, reaching an industrial and commercial level [2].

Continuous control set (CCS) MPC methods for PMSM current control, which is the focus of this paper, rely on a state-space model of the motor to build the predictive controller [3]–[5]. The model is commissioned by performing an experimental characterization of specific parameters. These procedures often include many different tests and they require specific measuring devices and proper test-bed setups. Then, the resulting accurate model can be exploited in real-time by means of look-up-tables. Alternatively, parameters could be estimated via offline [6] or online [7] procedures. Self-commissioning and auto tuning techniques are also consolidated strategies. In [8], an exhaustive survey of research and state-of-art parameter identification and self-commissioning methods for AC motor drives is discussed. In particular, these approaches are of interest when high performance control is required with sensorless applications. Finally, many methods have been proposed in literature to improve the robustness against parameter variations [9]–[13], although most of these strategies are implemented for finite-set MPCs.

The key idea behind data-driven predictive controllers is to avoid the model identification stage entirely, and design the controller directly from collected input/output (I/O) data, e.g. voltage/current samples. This approach overcomes the challenges of model selection and identification, resulting of particular interest for many industrial applications [14]. However, there are just a few examples of data-driven control applications for electric motor drives. In [15], an observer is coupled to an MPC to update the PMSM model, improving its reliability. However, this approach still relies on a parametric model. A controller design procedure was proposed by Wallscheid et al. in [16], based on deep reinforcement learning. The solution guarantees the benefits of an optimal controller, without requiring expensive computations. Many effective techniques have been presented which go toward the data-driven paradigm, named model-free [17]–[19] or parameter-free [20] algorithms. In particular, [17] and [18] propose to online update a parameter-free model, but they rely on the hypothesis that there are no data available for guessing an initial controller, which might be too restrictive.

In this work we show a transition from model-based to data-driven control design, considering as application the current control of PMSMs. This control task serves as a well-understood benchmark for new methods, despite the fact that

A preliminary version of part of the results contained in this manuscript has been previously presented at the 2020 IEEE Energy Conversion Congress and Exposition (ECCE), 11-15 Oct. 2020, Detroit, MI, USA, entitled "Data-driven predictive current control for synchronous motor drives".

This research was supported by Fondazione "Ing. Aldo Gini" – via Portello, 15 – 35129 Padova (Italy), by the University of Padova (Project SID 2017 -BIRD175428), and by ETH Zurich funds.

other traditional non-data-driven methods yield satisfactory results for this application. We consider optimization-based control scheme, i.e. MPC-type solutions. We first present a state-of-the-art CCS-MPC, whose model is obtained through a previous characterization of the motor parameters. Then, we move step by step towards more data-driven control designs, exploiting just voltage and current measurements collected from the motor. First, the prediction error method (PEM) technique coupled with MPC is presented, which is a consolidate solution for identifying a state-space model from data [21]. A further step is represented by the subspace predictive control (SPC) [22], where the collected data are processed offline by means of a least-square program, and the resulting auto-regressive model with exogeneous inputs (ARX) is de-noised by singular value thresholding. This pseudo-identification procedure is used to build a linear predictor for the currents dynamics. Finally, a completely data-driven control algorithm is presented, named the data-enabled predictive control (DeePC) [23], [24], where the system identification process is completely avoided and the collected data are directly used in the controller. This technique has already found application in power electronics [25]–[27].

The contributions of this work are manifold:

- we illustrate the perspective of data-driven control design using a predictive control framework;
- we demonstrate the practical real-time implementation of data-driven methods, which is not trivial since data-driven methods are expensive in terms of computation and samples;
- we show that data-driven paradigm can be a systematic design tool for PMSM current controllers;
- we compare the computational aspects of the presented control strategies;
- as a technical contribution, we address the problem of the offset-free tracking for the SPC and DeePC methods;
- we provide guidelines for the choice of the control parameters and excitation input signals for this application.

A relevant advantage of data-driven strategies is that they can be easily implemented as automatic procedures that excite the system with predefined input signals, perform offline calculations, and deliver a ready-to-use control law. No special skills or specialized commissioning personnel are required to set up the procedure. This approach could be interesting for some industrial challenges. For example, in compressor for refrigeration equipment or submersible pumps, offline characterizations cannot be performed when PMSMs are inaccessible. Another case of interest is multi-purpose drives, where algorithms suitable for different PMSM topologies are needed. In addition, PMSM and inverter manufacturers are often different companies and they were never meant to be integrated in the same application. Moreover, if the motor drive needs to be manually re-tuned during its life-cycle, data-driven procedures represent a simple and reliable method to adapt the initial design.

The goal of this paper is not to demonstrate the superiority of the data-driven paradigm over the model-based approach. Instead, this work provides some concrete, although preliminary, answer to the contemporary question *how data-driven techniques can impact the electric drives field.* Differently from other machine-learning based solution that can be found in literature, the proposed schemes are more computationally efficient, less data hungry and more suitable to rigorous stability and robustness analysis [28].

II. Model-Based MPC of PMSM Currents

According to the MPC paradigm, the future control input sequence $u = [u(k), u(k+1), \dots u(k+N-1)]^{\mathrm{T}}$ is optimized in order to steer the predicted future output $y = [y(k+1), y(k+2), \dots y(k+N)]^{\mathrm{T}}$ to a desired reference $r = [r(k+1), r(k+2), \dots r(k+N)]^{\mathrm{T}}$. Only the first optimal input of the sequence $u(k)$ is applied to the plant (receding horizon principle). Thus, the following optimization problem is solved at each control period:

$$\min_{u,x,y} \quad \left(\|y - r\|^2_{\mathrm{Q}} + \|u\|^2_{\mathrm{R}} \right) \tag{1a}$$

$$\text{subject to} \quad x(k+1) = \mathrm{A}x(k) + \mathrm{B}u(k),\ y = \mathrm{C}x, \tag{1b}$$

$$u(k) \in \mathcal{U},\ k = 0, \dots, N-1 \tag{1c}$$

where N is the prediction horizon, $\mathrm{Q} \geq 0$ and $\mathrm{R} > 0$ are two weighting matrices, A and B represent the state space model used to predict the output $y = \mathrm{C}x$, and $\mathcal{U}$ is the input feasible set. Considering the specific application, $\mathcal{U}$ is a polytope [29]. If the set $\mathcal{U}$ is neglected, the problem is referred to unconstrained, and it has a closed-from solution of reduced computational burden. On the opposite, if the polytopic constraints are included, the optimization problem becomes a quadratic program (QP) which requires an online QP solver like qpOASES, as in [30], but it is still usually solvable in real-time.

