Data Set Description: Identifying the Physics Behind an Electric Motor – Data-Driven Learning of the Electrical Behavior (Part II)

Sören Hanke*, Oliver Wallscheid, Joachim Böcker
Department of Power Electronics and Electrical Drives, Paderborn University, 33095 Paderborn, Germany,
*E-mail: hanke@lea.upb.de

Abstract—A data set was recorded to evaluate different methods for extracting mathematical models for a three-phase permanent magnet synchronous motor (PMSM) and a two-level IGBT inverter from measurement data. It consists of approximately 40 million multidimensional samples from a defined operating range of the drive. This document describes how to use the published data set [1] and how to extract models using introductory examples. The examples are based on known ordinary differential equations, the least squares method or on (deep) machine learning methods. The extracted models are used for the prediction of system states in a model predictive control (MPC) environment of the drive. In case of model deviations, the performance utilizing MPC remains below its potential. This is the case for state-of-the-art white-box models that are based only on nominal drive parameters and are valid in only limited operation regions. Moreover, many parasitic effects (e.g. from the feeding inverter) are normally not covered in white-box models. In order to achieve a high control performance, it is necessary to use models that cover the motor behavior in all operating points sufficiently well.

I. PRELIMINARY REMARK

The description of the data set consists of two parts. Part I gives a simplified introduction to the system behind the data and explains how to use the data set (https://arxiv.org/abs/2003.07273) [2]. Part II (this document) explains the system in more details, covers some basic approaches on how to extract models and discusses also a possible way to get a balanced data set where the samples are evenly distributed in a subset used for (deep) machine learning (ML) methods (https://arxiv.org/abs/2003.06268) [3].

II. INTRODUCTION

The idea of a data-driven modeling approach is to extract governing equations from measurement data. These equations can also incorporate effects that can hardly be considered in a white-box modeling approach relying only on domain-specific expert knowledge. In a permanent magnet synchronous motor (PMSM), which is often used as a traction motor in electric vehicles, such effects can be the strong magnetic saturation of the inductances, common and differential mode capacitive influences or temperature dependencies.

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

In the remainder of this paper, the drive system and its basic white-box modeling will be explained first (Sec. III). In Sec. IV the usage of the models within the model predictive control (MPC) is explained. Basic approaches for the extraction of models from data, including introductory examples, are discussed in Sec. V. With this basic understanding of the drive system, the recording and the data driven model extraction the characteristics of the data set are explained and analyzed (Sec. VI).

III. DRIVE SYSTEM

The structure of the drive system including the FCS-MPC with the used prediction models is shown in Fig. 1. The ordinary differential equations (ODE) of the PMSM in the rotor-flux oriented dq-system are given by (first principle model):

$$u_{dq} = R_s i_{dq} + \omega_{el} J \psi_{dq} + \frac{d}{dt} \psi_{dq},$$

$$J = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix},$$

$$\psi_{dq} = \begin{bmatrix} \psi_d(i_d, i_q, \varepsilon_{el}) \\ \psi_q(i_d, i_q, \varepsilon_{el}) \end{bmatrix}.$$  \hspace{1cm} (1)

Above, $u_{dq} = [u_d u_q]^T$ is the stator voltage, $R_s$ is the stator resistance, $i_{dq}$ is the stator current, $\omega_{el}$ is the electrical angular frequency, $\psi_{dq}$ is the flux linkage, and $\varepsilon_{el}$ is the electrical rotation angle of the PMSM. The flux linkage depends on the currents and the rotation angle to account for magnetic
saturation effects and effects like cogging torque due to the mechanical construction of the machine’s rotor and stator.

The dq-coordinate system is a typically used coordinate system for a mathematical modeling in the motor control domain. Modeled in this system, the PMSM is very similar to the classical DC motor where the control is rather simple to realize. Thus proven control concepts of the DC motor could easily be used as a basis for a PMSM controller. However, the dq-system variables resulting from the state transformation of the physical system variables cannot be measured directly in the system.

