Position and Speed Estimation of PMSMs Using Gaussian Processes

Jana Mayer, Ajit Basarur, Mariana Petrova, Fabian Sordon, Antonio Zea, and Uwe D. Hanebeck

Intelligent Sensor-Actuator-Systems Laboratory (ISAS)
Institute for Anthropomatics and Robotics
Karlsruhe Institute of Technology, Germany
jana.mayer@kit.edu, ajit.basarur@kit.edu,
mariana.petrova@student.kit.edu, fabian@sordon.com,
antonio.zea@kit.edu, and uwe.hanebeck@ieee.org

Abstract: In this paper, we present a novel low-cost technique to estimate both the position and the speed of a permanent magnet synchronous motor (PMSM) by sensing its stray magnetic field. At an optimal radial and axial distance, a low-cost magnetoresistive sensor is placed outside at the back of the PMSM. The magnetic field values are recorded for one complete rotor revolution at a resolution of less than a degree for different speeds of operation. Gaussian Processes (GPs) are employed to find a mapping function between the magnetic field values of the permanent magnet and the absolute angular positions. Then, by using the learned GP as a measurement function with an Extended Kalman Filter (EKF), both the angular position and speed of a PMSM can be estimated efficiently. Furthermore, we observe that the magnetic field depends not only on the position but also on the angular speed. To address this, we extend the GP to incorporate multivariate inputs. In order to take the periodicity of the data into account, we employ a periodic kernel for the GP. Additionally, a linear basis function model (LBFM) is introduced to incorporate more training points while maintaining the same computational cost. The GP and LBFM approaches are evaluated with data from a real PMSM experiment setup, and the accuracy of the position and speed state estimation is verified against a high-resolution optical encoder used as ground truth.

Keywords: Machine learning, estimation and filtering, experiment design, magnetic measurements, electric motor.

1. INTRODUCTION

Permanent Magnet Synchronous Motors (PMSMs) have found widespread applications owing to their high power-to-weight ratio and their availability in various compact forms. Unlike DC motors, PMSMs are not simple electromechanical systems, and require complex algorithms such as the Field Oriented Control (FOC), introduced in Lee et al. (2009), for efficient commutation. For variable speed drives, additional cascaded PI loops are used for PMSMs as mentioned in Microsemi (2012). These approaches, however, introduce important requirements: FOCs require high resolution angular position information, and PI loops demand accurate velocity information.

Solutions to these requirements need to keep cost constraints in mind. Many applications use only angular position sensors, and derive the velocity information from these. Most commonly, PMSMs are fitted with three Hall-sensors that provide low resolution angular position information. However, as a consequence of this low resolution, FOCs cannot commute efficiently, and PI loops fail to operate at low-speed, as shown in Gu et al. (2013).

Alternatives to inbuilt Hall-sensors include external magnetic resolvers, such as Ams AG (2019); Melexis (2019), and optical encoders, for example Trinamic (2019). External magnetic resolvers work by combining a magnetic piece of known characteristics together with a magnetic sensor tasked with measuring the magnetic field. The magnetic piece is usually mounted on the shaft of the motor, and the magnetic sensor is placed near the magnetic piece. Then, the angular position is derived by measuring the changes in the magnetic field. While these magnetic resolvers produce high resolution measurements, they do not offer absolute position information, because they are external to the system. Furthermore, angular velocity information needs to be calculated explicitly from position data. Another disadvantage is that magnetic resolvers might not be suitable if the end application has real space limitations because of the accompanying magnetic piece.

Alternatively, optical encoders can provide very high resolution angular data from visual information. Depending on the type of optical encoder, they can produce both relative and absolute angular positions. However, the

This work was partially supported by the German Research Foundation (DFG) under grant HA 3789/17-1 and by the Ministry of Economic Affairs, Labour, and Housing of the state Baden-Wuerttemberg, Germany.

biggest disadvantage of these sensors are their size and cost, as they are usually bulky and have to be mounted on the shaft. This may be undesirable if the end application has real space constraints and limited budget. Similarly to magnetic resolvers, the velocity information is derived from the position data, as shown in Merry et al. (2010); Pu and Wang (2012).

Approaches for estimating the angular velocity directly include the zero-crossing method introduced by Liu et al. (2017). Here, the number of times the magnetic field crosses its mean value is counted in a given period of time, and the velocity is estimated from this value. This type of velocity estimation has drawbacks similar to the optical encoder interface, as shown in Pu and Wang (2012).