In the context of PMSM currents control, future currents are estimated by exploiting a parametric model, based on the PMSM voltage balance equations, represented in the dq reference frame, synchronous with the rotor flux. The equations are arranged in a state-space form and discretized using the explicit Euler approximation technique:

$$\begin{aligned} &i_{\mathrm{dq}}(k+1) = \mathrm{A}i_{\mathrm{dq}}(k) + \mathrm{B}u_{\mathrm{dq}}(k) + \mathrm{B}h(k) \\ &\mathrm{A} = \begin{bmatrix} 1 - R_{\mathrm{s}}\dfrac{T_{\mathrm{s}}}{L_{\mathrm{d}}} & \omega_{\mathrm{e}}\dfrac{L_{\mathrm{q}}}{L_{\mathrm{d}}}T_{\mathrm{s}} \\ -\omega_{\mathrm{e}}\dfrac{L_{\mathrm{d}}}{L_{\mathrm{q}}}T_{\mathrm{s}} & 1 - R_{\mathrm{s}}\dfrac{T_{\mathrm{s}}}{L_{\mathrm{q}}} \end{bmatrix}, \mathrm{B} = \begin{bmatrix} \dfrac{T_{\mathrm{s}}}{L_{\mathrm{d}}} & 0 \\ 0 & \dfrac{T_{\mathrm{s}}}{L_{\mathrm{q}}} \end{bmatrix}, \end{aligned} \tag{2}$$

where R_{s} is the stator winding resistance, T_{s} is the sampling period, ω_{e} is the electric angular speed and L_{d} and L_{q} are the d and q-axis inductances, respectively. Moreover, i_{dq} and u_{dq} are the dq currents and voltages, respectively. u_{dq} are the inputs of the system whereas i_{dq} are the states. Finally, $h = [0 - \omega_{\mathrm{e}}\Lambda_{\mathrm{pm}}]^{\mathrm{T}}$ is the back-electro motive force (back-EMF) due to the permanent magnet flux linkage Λ_{pm}. In the considered application, the full state, i.e. motor currents, is measurable. This model neglects the cross-saturation phenomena, as well as iron-saturation and back-EMF harmonics effects. Thus, the model can result as oversimplified for some PMSM topologies, such as pure reluctance motors. However, many CCS-MPCs proposed in literature work with even more simplified models, obtaining indeed good results. In particular, the dependence of

matrix A on the operating speed ω_e is neglected, preferring a constant A matrix for the real-time implementation [29]. We will see later that the data-driven paradigm overcomes the problem of selecting a model structure. Moreover, it is worth noticing that the Euler discretization does not introduce significant errors because of the high sampling rates, which are typical of the power electronics area.

An integral action is included in the MPC formulation by means of the velocity form of the MPC problem (1) [31], in order to achieve an unbiased current reference tracking. The discussion about the offset-free data-driven control is given in Sec. III-C.

III. Towards Data-Driven Control of PMSM Drives

A data-driven controller for PMSM solves essentially a current reference tracking problem analogous to the one presented in (1). However, in contrast to the parametric model (2) used in the MPC solution, a non-parametric model is adopted, consisting of raw measurements arranged in a matrix representation. The construction of this model happens offline, therefore it is not an adaptive controller. A data-driven controller design procedure consists of two steps:

- A *data collection step*, followed by offline rearrangement of the voltages/currents samples into proper matrices;
- An *online program*, when the tracking problem is solved, with the voltages/currents samples matrices acting as a constraint. In this online step, the controller has access to the latest I/O (voltage/current) samples and optimizes the predictions over an horizon of N steps.

A. Data Collection and Offline Computations

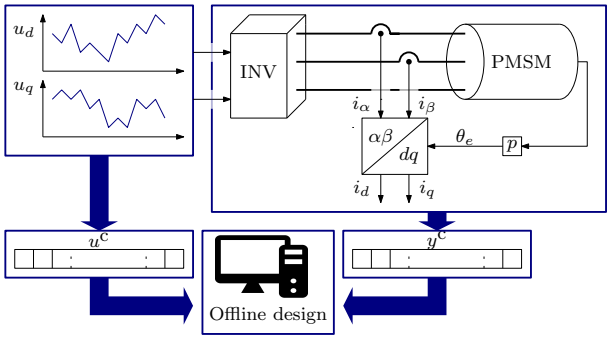
All the considered data-driven designs begin from the collection of a T-long sequence of I/O voltages u^{c} and currents y^{c} measurements (Fig. 1(a)), where the superscript c stands for collected. The sequence $u^{\mathrm{c}} = [u_1^{\mathrm{c}}; u_2^{\mathrm{c}}; ...; u_T^{\mathrm{c}}] \in \mathbb{R}^{2T}$ contains the inverter reference voltages and it fulfills the *persistency of excitation* requirement [32, Corollary 2], that is the Hankel matrix of inputs in (4) needs to have full raw rank. The selection of the input signal is further discussed in Sec. IV-A. The resulting output sequence contains the dq currents $y^{\mathrm{c}} = [y_1^{\mathrm{c}}; y_2^{\mathrm{c}}; \dots; y_{\mathrm{T}}^{\mathrm{c}}] \in \mathbb{R}^{2T}$.

PEM-MPC — In the PEM-MPC method the state-space model (1b) is used, similarly to a standard MPC approach. However, the coefficients of the state-space matrices A and B used in (1b) are inferred from data by means of an ordinary least-square problem[1] that involves the sequence u_c and y_c:

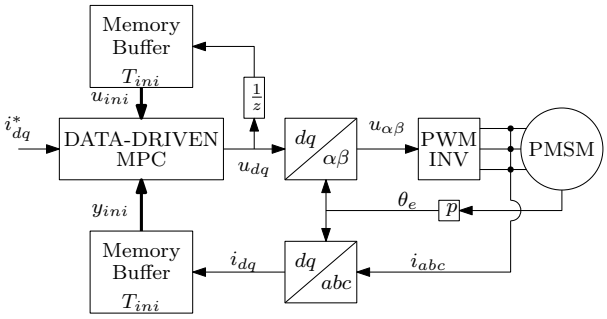
$$\min_{\mathrm{A,B}} \sum_{k=1}^{T-1} \|x_c(k+1) - \mathrm{A}x_c(k) - \mathrm{B}u_c(k)\|^2. \tag{3}$$

The main difference between the resulting model and the parametric voltage balance equation (2) is that the latter inherently requires the ad-hoc identification procedures to identify all the electric parameters (R_{s}, L_{d}, L_{q}, Λ_{pm}). The PEM method, instead, does not enforce any parametrization of the model and the resulting matrices can, in general, have a structure that is different from the one of (2).

[1]We refer to [25] for a discussion on how to solve this problem numerically.



(a) Scheme of the data collection process.



(b) Block scheme of the online controller.

Fig. 1: Overview of the data collection step and the online program.

SPC — In the SPC algorithm, the state-space model (1b) is replaced by a different algebraic constraint that relates the future currents trajectory with the past T_{ini} voltages/currents samples and the future N input voltage samples.

To obtain this model, two Hankel matrices $\mathscr{H}(u^{\mathrm{c}})$ and $\mathscr{H}(y^{\mathrm{c}})$ are built using the collected sequences u^{c} and y^{c}:

$$\mathscr{H}(u^{\mathrm{c}}) := \begin{bmatrix} u_1 & u_2 & \cdots & u_{T-T_{\mathrm{ini}}-N+1} \\ u_2 & u_3 & \cdots & u_{T-T_{\mathrm{ini}}-N+2} \\ \vdots & \vdots & & \vdots \\ u_{T_{\mathrm{ini}}+N} & u_{T_{\mathrm{ini}}+N+1} & \cdots & u_T \end{bmatrix}. \tag{4}$$