Assuming

\[
\begin{align*}
\psi_{dq} &= \begin{bmatrix} \psi_d \\ \psi_q \end{bmatrix} = \begin{bmatrix} L_d i_d + \psi_p \\ L_q i_q \end{bmatrix} = L_{dq} i_{dq} + \begin{bmatrix} 0 \\ \psi_p \end{bmatrix}, \\
L_{dq} &= \begin{bmatrix} L_d & 0 \\ 0 & L_q \end{bmatrix},
\end{align*}
\]

(2)
a basic PMSM model can be derived:

\[
\frac{d}{dt} i_{dq} = \begin{bmatrix} -\frac{R_d}{L_d} & \frac{L_d \omega_d}{L_q} \\ \frac{L_d \omega_d}{L_q} & -\frac{R_q}{L_q} \end{bmatrix} i_{dq} + \begin{bmatrix} 0 \\ -\frac{U_{DC}}{L_q} \end{bmatrix} v_n + \begin{bmatrix} \psi_p \omega_d \\ 0 \end{bmatrix}. \tag{3}
\]

Above, \( L_{dq} \) is the inductance matrix and \( \psi_p \) is the permanent magnet flux linkage. In this basic first principle ODE model, saturation effects and angle dependencies of the flux are neglected. Moreover, parasitic effects such as inductive and capacitive influences of the cabling between inverter and motor or motor-specific construction asymmetries are not covered since those cannot be easily introduced to the white-box model.

The voltage \( u_{dq} \) is supplied to the motor by the 2-level voltage source inverter and can be mathematically expressed by

\[
u_{dq} = Q(\varepsilon_{el}) \frac{U_{DC}}{3} \begin{bmatrix} 1 & -\frac{1}{\sqrt{3}} & -\frac{1}{\sqrt{3}} \\ 0 & \frac{\sqrt{3}}{2} & -\frac{\sqrt{3}}{2} \end{bmatrix} \begin{bmatrix} s_{a,n} \\ s_{b,n} \\ s_{c,n} \end{bmatrix}, \tag{4}
\]

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

Here, \( Q \) is the rotation matrix, \( U_{DC} \) is the DC-link voltage and \( v_n \) the vector comprising the switching state of each phase for the eight elementary vectors of the 2-level inverter:

\[
v_n = \begin{bmatrix} s_{a,n} & s_{b,n} & s_{c,n} \end{bmatrix}^T, \ n = 1, \ldots, 8 \tag{5}
\]

Each elementary vector \( v_n \) defines an autonomous system with its three switching states \( s_{a,n}, s_{b,n} \), and \( s_{c,n} \) (Tab. I). The index \( n \) denotes the corresponding autonomous system.

| n | \( s_{a,n} \) | \( s_{b,n} \) | \( s_{c,n} \) |
|---|---|---|---|
| 1 | -1 | -1 | -1 |
| 2 | +1 | -1 | -1 |
| 3 | +1 | +1 | -1 |
| 4 | -1 | +1 | -1 |
| 5 | -1 | +1 | +1 |
| 6 | -1 | -1 | +1 |
| 7 | +1 | -1 | +1 |
| 8 | +1 | +1 | +1 |

To include the switching state information directly as part of the plant model, the basic model of the PMSM can also be expressed in the form given by (5) with the system matrix \( A_n \). Here, the state vector \( x \) includes the stator currents, the sine and cosine of the rotor angle and a constant value. The constant value is required to include the induced voltage term of the q-current equation into the matrix as well.

Since the elementary vectors \( v_1 \) and \( v_8 \) lead to the same system matrix (both applying zero voltage to the motor), it is sufficient to consider only one of these two vectors in the following. Therefore, the data set presented in the following will only contain samples for the vectors \( v_1 \) to \( v_7 \).

IV. Finite-Control-set Model Predictive Control

To solve an optimal control problem on a receding prediction horizon is the basic concept of the model predictive control. In each controller cycle, a mathematical model of the plant in conjunction with a cost function is used to find an optimal sequence of the actuating variables that minimizes the costs. Applying the first element of the sequence to the actuating variables that minimizes the costs. Applying the first element of the sequence to the actuating variables that minimizes the costs. Applying the first element of the sequence to the actuating variables that minimizes the costs. Applying the first element of the sequence to the actuating variables that minimizes the costs.
Since the controller is implemented on a digital hardware, the controller computations must be performed in a discrete-time manner, which implies the need for discretization if the models to be used are based on ODEs.

The finite-control-set model predictive control (FCS-MPC) selects the actuating variables among the eight possible autonomous systems \( n \) which are defined by the elementary vectors of the inverter. Fig. 2 shows an arbitrary curve shape of the dq-currents when using an FCS-MPC and highlights the measurements carried out at each control cycle.