In this paper, we introduce a novel approach to address the problem of angular position estimation that a) offers high resolution angular position along with angular velocity information, b) is low-cost c) has no requirement of mounting a separate magnetic piece on the shaft, and d) has small form factor to cater for the space constraints of the end application. By estimating both the angular position and velocity directly, we overcome all the mentioned drawbacks.

The key idea is to place a sensor at a suitable radial and axial distance outside of the PMSM housing, and then measure the stray magnetic field of the permanent magnets. Measurements taken with a real motor are shown in Fig. 3. The functions have a complex shape and a speed dependence is observed. Further discussions on these properties are provided in the next section. However, instead of calculating zero-crossings like Liu et al. (2017), we model the complex relationship between the angular position, the speed, and the measured magnetic field of the permanent magnets explicitly with a Gaussian Process (GP), described in Rasmussen and Williams (2006). Then, during normal operation, we employ this function with a recursive estimator to obtain both the position and the velocity efficiently.

Our paper is divided into the following sections. First, we start with the theoretical considerations of the Gaussian Process, described in Section 2. Then, Section 3 introduces the motor setup we have used for measurements and evaluation. The evaluation results and the comparison between different models are given in Section 4. Finally we conclude the paper in Section 5.

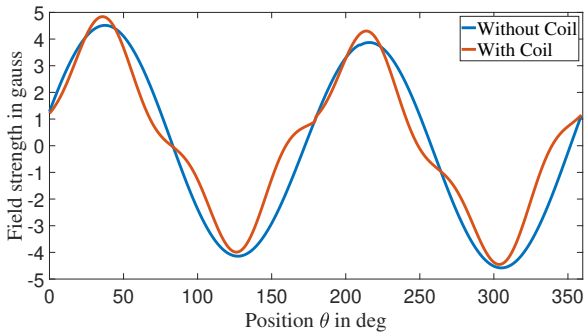


Fig. 1. Example of magnetic field strengths measured in a real motor setup, with and without coils.

2. PROPOSED APPROACH

A key component of our contribution is that we want to model the relationship between the angular position, the speed, and the measured magnetic field. This allows us to track them accurately even as they change in time. However, this task is not straightforward. In Fig. 1 we see the magnetic fields obtained at different rotor angles for two PMSM setups: one with coils (red), and one without (blue). The curve for the setup without coils is relatively simple, and could be closely approximated with a Fourier series with few coefficients, and computationally efficient algorithms such as CORDIC from Volder (1959) could be used for estimation. The coil winding profile and the motor housing produce perturbations that depend strongly on the characteristics of the specific motor setup being used, and are very difficult to describe a priori. Most importantly, the magnetic field measurements are speed dependent. Sensor properties result in a speed dependent phase shift as presented in Fig. 2. Additionally, the speed also has an effect on the shape of the function (see Fig. 3).

It is reasonable to expect hardware setups in most real applications to have artifacts and the influence of coil is one of among them. This raises the need for a mechanism to accurately describe the complex red curve for any setup.

To address this, we propose the following two-staged approach. First, we will learn the relationship between the angle, speed, and the magnetic field strength, i.e., the *measurement function*, 'offline' with a GP. To achieve this, we will use a set of reference measurements as training data. The angle and speed are assumed to be known accurately, while the strength of the magnet field is expected to be noisy. Then, during normal operation ('online'), we can use this measurement function to estimate the absolute angle of the rotor using a nonlinear recursive estimator, such as the Extended Kalman filter (EKF). Note that the measurement function only needs to be trained a single time for one specific hardware setup.