The number of columns of a Hankel matrix is hereinafter denoted as L. Given T data and the horizon lengths N and T_{ini}, L can be computed as: $L = T - T_{\mathrm{ini}} - N + 1$. The output matrix $\mathscr{H}(y^{\mathrm{c}})$ is built in an analogous way from the samples y^{c}. Then, the matrices are partitioned in *Past* and *Future* sub-blocks:

$$\begin{bmatrix} \mathrm{U}_P \\ \mathrm{U}_F \end{bmatrix} := \mathscr{H}(u^{\mathrm{c}}), \qquad \begin{bmatrix} \mathrm{Y}_P \\ \mathrm{Y}_F \end{bmatrix} := \mathscr{H}(y^{\mathrm{c}}), \tag{5}$$

where U_P contains the first T_{ini} block rows of $\mathscr{H}(u^{\mathrm{c}})$, i.e. $2T_{\mathrm{ini}}$ rows, and U_F the remaining N block rows. The block Hankel matrices Y_P and Y_F are similarly obtained. The dimensions of all the presented matrices are summarized in Table I for convenience. The I/O block Hankel matrices U_P, U_F, Y_P and Y_F are used in the SPC design to construct an ARX predictor [33]:

$$y = \mathrm{P}_w \begin{pmatrix} u_{\mathrm{ini}} \\ y_{\mathrm{ini}} \end{pmatrix} + \mathrm{P}_u u, \tag{6}$$

where $u_{\mathrm{ini}}, y_{\mathrm{ini}} \in \mathbb{R}^{2T_{\mathrm{ini}}}$ are the past dq voltage and current samples, respectively, $u, y \in \mathbb{R}^{2N}$ are the future ones. The

TABLE I: Overview of matrices dimensions for the considered PMSM current control application.

Matrix	$\mathscr{H}(u^c)$	$\mathscr{H}(y^c)$	U_P	Y_P	U_F	Y_F	P_w	P_u	M	Φ	A	B
rows	$2(T_{\text{ini}}+N)$	$2(T_{\text{ini}}+N)$	$2T_{\text{ini}}$	$2T_{\text{ini}}$	$2N$	$2N$	$2N$	$2N$	L	L	2	2
columns	L	L	L	L	L	L	$4T_{\text{ini}}$	$2T_{\text{ini}}$	$4T_{\text{ini}}$	$2N$	2	2

matrices P_w and P_u are computed solving the least-square problem

$$\min_{\mathrm{P}_w,\mathrm{P}_u} \left\| \mathrm{Y}_F - \begin{bmatrix} \mathrm{P}_w & | & \mathrm{P}_u \end{bmatrix} \left[\begin{array}{c} \mathrm{U}_P \\ \mathrm{Y}_P \\ \hline \mathrm{U}_F \end{array}\right] \right\|^2, \tag{7}$$

where P_w multiplies the two blocks $\begin{bmatrix}\mathrm{U}_P \\ \mathrm{Y}_P\end{bmatrix}$ and P_u multiplies the block U_F. The matrix P_w is exploited in (6) to set the initial condition of the prediction, i.e. to compute the term $\mathrm{P}_w(u_{\text{ini}},\, y_{\text{ini}})^{\mathrm{T}}$. A singular value decomposition (SVD) of the initial trajectory predictor P_w can be performed to mitigate the noise effect in the data [22]. Only the dominant singular values are used to construct a reduced-rank matrix.

DeePC — The design of a DeePC controller is purely data-driven, as the data block Hankel matrices defined in (5) are used in their raw form in the controller. This method is based on the so called *Fundamental Lemma* of *behavioral system theory* [32], which guarantees that (under persistency-of-excitation assumptions on u^c) any trajectory of the system needs to satisfy, for a unique $g \in \mathbb{R}^L$, the linear equations

$$\begin{bmatrix}\mathrm{U}_P \\ \mathrm{Y}_P \\ \mathrm{U}_F \\ \mathrm{Y}_F\end{bmatrix} g = \begin{bmatrix} u_{\text{ini}} \\ y_{\text{ini}} \\ u \\ y \end{bmatrix}. \tag{8}$$

Implicitly, (8) serves as a predictor of the future N-long I/O voltages/currents trajectory $(u,\, y)$ based on T_{ini}-long I/O initial trajectory $(u_{\text{ini}},\, y_{\text{ini}})$. If we consider $(u,\, y)$ as free optimization variables, the vector g that satisfies the first two block-equations of (8) can be expressed explicitly as

$$g = \begin{bmatrix}\mathrm{U}_P \\ \mathrm{Y}_P\end{bmatrix}^{\dagger} \begin{bmatrix} u_{\text{ini}} \\ y_{\text{ini}} \end{bmatrix} + \Phi z = \mathrm{M} \begin{bmatrix} u_{\text{ini}} \\ y_{\text{ini}} \end{bmatrix} + \Phi z, \tag{9}$$

where $\dagger$ denotes the Moore-Penrose pseudo-inverse operator, and Φ, such that $[\mathrm{U}_P, \mathrm{Y}_P]^{\mathrm{T}} \backslash \Phi = 0$, represents a basis of the kernel of M. Both Φ and M can be computed offline using standard linear algebra routines. This decomposition allows expressing the future trajectory as a function of the lower-dimensional variable z, and turns out to be useful in the online phase of the unconstrained control problem, as explained in the next subsection.

B. Computational Aspects of the Online Stage

In the online stage, the MPC tracking problem (1) is solved, but with different representations in place of (1b) depending on the adopted data-driven method. Both the unconstrained and constrained solutions are now discussed for each data-driven method, clarifying the practicality of their real-time implementation from the computational burden point of view. A complete overview of the data-driven design procedures is provided in Fig. 2, where the differences between the presented methods are highlighted both for the offline and online stage.

PEM-MPC — PEM-MPC algorithm is completely analogous to a standard model-based MPC, from the point of view of the online program. It is worth remembering that two possible online controllers can be obtained, depending on the presence or not of the constraints (1c). If the problem is unconstrained ((1c) is absent), the PEM-MPC yields a linear feedback controller [5] of the form $u = \mathrm{K}^r r + \mathrm{K}^x x(k)$. On the other hand, the QP problem requires an iterative solver as in [30], if input constraints are included. In both situations, the complexity of the PEM-MPC is the same of a standard model-based MPC, which is amenable for real-time implementation on adequate hardware. The dimension of the decision variable coincides with the dimension of $u \in \mathbb{R}^{2N}$, thus it scales linearly with the prediction horizon. In the considered application, the full state of the system is available, but in general the PEM-MPC requires a state estimator. The other two data-driven methods, SPC and DeePC, do not require a state estimator, since they naturally work with the plant outputs.

SPC — The SPC algorithm solves the same tracking problem (1) as in MPC or PEM-MPC, but with the state-space model (1b) replaced by the predictor (6):

$$\min_{u,y} \quad ||y-r||^2_{\mathrm{Q}} + ||u||^2_{\mathrm{R}} \tag{10a}$$

$$\text{subject to} \quad y = \mathrm{P}_w \begin{bmatrix} u_{\text{ini}} \\ y_{\text{ini}} \end{bmatrix} + \mathrm{P}_u u \tag{10b}$$

$$u(k) \in \mathcal{U},\ k = 0, \ldots, N-1. \tag{10c}$$

Similarly to the PEM-MPC, if the constraints (10c) are not present, then we can solve the problem in closed-form by substituting the predictor equation (10b) into (10a) and by setting the gradient of the resulting convex quadratic cost to zero. The resulting online controller is a linear feedback of the form $u = \mathrm{K}^{r'} r + \mathrm{K}^{\text{ini}}[u_{\text{ini}}, y_{\text{ini}}]^{\mathrm{T}}$. If the constraints (10c) are present, the minimization program can be solved online, at the same computational complexity of the PEM-MPC one. In fact, the computational burden depends on the length of u. We remind that this property does not hold for those systems whose states and outputs have different dimensions.