![FCS-MPC Diagram](image)

Fig. 2. FCS-MPC: arbitrary curve shape with highlighted measurements

Since the FCS-MPC yields constant inputs within each controller cycle, a linear FCS-MPC predicts the future system states for the time point \((k+1)T_s\) based on a discrete-time model. In the context of (deep) machine learning (ML) methods (like Artificial Neural Networks (ANNs), Gradient-Boosting-Machines (GBM) or decision trees), usually nonlinear models (7) are used for an approximation of the plant behavior. The states are comprised in the vector \( x \).

\[
\hat{x}_{k+1} = f_n(x_k)
\]

With least squares (LS) methods, often linear discrete-time models (8) are used for the regression. Here, the models for the drive’s autonomous systems are denoted as \( K_n \).

\[
\hat{x}_{k+1} = K_n x_k
\]

In comparison, nonlinear models provide a higher number of degrees of freedom and, thus, allow a more precise approximation of effects like saturation. However, there is always a trade-off between prediction accuracy and computational complexity of a given model topology.

The main interest is in the prediction of the two currents \( i_d \) and \( i_q \) which are part of the defined system states \( x \). The increment in the rotation angle \( \epsilon_{el} \) can be calculated easily, as the rather slow varying rotational speed \( \omega_{el} \) is tracked by an phase locked loop (PLL) and the time increment is known to be \( T_s \). The latter is assumed to be constant during all experiments i.e. the FCS-MPC is operated at a fixed controller cycling time.

V. PREDICTION MODELS FOR FINITE-CONTROL-SET MODEL PREDICTIVE CONTROL

The plant models can be obtained by different approaches. Some of them which will be also described in the following are:

a) discretization of the white box ODE-based models (Sec. III),

b) extraction from measurement data by using a least squares (LS) method,

c) extraction from measurement data by (deep) machine learning (ML) methods.

An explanation on how to evaluate the accuracy of obtained models including an appropriate cost function is given in Part I of the data set description.

**Approach a) Discretization of ODE-based models**

The usage of the basic plant model for approach a) results in a discretization of the continuous-time ODE system (6). The discrete-time form is obtained by using the transition matrix \( \Phi_n \) with a fixed time increment of \( T_s \):

\[
\begin{aligned}
    x((k+1)T_s) & = \Phi_n(T_s) x(kT_s). \\
    \hat{x}_{k+1} & = \Phi_n(T_s) \hat{x}_k
\end{aligned}
\]

The transition matrix can be calculated with a series expansion, also known as matrix exponential:

\[
\Phi_n(\Delta t) = e^{A_n \Delta t} = \sum_{\nu=0}^{\infty} \frac{(A_n \Delta t)^\nu}{\nu!} = I + A_n \Delta t + \frac{(A_n \Delta t)^2}{2!} + \cdots
\]

Usually, the linear discrete-time FCS-MPC models \( K_n \) built according to approach a) are an approximation of the transition matrix \( \Phi_n(T_s) \) using the series expansion up to the 1st-order term:

\[
K_n = I + A_n T_s \approx \Phi_n(T_s).
\]

However, this type of discretization assumes constant parameters in the white-box model \( A_n \), as they result from the simplification (2). The more general model (11) can also be expressed as autonomous systems without the need for additional elements in \( x \). But then the elements of the matrices \( A_n \) are dependent on the dq-currents and the rotor angle, resulting in a parameter-variant system. The discretization itself then also depends on these parameters and the series expansion (10) or its approximation have to be recalculated for each controller cycle, which also results in an increased computational burden.

**Approach b) Least-squares-based models**

The extraction of the matrices \( K_n \) from data is one solution to account for the parameter-variant characteristic of the system. Here, also effects which are not or only partially considered
in a mathematical white-box model can be covered by the
data-driven approaches b) and c).

Using the least squares (LS) approach, a multiple linear
regression can be conducted. During operation of the drive,
the vector \( x \) is measured at \( kT_s \) and \( (k+1)T_s \) and the
used elementary vector \( \varepsilon \), between these points in time
is known. Thus, with measurements that reflect the behavior
of the autonomous system \( n \), the vector \( w_{k,n} \) represents the
regressors for the least squares method and the vector \( y_{k+1,n} \)
comprises the values to be predicted by the searched model
\( K_n \):

\[
w_{k,n} = \begin{bmatrix} i_{d,k} & i_{q,k} & \sin(\varepsilon_{el,k}) & \cos(\varepsilon_{el,k}) \end{bmatrix}^T, \\
y_{k+1,n} = \begin{bmatrix} i_{d,k+1} & i_{q,k+1} \end{bmatrix}^T.
\]