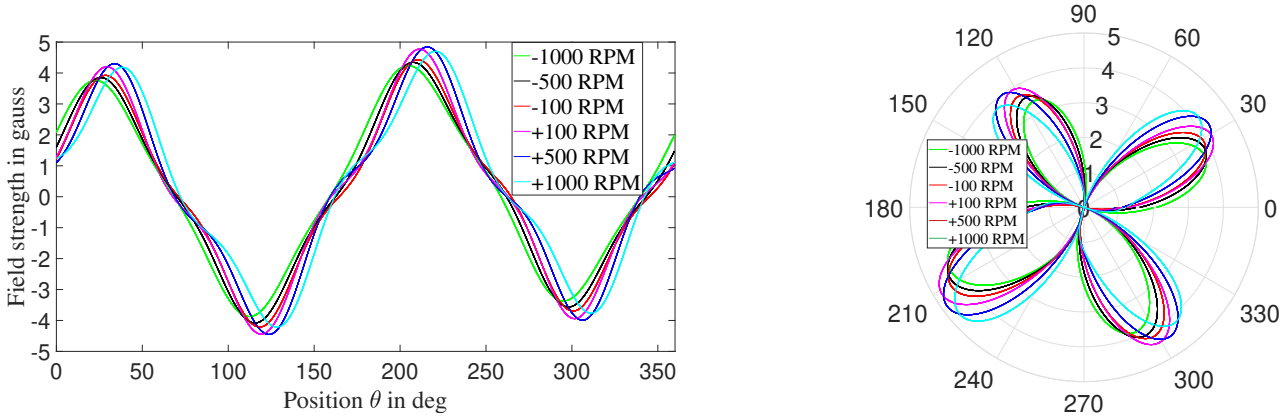
This section is structured as follows. First, we will introduce a mathematical formulation of our problem. Then, we will describe how the GP works, and propose some optimizations for it. Finally, keeping in mind applications with limited resources, we will introduce an approximation of the GP that keeps runtime constant and independent from the number of training points.

2.1 Problem Formulation

At a given timestep k, the task is to estimate the angular position θ_k and speed ω_k of the rotor based on measurements b_k from the magnetic field of the permanent magnets. The state is defined as $\underline{x}_k = [\theta_k, \omega_k]^T \in \mathbb{R}^2$. The measurement equation, which maps the state to the measured magnetic field, is determined by

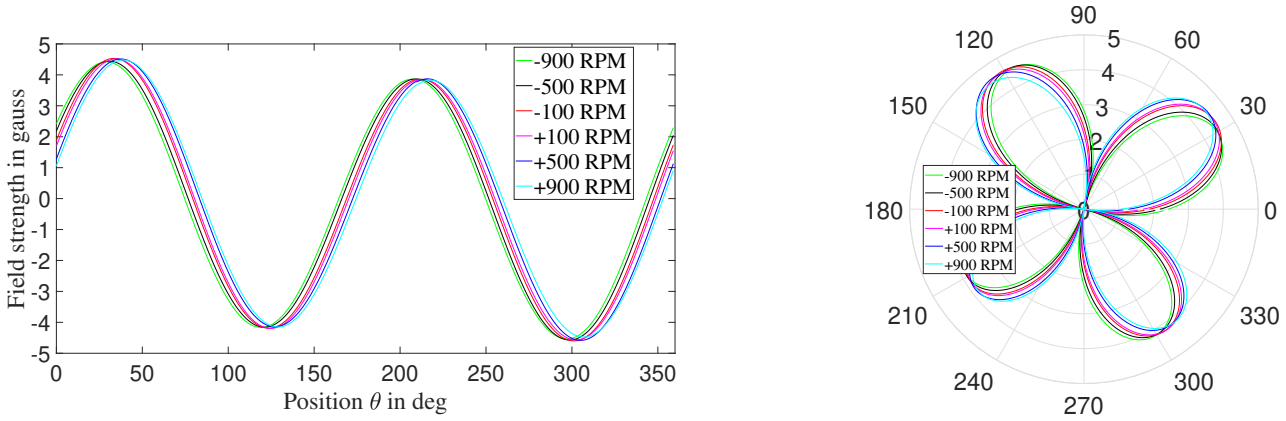
$$b_k = h\left(\underline{x}_k\right) + v_k \quad , \tag{1}$$

where $h : \mathbb{R}^2 \to \mathbb{R}$ is the measurement function, and $v_k \sim \mathcal{N}(0, \sigma_{v,k}^2)$ represents the measurement noise. If there are multiple measurements available at the same timestep, e.g. orthogonal components of the magnetic field vector $b_{k,x}$, $b_{k,y}$ and $b_{k,z}$, they can be stacked vertically. Furthermore,



(a) Cartesian Plot (b) Polar Plot

Fig. 2. Magnetic field dependency on different angular speeds of a PMSM with coils.