DeePC — The DeePC algorithm, because of the implicit form of the algebraic constraint, requires the minimization over the decision variables g, u, y:

$$\min_{g,u,y} \quad ||y-r||^2_{\mathrm{Q}} + ||u||^2_{\mathrm{R}} + \lambda_g ||g||^2 \tag{11a}$$

$$\text{s.t.} \begin{bmatrix}\mathrm{U}_P \\ \mathrm{Y}_P \\ \mathrm{U}_F \\ \mathrm{Y}_F\end{bmatrix} g = \begin{bmatrix} u_{\text{ini}} \\ y_{\text{ini}} \\ u \\ y \end{bmatrix}, u(k) \in \mathcal{U},\ k = 0, ..., N-1, \tag{11b}$$

where λ_g adds a regularization on the decision variable g. In fact, if noisy data are used, the Hankel matrices are full raw rank, but the realized control error in (11a) could be

different from the predicted one. Thus, the term $\lambda_g||g||^2$ helps to robustify the control problem [34, Section III.C]. In the unconstrained case, the problem can be solved directly using the null-space representation presented in (9). The future currents and voltages sequences u and y are replaced in (11a) with $\mathrm{U}_F g$ and $\mathrm{Y}_F g$, respectively, obtaining

$$\min_z \; ||\mathrm{Y}_F(\mathrm{M}\begin{bmatrix} u_{\text{ini}} \\ y_{\text{ini}} \end{bmatrix} + \Phi z) - r||^2_{\mathrm{Q}} + ||\mathrm{U}_F(\mathrm{M}\begin{bmatrix} u_{\text{ini}} \\ y_{\text{ini}} \end{bmatrix} + \Phi z)||^2_{\mathrm{R}}. \tag{12}$$

The solution of the problem is available in closed form as $z^{\text{opt}} = H^{-1}d^{\mathrm{T}}$, where the Hessian matrix H and the linear term d are defined as:

$$\begin{aligned} H &:= \Phi^{\mathrm{T}}\mathrm{Y}_F^{\mathrm{T}}\mathrm{Q}\mathrm{Y}_F\Phi + \Phi^{\mathrm{T}}\mathrm{U}_F^{\mathrm{T}}\mathrm{R}\mathrm{U}_F\Phi \\ d &:= \left(r - \mathrm{Y}_F\mathrm{M}\begin{bmatrix} u_{\text{ini}} \\ y_{\text{ini}} \end{bmatrix}\right)^{\mathrm{T}}\mathrm{Q}\Phi - \left(\mathrm{U}_F\mathrm{M}\begin{bmatrix} u_{\text{ini}} \\ y_{\text{ini}} \end{bmatrix}\right)^{\mathrm{T}}\mathrm{R}\Phi. \end{aligned} \tag{13}$$

The Hessian inversion can be evaluated offline with proper numerical techniques, further reducing the complexity of the scheme. More details on the closed-form solution of the unconstrained DeePC can be found in [25]. Starting from the optimal value of z^{opt}, (9) is used to compute g^{opt}, and, finally, the sequence of optimal input u^{opt}. It is still possible to condense this controller in a feedback law similar to the SPC, with a decision variable that scales linearly with the prediction horizon length. The constrained solution of (12) would instead require an online QP-solver. However, the dimension of the decision variable g can be large, as it depends on the number of samples used in (8). Thus, the real-time implementation of the DeePC algorithm is still a challenging problem.

In conclusion, three main aspects differentiate the SPC and DeePC methods [34]: the way the predictor is built, the underlying prediction model and the variables over which the QP problem is solved. In fact, the SPC forces a least-square fit to a linear time invariant (LTI) system model, whereas the DeePC does not. Thus, SPC is more suited for LTI systems or linear parameter varying ones. On the other hand, DeePC exhibits interesting features also when applied to non-linear system, even if the fundamental lemma requires in theory a linear time invariant plant, e.g. the grid connected inverter application shown in [25]. Finally, SPC solves the tracking problem in the input u, whereas the DeePC in g.

C. Integral Action

An integral action is needed to avoid bias errors in the currents reference tracking for the SPC and DeePC algorithms. For instance, the back-EMF induced by the magnets acts as a constant disturbance in the voltage equation, inducing a steady-state error in reference tracking. Following this principle, we introduce this framework also for data-driven controllers. The integral action can be included by formulating the optimization problem in its velocity-form [31]. The idea is to perform the data collection stage filling the matrices with incremental data,

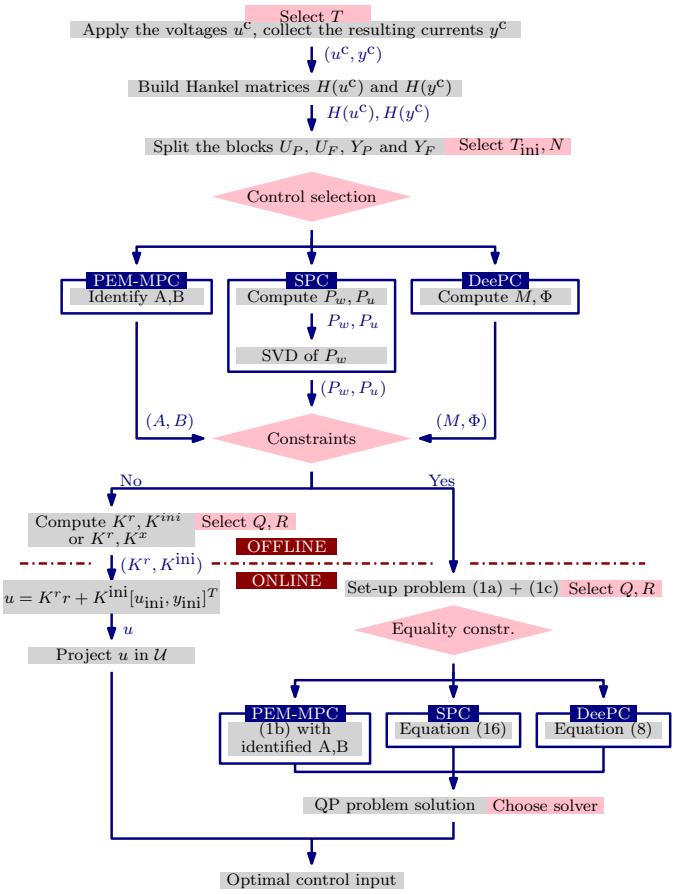
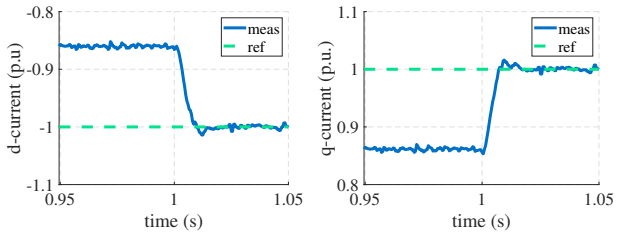


Fig. 2: Overview of the three proposed data-driven controller design procedures.