(12)

Afterwards, data matrices \( X_{k,n} \) and \( X_{k+1,n} \) can be built
with \( j \) corresponding pairs for each autonomous system \( n \):

\[
W_{k,n} = \begin{bmatrix} w_{k,n,1} & w_{k,n,2} & \ldots & w_{k,n,j} \end{bmatrix}, \\
Y_{k+1,n} = \begin{bmatrix} y_{k+1,n,1} & y_{k+1,n,2} & \ldots & y_{k+1,n,j} \end{bmatrix}.
\]

(13)

Assuming a sufficiently large set of independent measure-
ments, this leads to an overdetermined system of equations
from which the matrix \( K_n \) is calculated, with \((\cdot)^{+}\) denoting
the pseudo inverse of a matrix:

\[
K_n = Y_{k+1,n}W_{k,n}^T \cdot (W_{k,n}W_{k,n}^T)^{+}.
\]

(14)

Using pairs where \( w_{k,n} \) is within a defined neighborhood
of an operating point results in a prediction model \( K_n \) that
takes parasitic effects like flux harmonics, inverter nonlinearity
or measurement offsets at this operating point implicitly into
account. To use these models in an FCS-MPC, the prediction
models would have to be calculated for different operating
points and then stored in the controller.

One possible approach to avoid the calculation of different
models while still considering the parameter variants of the
system, is to extend the vector \( x \) by further observations or
regressors. Therefore, it should be pointed out that the regres-
sor configuration of (12) is only an example of one possible
LS-setup. However, finding suitable further regressors in the
LS framework is not a straightforward way and, therefore, a
comprehensive feature engineering should be carried out during
the pre-processing.

Among other, the SINDy Toolbox can be used to find
and analyze additional regressors from a library of possible
combinations and functions of the measured quantities \( i_d, i_q \)
and \( \varepsilon_{el} \).

**Approach c) Machine-learning-based models**

The behavior of the plant can also be extracted from data by
(deep) machine learning (ML) methods.

For example, supervised learning of artificial neural net-
works (ANN) can be used for mapping the observations at
\( kT_s \) to the ones at \( (k+1)T_s \). Later on, these networks
are implemented online and used for the prediction of the system
states. The number of units in the input and output
layer is defined by the number of supplied observations. The
number of hidden layers, the number of neurons per layer,
the activation functions and the overall network topology (e.g.
feedforward, convolutional, recurrent, ...) are so-called hyper-
parameters which are the higher level degrees of freedom.

For a basic ANN the same observations \( x_{k,n} \) as for the
LS (12) can be used as input to the network. However, the
constant observation can be omitted. For the output layer,
the two predictions \( i_{d,k+1} \) and \( i_{q,k+1} \) are sufficient as targets.
Similar to LS, a feature engineering pre-processing may also
increase the prediction accuracy.

But also other methods from the domain of machine learn-
ing, like Gradient-Boosting-Machines (GBM) or decision trees
may be promising approaches.

Especially with ML methods, it is simple to include the
information about the vector which was used in the interval
before \( (n_{k-1}) \) as an input. This might be helpful to con-
sider more detailed effects like the inverter-deadtime or the
interlocking time, as they appear when switching between
elementary vectors.

**VI. DATA SET**

For the comparison of the different modeling approaches, a
data set including measurements at different operating points is
recorded. This data set consists of approx. 40 million samples
from a defined operating range of the drive.

A sample in the data set (each row) consists of the measured
dq-currents at two consecutive time points (e.g. \( k \) and \( k+1 \)),
the angle at the earlier of the two time points, and the infor-
mation about the elementary vector selected in the controller
cycle between them \( (n_{k}) \) as well as the vector selected in the
cycle before \( (n_{k-1}) \). An overview of the included variables is
given in Tab. [III]. However, the successive rows or samples in
the set do not constitute a time series.

**TABLE II**

| Variable | Description | Data type | Classification |
|----------|-------------|-----------|----------------|
| \( i_{d,k} \) | measured d-current at \( k \) | single | inputs |
| \( i_{q,k} \) | measured q-current at \( k \) | single | inputs |
| \( \varepsilon \) | measured rotational angle at \( k \) | single | inputs |
| \( n_k \) | element, vector applied at \( k \) | integer | targets |
| \( n_{k-1} \) | element, vector applied at \( k-1 \) | integer | targets |
| \( i_{d,k+1} \) | measured d-current at \( k+1 \) | single | targets |
| \( i_{q,k+1} \) | measured q-current at \( k+1 \) | single | targets |

As a result of the measurements at \( k \) and \( k+1 \), the real
behavior of the currents for a given vector is known. This
knowledge can now be used to derive models.

The drive system under test consists of an interior magnet
permanent magnet synchronous motor (IPMSM) of 57 kW
and a 2-level IGBT inverter. The most important test bench
parameters are summarized in Tab. [III]. Fig. [E] shows the test
bench with the transient recorder in the front and the used
motor in the background.

The rotational speed of the motor and the DC link voltage
for all samples were constant at \( n_{me} = 1000 \text{ min}^{-1} \) and
TABLE III

| TEST BENCH PARAMETERS |
|-----------------------|
| **DC Power supply**   | Gustav Klein |
| DC output             | galvanically isolated |
| Max. apparent power   | $S_{\text{max}}$ | 200 kW |
| Max. DC current       | $I_{\text{DC, max}}$ | 500 A |
| Variable DC output voltage | $U_{\text{DC}}$ | 6-600 V |

| IPMSM                  |
|-----------------------|
| Stator resistance     | $R_s$ | 18 m$\Omega$ |
| Inductance in d-direction | $L_d$ | 370 $\mu$H |
| Inductance in q-direction | $L_q$ | 1200 $\mu$H |
| Permanent magnet flux  | $\psi_p$ | 66 mV s |
| Pole pair number      | $p$ | 3 |
| Rated mechanical power | $P_{\text{me}}$ | 57 kW |
| Rated torque          | $M$ | 130 Nm |
| Max. stator current in dq-system | $|i_{\text{dq}}|_{\text{max}}$ | 240 A |

| Inverter              |
|-----------------------|
| 3×SKiiP 122GBT120-4D  |
| Typology              | voltage source inverter |
| Max. phase current    | $I_{C, \text{max}}$ | 1200 A |

| Controller hardware   |
|-----------------------|
| Processor board       | DS1006MC, 4 cores, 2.8 GHz |
| FPGA board            | DS2035, Xilinx Virtex-5 |
| ADC board             | DS2004, 16 channel, 16 bit |
| PWM board             | DS5101 |
| CAN board             | DS4302 |
| Digital I/O board     | DS4003 |
| Incremental encoder board | DS3002 |

| Measurement devices   |
|-----------------------|
| Transient recorder    | Yokogawa DL850 |
| Power analyzer        | Yokogawa WT3000 |
| Current probes        | 4×Danfysik, 700 A, 100 kHz |
| (all zero-flux transducers) | 3×Yokogawa, 500 A, 2 MHz |
| Torque sensors        | HBM, T10FS, 2 kN m |

$U_{\text{DC}} = 300$ V. Hence, these variables are not part of the data set. In the future, an extended data set for varying rotational motor speeds may be added and then the rotational speed would be added to the input space. Similar, the motor temperature was nearly constant during all measurements and, therefore, does not need to be considered in the given data set. The parameters that are specific for this data set are summarized in Tab. IV.

Besides the constant variables, the operating range is defined by a variation of the $i_{\text{dq}}$ currents within the shown quadrant of the dq-plane (Fig. 4). For this motor, the maximum allowed length of the $i_{\text{dq}} = [i_d, i_q]^T$ current vector is 240 A. The value of the rotor angle $\varepsilon_{el}$ ranges from $-\pi$ to $\pi$.

\[ i_d \in [-240 A, 240 A] \]

Fig. 4. Defined current operating range in the dq-plane

In order to record measurements from the entire operating range, a sequence of 478 different $i_{\text{dq}}$ set points was commanded to the FCS-MPC. At each resulting $i_{\text{dq}}$ operating point, a few seconds of measurement data were then recorded. To capture the behavior for an extensive combination of $v_n$ and $\varepsilon_{el}$ at a given $i_{\text{dq}}$ operating point, some of the elementary vectors selected by the MPC were replaced by randomly selected vectors.

A. Balancing of the data set

If a model is trained for the whole operating range, a homogeneous distribution of the measurements in the data set over the operating range of $i_d, i_q$ and $\varepsilon_{el}$ is important. This ensures that the model is not biased towards regions in the operating range where the sample concentration is higher resulting in a reduced accuracy in regions with a minor sample concentration [5]. This generally applies to all kinds of models that are learned or built on the basis of data and are intended to cover the entire operating range.