(a) d-current (b) q-current

Fig. 3: Offset-free tracking error: simulation of DeePC algorithm with no integral action (before $t = 1$s) and with the offset-free implementation after $t = 1$s.

e.g. $\Delta y = y(k) - y(k-1)$. For instance, the DeePC problem in (11) is written as follows:

$$\begin{aligned} \min_{g,\Delta u,\Delta y} \quad & ||\Delta y - r'||^2_{\mathrm{Q}} + ||\Delta u||^2_{\mathrm{R}} + \lambda_g||g||^2 \\ \text{subject to} \quad & \begin{bmatrix} \mathrm{U}'_P \\ \mathrm{Y}'_P \\ \mathrm{U}'_F \\ \mathrm{Y}'_F \end{bmatrix} g = \begin{bmatrix} \Delta u_{\text{ini}} \\ \Delta y_{\text{ini}} \\ \Delta u \\ \Delta y \end{bmatrix} \\ & u(k) = u(k-1) + \Delta u(k) \in \mathcal{U}, \; k = 0, 1..., N-1 \\ & r(k)' = r(k) - y(k), \; k = 1, ..., N. \end{aligned} \tag{14}$$

$[\mathrm{U}'_P\mathrm{Y}'_P\mathrm{U}'_F\mathrm{Y}'_F]^{\mathrm{T}}$ are the Hankel matrices filled with incremental data. The optimization problem (14) is solved for g, then $\Delta u(k)$ is found.

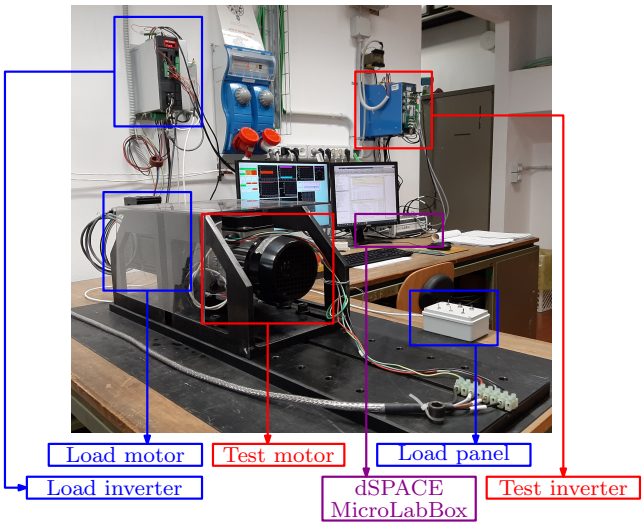


Fig. 4: Test-bed layout.

TABLE II: Overview of the drive parameters.

Parameter	Symbol	Value
Pole pairs	p	3
Phase resistance	R_s	$1\,\Omega$
d-axis inductance	L_d	$0.010\,\mathrm{H}$
q-axis inductance	L_q	$0.014\,\mathrm{H}$
PM flux-linkage	Λ_pm	$0.26\,\mathrm{V\,s}$
Nominal current	I_N	$6.2\,\mathrm{Arms}$
Nominal d current	$I_\mathrm{N,d}$	$-1.1\,\mathrm{A}$
Nominal q current	$I_\mathrm{N,q}$	$8.7\,\mathrm{A}$
Nominal speed	Ω_N	$1000\,1/\mathrm{min}$
DC bus voltage	U_DC	$300\,\mathrm{V}$
Sampling period	T_s	$100\,\mu\mathrm{s}$

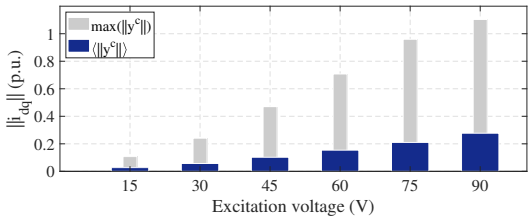
The effectiveness of the proposed solution is shown in Fig. 3. The nominal current reference has been set and steady state is reached, while the motor is kept at nominal speed. Before time $t = 1\mathrm{s}$, the standard data-driven formulation is considered as controller. As can be seen, a bias appears in the tracking. At time $t = 1\mathrm{s}$, the controller designed with incremental data is selected and the bias is removed.

IV. Experimental Validation

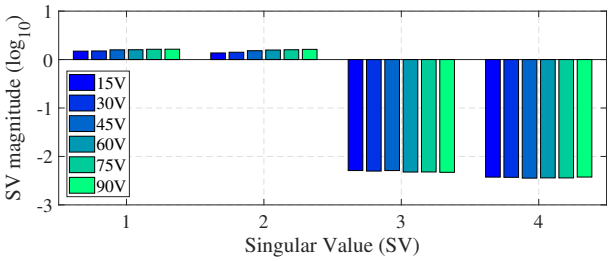
The authors propose the experimental validation on an interior permanent magnet motor. The nominal parameters of the considered machine are reported in Table II, while the test-bench layout is shown in Fig. 4. All the algorithms, i.e. the MPC, PEM-MPC, SPC and DeePC, are real-time implemented on the dSPACE MicroLabBox at a sample rate of $T_\mathrm{s} = 100\,\mu\mathrm{s}$. The MPC nominal model is commissioned by means of standard tests and offline post-processing. The motor under test is not significantly affected by the magnetic non-linearity.

A. Data Acquisition Step

The test designed to collect I/O data from the interior permanent magnet (IPM) motor consists of excitation with a random (detailed below) dq voltage vectors sequence u^c and the measurement of the dq currents via LEM sensors. Thanks to this



(a) Output currents during the excitation tests for different u_exc values: mean and maximum output currents values.



(b) Singular values analysis of P_w for different u_exc values.

Fig. 5: Overview of some key parameters of the data collection test.

choice, the rotor is not required to be locked or to be maintained at standstill by another motor. The selected zero-mean voltage sequence induces zero-mean currents and, consequently, a zero mean torque. Since the mechanical dynamic is much slower than the electric one, the rotor remain at standstill even if instantaneously the torque is not zero. In addition non-linear frictions help to avoid rotations of the motor. If the mechanical inertia of the system is too low or the frictions are negligible the rotor could slightly move, as happen for other injection-based commissioning techniques [35]. Designers should be aware of this potential issue in some specific applications.

The criteria to select the voltages amplitude is here discussed. The motor is driven by a two-level voltage source inverter with a DC bus voltage of 300 V. The voltage sequence is generated by picking the values from a uniform probability distribution in the interval $[-u_\mathrm{exc}, u_\mathrm{exc}]$. We propose a test to analyze the effects of u_exc on the sequence y^c and the data-driven design. Fig. 5(a) refers to several excitation tests, characterized by different values of u_exc. On one hand, the maximum excitation voltage should be limited to avoid over-currents, preserving a safe motor operation. The figure, in fact, shows that the mean value of the currents samples are quite low with respect to the nominal value. However, the nominal current value, for the proposed motor, is achieved using $u_\mathrm{exc} = 90\,\mathrm{V}$, i.e. the 30% of the DC bus voltage. Higher excitation voltages should be avoided. On the other hand, a too low voltage excitation could lead at least to current sampling issues, due to small signal to noise ratios. Moreover, we need to take into consideration also other problems, i.e. if the information carried by the data is rich enough to describe the current dynamics. The PWM synthesis of low voltages could emphasize some inverter non-linearities, e.g. not properly compensated dead-times, that are not of interest of our identification. In order to evaluate if the data are collected properly, the dominant singular values of the matrix P_w are analyzed (see the logarithmic plot in Fig. 5(b)). The number of dominant values should be coherent with the

TABLE III: Matrices dimensions resulting from the design choices: $N = 3$, $T_{\text{ini}} = 1$ and $T = 100$ samples.

Matrix	$\mathscr{H}(u_c)$	$\mathscr{H}(y_c)$	U_P	Y_P	U_F	Y_F	P_w	P_u	M	Φ	A	B
rows	8	8	2	2	6	6	6	6	97	97	2	2
columns	97	97	97	97	97	97	4	2	4	6	2	2

anticipated dimension of the state, see Table I. Two dominant values characterize the considered dynamic, as expected.

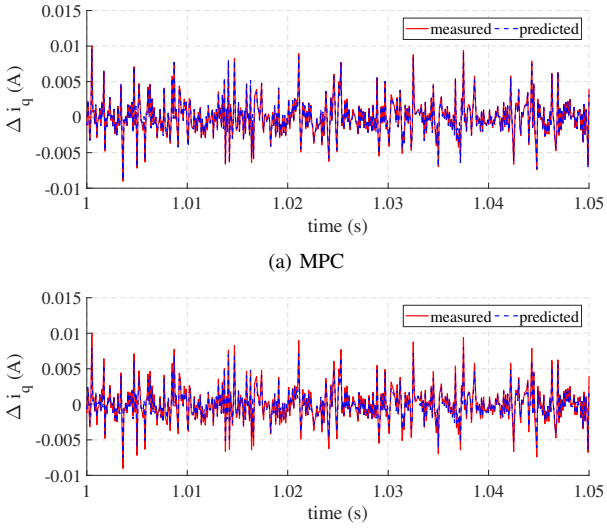
B. Parameters Selection

In this section we address the problem of parameters selection for designing the data-driven controls. The prediction horizon length N is chosen according to the MPC framework, i.e. $N = 3$. This value is a good trade-off between accuracy and computational effort for this application [29]. Moreover, all the these controllers share the same cost function; thus, equal weighting matrices Q and R are chosen. In particular, Q is the identity matrix, whereas R is the identity scaled by a factor 0.0001. We consider the robust formulation of the DeePC, and the related parameter in (11) has been set to $\lambda_g = 0.1$.

Two parameters that characterize the data-driven algorithms are the length of the initial trajectory T_{ini} and the number of samples T. The trajectory $[u_{\text{ini}}, y_{\text{ini}}]^{\mathrm{T}}$ replaces the initial condition for the prediction. Thus, it determines the inherent system state, and the parameter T_{ini} provides a complexity for the model. In [32], the system lag[2] l is used to find a lower bound for T_{ini}. In particular, if $T_{\text{ini}} \geq l$ the system prediction is uniquely determined. Thank to this criteria, the value of T_{ini} can be chosen even without knowing the system dimension, but using an estimate of it. Since the system lag is known for the considered application (i.e $l = 1$), we set $T_{\text{ini}} = 1$. The length T of the recorded I/O vectors should be long enough to make sure that the Hankel matrices have full rank. The *Fundamental Lemma* in [32] gives a lower bound for T, whose value for the considered application is $T \geq 3(T_{\text{ini}} + N + 2) - 1$. We selected $T = 100$ samples, which satisfies the inequality.

C. Complexity of the online program

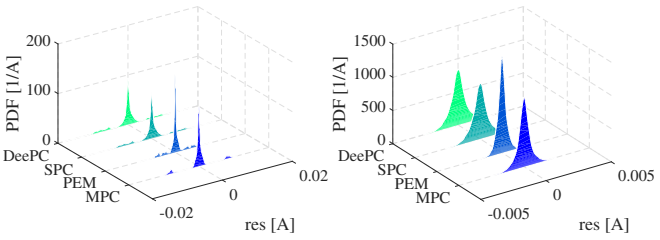
The design choices described in the previous subsection set all the dimensions of the matrices presented in TABLE III. All the controllers have been implemented in their unconstrained version, i.e. a feedback law of the form $u = K^{\mathrm{r}} r + K^{\text{ini}}[u_{\text{ini}}, y_{\text{ini}}]^{\mathrm{T}}$ or $u = K^{\mathrm{r}} r + K^{\mathrm{x}} x(k)$. This means that the turn-around-time of all the controllers are similar. A slightly higher computation time is required for the first feedback law. The turn-around-time of each predictive control scheme is about 9.6 µs - 9.7 µs, depending on the specific feedback-law. The dSPACE MicroLabBox is equipped with a 2 GHz NXP QorIQ P5020 microprocessor. The number of computations required by the feedback laws scales linearly with respect to the length of the prediction horizon N and the length of the initial trajectory T_{ini}.



(a) MPC

(b) DeePC

Fig. 6: Accuracy of the data-driven predictors in the estimation of the q-axis current variation.



(a) d-axis current (b) q-axis current

Fig. 7: Residual analysis of the prediction error: probability distribution function of the residuals for the presented predictors.

D. Accuracy of the Data-Driven Predictor

The accuracy of the data-driven predictors is investigated in this subsection, taking the model-based MPC as benchmark. This analysis is performed during steady-state operation, when the motor is working at the nominal maximum-torque-per-ampere current point (see Table II) at standstill. During the tests the currents are regulated by standard PI controllers. We are interested on the open-loop prediction accuracy of the methods. This means that the predictors are fed by current measurements and the reference voltages computed by the PIs. A first qualitative information on the accuracy is provided by Fig. 6(a) and Fig. 6(b). The figures show the comparison between the measured q-axis current increments and the predicted ones, using respectively the predictor obtained with (3) and (12). We observe a good correspondence between measurements and

[2]The lag l of a linear system is the smallest integer value for which the observability matrix $\mathcal{O} = [C\ CA ... CA^{l-1}]^{\mathrm{T}}$ has full rank.

TABLE IV: Residual analysis of the prediction error: mean and standard deviation of the residuals.

Controller	MPC		PEM		SPC		DeePC	
	d-axis	q-axis	d-axis	q-axis	d-axis	q-axis	d-axis	q-axis
Mean (μA)	2.4	−0.4	−1.2	−0.2	−2.9	−0.4	−2.8	−0.4
Standard deviation (μA)	3100	500	2100	350	3800	590	3800	590

predictions for both the controllers.

The residuals between measured and estimated currents are considered as performance index, as suggested in [15]. The results of this analysis are reported in Fig. 7. The figures show the estimated probability density function of the d and q residuals for all the described predictors. From literature [15], we expect a zero mean normal distribution of the residuals, which is coherent with the obtained results. The PEM-MPC predictor appears the most accurate one, proving that using data to validate the commissioning tests is an interesting tool.

E. Online Unconstrained Controller

In this subsection we provide a comparison between model-based and data-driven designed controllers in terms of step current reference response. In particular, the reference r is changed from zero to the nominal maximum-torque-per-ampere current. The model-based MPC adopts the motor parameters which were previously obtained by means of characterization procedures (see Table II). All the data-driven controllers are designed from the same data recording, in particular the one defined by a $u_{\text{exc}} = 50\,\text{V}$.