One method to obtain a balanced data set is described in the following. First, the operating range can be divided into classes by a grid. The grid step size and the operating range that are used for this example are summarized in Tab. V. The number of samples per class and, thus, the balance of the data set can then be analyzed.

For the dq-plane, the grid is shown in Fig. 5. Valid classes are all classes that are fully or partially within the specified current operating range, they are shaded red. This pattern continues for the range of the rotor angle as shown in Fig. 6.

![Fig. 3. Test bench with the used PMSM in the background](image-url)
Table V
GRID STEP SIZES AND OPERATING RANGE

| Dimension    | Operating range          | Grid step size |
|--------------|--------------------------|----------------|
| d-current    | -240 A to 0 A           | 10 A           |
| q-current    | 0 A to -240 A           | 10 A           |
| rotation angle | -π to π               | π/18           |

Fig. 5. Classes within the defined current operating range in the dq-plane, of which 471 are valid.

Furthermore, the data set is divided into subsets where each subset comprises the samples of one elementary vector \( v_n \). This provides the opportunity to extract a model for each autonomous system \( n \), as already described in Sec. V for the LS approach.

Fig. 7 shows the assignment of recorded measurements to samples, subsets and classes within the subsets. A sample consists of the dq-currents of two consecutive time points, the angle at the earlier of the two time points, and the information about the chosen elementary vector. The assignment of a sample to a subset is done by means of the elementary vector which is applied between the two time points. The class assignment of a sample is determined according to the currents and the angle at the earlier of the two points in time. For the shown blue sample, the class to which this sample belongs is determined by the vector \([i_{d,k}, i_{q,k}, \varepsilon_{el,k}]^\top\).

After the assignment of all available samples, the distribution of samples regarding the valid classes can be analyzed. This is done for each subset. Fig. 8 shows the homogeneity for each subset in dependence of the number of samples per class. As an example, if the desired number of samples per class is set to 48 samples, 99% of the valid classes meet this requirement because they contain more or exactly this desired number of samples (red marker).

Thus, limiting the number of samples to 48 in each class, leads to nearly homogenous (99 %) data set which can be used for the training of an LS or an ANN. If a class contains more than 48 samples, the remaining samples are transferred to an non-homogeneous data set that can be used to test the learned models on samples that were not utilized during learning or built-up.

Since elementary vector \( v_1 \) was selected relatively often by the controller when recording the data set, considerably more samples are available per class. The frequent selection of this so-called zero voltage vector results from the chosen value of DC link voltage in combination with the operating range, especially with the rotational speed. Even with a higher number of desired samples, there is still a high degree of homogeneity for this subset as can be seen from the blue curve.

Please note:
For sake of simplicity, \( U_{DC} \) is considered as ideally constant in this contribution. Moreover, the rotational speed \( n_{me} \) and...
the motor temperature are kept constant, too. It is planned to extend the data set to variations of this three variables in the future. However, the presented data-driven modeling ideas can be directly extended to consider these varying operation conditions by extending the input space with this additional features.

**Link to the uploaded data set:**
The data set is published on Kaggle, an online community of data scientists: https://www.kaggle.com/hankelea/system-identification-of-an-electric-motor

**REFERENCES**

[1] S. Hanke. Data set: Identifying the Physics Behind an Electric Motor - Data-Driven Learning of the Electrical Behavior. https://www.kaggle.com/hankelea/system-identification-of-an-electric-motor

[2] O. Hanke S., Wallscheid and Böcker. Data Set Description: Identifying the Physics Behind an Electric Motor – Data-Driven Learning of the Electrical Behavior (Part I). arXiv:2003.07273, 2020. https://arxiv.org/abs/2003.07273

[3] O. Hanke S., Wallscheid and Böcker. Data Set Description: Identifying the Physics Behind an Electric Motor – Data-Driven Learning of the Electrical Behavior (Part II). arXiv:2003.06268, 2020. https://arxiv.org/abs/2003.06268

[4] Steven L. Brunton, Joshua L. Proctor, and J. Nathan Kutz. Discovering governing equations from data by sparse identification of nonlinear dynamical systems. Proceedings of the National Academy of Sciences, 113(15):3932–3937, 2016.

[5] B. Mac Namee, P. Cunningham, S. Byrne, and O. I. Corrigan. The problem of bias in training data in regression problems in medical decision support. Artificial Intelligence in Medicine, 24(1):51–70, 2002.