The step responses are compared at standstill in Fig. 8. It is interesting to notice that the data-driven designs allow achieving similar performances with respect to the model-based controller. In fact, the commissioning effort of all the proposed algorithms in terms of measurement apparatus, number of carried out tests and their complexity and duration is much lighter compared to the characterization required to build an accurate model-based controller [8]. Among data driven controllers, the DeePC is considered the most data-oriented algorithm, because it uses raw data without any pre-processing. Despite the direct exploitation of raw data, it has almost the same performance as the others. We reported the same step response analysis also at nominal speed rate in Fig. 9. This test confirms the effectiveness of the integral action included in the data-driven control framework. The back-EMF and the state transition matrix A of the IPM motor model (2) depend on the operating speed. Thus, a bias in the current tracking should be observed if the integral action is missing (as in Fig. 3). We underline that the proposed data-driven methods seems very effective for the q-axis current. Moreover, other tools can be used to further improve their behavior, in particular the one of the DeePC (see [23]). In addition, accordingly to [36], a feed-forward term can be nested in the controller to improve disturbance rejection performances, without penalizing the overshoot in the dynamics. We therefore believe that there is much unexplored potential to improve the performance to data-driven controllers.

Concerning the steady-state behaviors, a total harmonic distortion analysis is performed on the motor currents at nominal speed. The results are briefly summarized in TABLE V. The distortion is quite low, as expected from a continuous-set MPC, which encapsulates a modulator in the controller structure. The PEM-MPC seems again the preferable structure, as observed also in the previous section. Nevertheless, both SPC and DeePC grant the same harmonic distortion of a benchmark (model-based) velocity-form MPC.

TABLE V: Total harmonic distortion analysis at the nominal point.

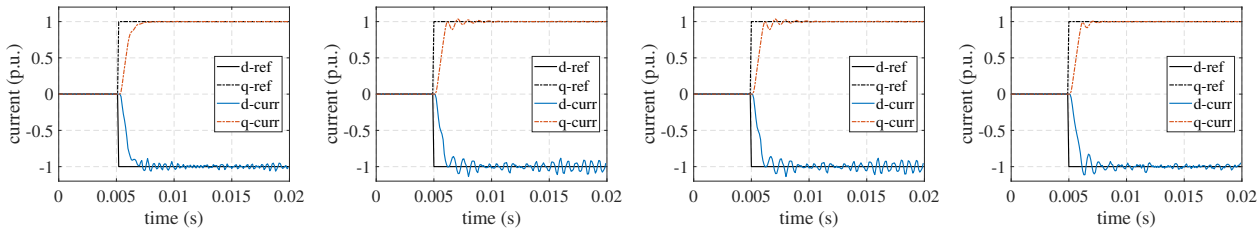
Controller	MPC	PEM	SPC	DeePC
THD	0.32%	0.25%	0.36%	0.36%

Finally, the closed-loop cost analysis is proposed to provide a different insight on the steady-state performance. The closed-loop cost is computed for each control step using (14), where the weight matrices are designed as described in Sec. IV-B. The test considers the machine at standstill (Fig. 8) and at nominal speed (Fig. 9), when steady-state condition is reached. Results are described in Fig. 10, where the cost is reported in decibel. The closed-loop cost achieved by the data-driven and model-based controllers are not significantly different, both at standstill Fig. 10(a) and at nominal speed Fig. 10(b). Thus, a purely data-driven design assure the same closed-loop cost of a standard model-based procedure, even if no assumption on the model structure is required.

V. Conclusion

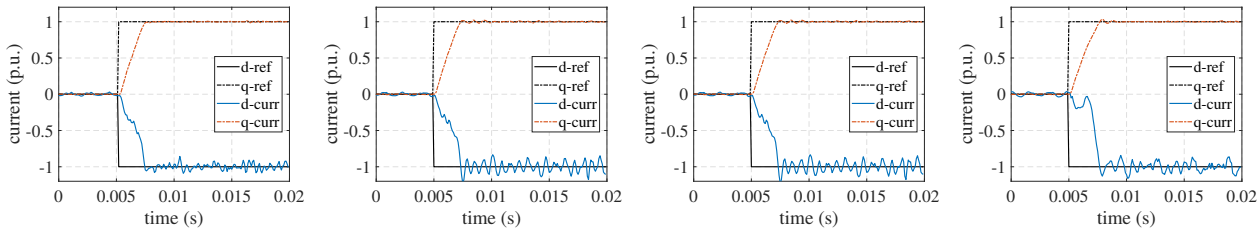
In this work we present a transition path from model-based to data-driven design of PMSM current controllers. Different data-driven algorithms are considered: the prediction error method model predictive control, the subspace and the data-enabled predictive controls (DeePC). All the algorithms were online implemented in the unconstrained version, proving their online feasibility. Similar accuracy between model-based and data-driven predictors is demonstrated with experimental data. Experimental results show that all these controllers have comparable performance, considering the MPC with an accurate model as benchmark. Moreover, among data-driven controllers, the DeePC performs well both in steady state and dynamics.

There are several challenges to address in the future. First, a comparison between data-driven and self-commissioning techniques would be valuable. This could help to design effective strategies for the excitation voltage signals. Second, the extension of data-driven methods for nonlinear system is at the beginning. The possibility to automatically include the motor nonlinearities in the control law, e.g. magnetic cross saturation effects, is of particular interest. Third, finding computationally efficient methods for implementing high-dimensional data-driven methods that include constraints in real-time is still an open challenge. Finally, other future research will focus on online adaptation of data-driven control structures and applications to other drives problems.



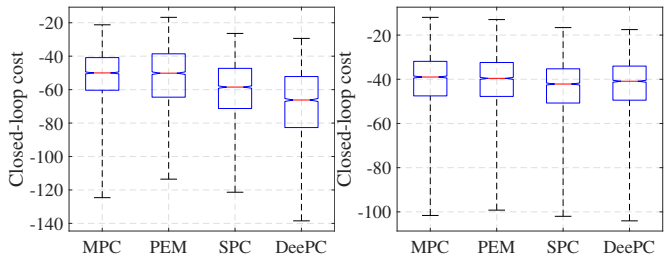
(a) MPC (b) PEM-MPC (c) SPC (d) DeePC

Fig. 8: Comparison of the step responses of model-based and data-driven current controllers at standstill.



(a) MPC (b) PEM-MPC (c) SPC (d) DeePC

Fig. 9: Comparison of the step responses of model-based and data-driven current controllers at nominal speed.



(a) Standstill (b) Nominal speed

Fig. 10: Closed-loop cost analysis at steady state.

REFERENCES

[1] Z. S. Hou and Z. Wang, "From model-based control to data-driven control: Survey, classification and perspective," *Information Sciences*, vol. 235, pp. 3–35, 2013.

[2] S. Vazquez, J. Rodriguez, M. Rivera, L. G. Franquelo, and M. Norambuena, "Model predictive control for power converters and drives: Advances and trends," *IEEE Trans. Ind. Electron.*, vol. 64, no. 2, pp. 935–947, Feb 2017.

[3] S. Hanke, O. Wallscheid, and J. Böcker, "Continuous-control-set model predictive control with integrated modulator in permanent magnet synchronous motor applications," in *Int. Electric Machines Drives Conf.*, 2019.

[4] S. Bolognani, R. Kennel, S. Kuehl, and G. Paccagnella, "Speed and current model predictive control of an IPM synchronous motor drive," in *Int. Electric Machines Drives Conf. (IEMDC)*, 2011.

[5] A. Favato, P. G. Carlet, F. Toso, and S. Bolognani, "A model predictive control for synchronous motor drive with integral action," in *44th Annu. Conf. of the IEEE Ind. Electron. Soc. (IECON)*, 2018.

[6] Q. Wang, G. Zhang, G. Wang, C. Li, and D. Xu, "Offline parameter self-learning method for general-purpose PMSM drives with estimation error compensation," *IEEE Trans. Power Electron.*, vol. 34, no. 11, pp. 11 103–11 115, 2019.

[7] A. Boglietti, A. Cavagnino, and M. Lazzari, "Experimental high-frequency parameter identification of ac electrical motors," *IEEE Trans. Ind. Appl.*, vol. 43, no. 1, pp. 23–29, 2007.

[8] S. A. Odhano, P. Pescetto, H. A. A. Awan, M. Hinkkanen, G. Pellegrino, and R. Bojoi, "Parameter identification and self-commissioning in AC motor drives: A technology status review," *IEEE Trans. Power Electron.*, vol. 34, no. 4, pp. 3603–3614, Apr. 2019.

[9] F. Wang, K. Zuo, P. Tao, and J. Rodríguez, "High performance model predictive control for PMSM by using stator current mathematical model self-regulation technique," *IEEE Trans. Power Electron.*, vol. 35, no. 12, pp. 13 652–13 662, 2020.

[10] X. Zhang, L. Zhang, and Y. Zhang, "Model predictive current control for PMSM drives with parameter robustness improvement," *IEEE Trans. Power Electron.*, vol. 34, no. 2, pp. 1645–1657, 2019.

[11] X. Liu, L. Zhou, J. Wang, X. Gao, Z. Li, and Z. Zhang, "Robust predictive current control of permanent-magnet synchronous motors with newly designed cost function," *IEEE Trans. Power Electron.*, vol. 35, no. 10, pp. 10 778–10 788, 2020.

[12] M. Yang, X. Lang, J. Long, and D. Xu, "Flux immunity robust predictive current control with incremental model and extended state observer for PMSM drive," *IEEE Trans. Power Electron.*, vol. 32, no. 12, pp. 9267–9279, 2017.

[13] M. De Soricellis, D. Da Rù, and S. Bolognani, "A robust current control based on proportional-integral observers for permanent magnet synchronous machines," *IEEE Trans. Ind. Appl.*, vol. 54, no. 2, pp. 1437–1447, 2018.

[14] S. Yin, H. Gao, and O. Kaynak, "Data-driven control and process monitoring for industrial applications - Part I," *IEEE Trans. Ind. Electron.*, vol. 61, no. 11, pp. 6356–6359, 2014.

[15] A. Brosch, S. Hanke, O. Wallscheid, and J. Böcker, "Data-driven recursive least squares estimation for model predictive current control of permanent magnet synchronous motors," *IEEE Trans. Power Electron.*, vol. 36, no. 2, pp. 2179–2190, 2021.

[16] M. Schenke, W. Kirchgässner, and O. Wallscheid, "Controller design for electrical drives by deep reinforcement learning: A proof of concept," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 7, pp. 4650–4658, 2020.

[17] Y. Zhang, J. Jin, and L. Huang, "Model-free predictive current control of PMSM drives based on extended state observer using ultralocal model," *IEEE Trans. Ind. Electron.*, vol. 68, no. 2, pp. 993–1003, 2021.

[18] F. Tinazzi, P. G. Carlet, S. Bolognani, and M. Zigliotto, "Motor parameter-free predictive current control of synchronous motors by recursive least-square self-commissioning model," *IEEE Trans. Ind. Electron.*, vol. 67, no. 11, pp. 9093–9100, 2020.

[19] S. Aghaei Hashjin, S. Pang, E. H. Miliani, K. Ait-Abderrahim, and B. Nahid-Mobarakeh, "Data-driven model-free adaptive current control of a wound rotor synchronous machine drive system," *IEEE Trans. Transport. Electrific.*, vol. 6, no. 3, pp. 1146–1156, 2020.

[20] N. A. Losic and L. D. Varga, "A current-free and parameter-free control algorithm," *IEEE Trans. Ind. Appl.*, vol. 30, no. 2, pp. 324–332, 1994.

[21] J. B. Jørgensen, J. K. Huusom, and J. B. Rawlings, "Finite horizon MPC for systems in innovation form," in *50th IEEE Conf. Decision and Control and Eur. Control Conf.*, 2011.

[22] W. Favoreel, B. De Moor, and M. Gevers, "Spc: Subspace predictive control," *IFAC Proc. Volumes*, vol. 32, no. 2, pp. 4004 – 4009, 1999.
[23] J. Coulson, J. Lygeros, and F. Dörfler, "Data-enabled predictive control: In the shallows of the deepc," *18th Eur. Control Conf. (ECC)*, 2018.
[24] J. Coulson, J. Lygeros, and F. Dörfler, "Distributionally robust chance constrained data-enabled predictive control," 2020.
[25] L. Huang, J. Coulson, J. Lygeros, and F. Dörfler, "Data-enabled predictive control for grid-connected power converters," in *58th Conf. Decision and Control (CDC)*, 2019.
[26] L. Huang, J. Zhen, J. Lygeros, and F. Dörfler, "Quadratic regularization of data-enabled predictive control: Theory and application to power converter experiments," 2020.
[27] P. G. Carlet, A. Favato, S. Bolognani, and F. Dörfler, "Data-driven predictive current control for synchronous motor drives," in *IEEE Energy Convers. Congr. and Expo. (ECCE)*, 2020.
[28] C. De Persis and P. Tesi, "Formulas for data-driven control: Stabilization, optimality, and robustness," *IEEE Transactions on Automatic Control*, vol. 65, no. 3, pp. 909–924, 2020.
[29] G. Cimini, D. Bernardini, S. Levijoki, and A. Bemporad, "Embedded model predictive control with certified real-time optimization for synchronous motors," *IEEE Trans. Control Syst. Technol.*, pp. 1–8, 2020.
[30] F. Toso, P. G. Carlet, A. Favato, and S. Bolognani, "On-line continuous control set MPC for PMSM drives current loops at high sampling rate using qpOASES," in *IEEE Energy Convers. Congr. and Expo. (ECCE)*, 2019.
[31] G. Pannocchia, M. Gabiccini, and A. Artoni, "Offset-free MPC explained: novelties, subtleties, and applications," *IFAC-PapersOnLine*, vol. 48, no. 23, pp. 342 – 351, 2015.
[32] J. Willems, P. Rapisarda, I. Markovsky, and B. De Moor, "A note on persistency of excitation," *Syst. Control Lett.*, vol. 54, Apr. 2005.
[33] B. Huang and R. Kadali, *Dynamic Modeling, Predictive Control and Performance Monitoring: A Data-driven Subspace Approach*. Springer-Verlag London, 2008.
[34] F. Dörfler, J. Coulson, and I. Markovsky, "Bridging direct & indirect data-driven control formulations via regularizations and relaxations," 2021.
[35] A. Varatharajan, P. Pescetto, and G. Pellegrino, "Sensorless self-commissioning of synchronous reluctance machine with rotor self-locking mechanism," in *2019 IEEE Energy Conversion Congress and Exposition (ECCE)*, 2019, pp. 812–817.
[36] R. Kadali, B. Huang, and A. Rossiter, "A data driven subspace approach to predictive controller design," *Control Engineering Practice*, vol. 11, no. 3, pp. 261–278, 2003.