(a) Cartesian Plot (b) Polar Plot

Fig. 3. Magnetic field measurements of a PMSM without motor coils.

we assume that each noise term is independent from other noise terms and from the state.

The state is assumed to evolve according to the system equation

$$\underline{x}_{k+1} = \boldsymbol{A}_k \cdot \underline{x}_k + \underline{w}_k \ , \tag{2}$$

where $\underline{w}_k \sim \mathcal{N}(\underline{0}, \boldsymbol{R}_k)$ represents the system noise. For this work, we will employ a constant velocity model, i.e.,

$$\boldsymbol{A}_k = \begin{pmatrix} 1 & \Delta T_k \\ 0 & 1 \end{pmatrix} \ , \tag{3}$$

where ΔT_k represents the time change. However, the ideas presented here do not impose any constraints on the motion model, and can incorporate other terms such as acceleration, jerk, or even nonlinear components. In the following, we will drop the index k for better legibility unless needed.

2.2 Choosing an Appropriate Kernel

In order to define a GP, first we need to determine an appropriate kernel function $\kappa : \mathbb{R}^2 \times \mathbb{R}^2 \to \mathbb{R}$ which tells us how 'similar' the measurements generated by two arbitrary states are assumed to be. In the following, we will define the kernel function based on the angular position and speed. Note that, under ideal circumstances, we would expect the measured magnetic field to depend exclusively on the rotor angle. However, as shown in Fig. 2 and Fig. 3, artifacts like induction on the motor coils and sensor-related effects cause the motor speed to also have an influence on the magnetic field . In turn, this means that both the angle and the speed can be observed from the measured magnetic field.

Let $\underline{x}_*^i = \left[\theta_*^i, \omega_*^i\right]^T$ and $\underline{x}_*^j = \left[\theta_*^j, \omega_*^j\right]^T$ be two arbitrary state inputs. The angular positions θ_*^i and θ_*^j are periodic and defined on the interval $[0°, 360°)$. Thus, we choose a periodic kernel for the angle as presented in Rasmussen and Williams (2006), i.e.,

$$\kappa_\theta(\theta_*^i, \theta_*^j) = \sigma_\kappa \cdot \exp\left(-\frac{2\sin^2\left(\frac{\theta_*^i - \theta_*^j}{2}\right)}{l_\theta^2}\right) \ . \tag{4}$$

The angular speed is not periodic, and thus, the simpler squared exponential kernel can be used, leading to

$$\kappa_\omega(\omega_*^i, \omega_*^j) = \sigma_\kappa \cdot \exp\left(-\frac{2\left(\omega_*^i - \omega_*^j\right)^2}{l_\omega^2}\right) \ . \tag{5}$$

To construct a kernel for both input properties, we use the product of both kernels

$$\kappa(\underline{x}_*^i, \underline{x}_*^j) = \sigma_\kappa^2 \exp\left(-\frac{2\sin^2\left(\frac{\theta_*^i - \theta_*^j}{2}\right)}{l_\theta^2} - \frac{2\left(\omega_*^i - \omega_*^j\right)^2}{l_\omega^2}\right) \ . \tag{6}$$

The scalars σ_κ, l_θ, and l_ω are hyperparameters to be learned. In the following, we will refer to this combined kernel as the *Periodic Automatic Relevance Determination kernel* (PER-ARD).

2.3 Learning the Measurement Function

In order to train the GP, we proceed to measure the magnetic field at N different angle-speed configurations. This yields the state inputs $\boldsymbol{X} = \begin{bmatrix}\underline{x}_*^1 & \cdots & \underline{x}_*^N\end{bmatrix}$ with the corresponding measurement vector $\underline{b}_* = \begin{bmatrix}b_*^1 \cdots b_*^N\end{bmatrix}^T$, where each entry is assumed to be a realization of the measurement equation

$$b_*^i = h(\underline{x}_*^i) + v^i, \qquad v^i \sim \mathcal{N}(0, \sigma_n^2) \ , \tag{7}$$

for $1 \leq i \leq N$, where v^i is the measurement noise term, and σ_n is an hyperparameter to be learned.

For the subsequent derivations, we introduce the following extensions to the kernel function

$$\begin{aligned}\kappa(\underline{x}_*, \boldsymbol{X}) &= \begin{bmatrix}\kappa(\underline{x}_*, \underline{x}_*^1) & \kappa(\underline{x}_*, \underline{x}_*^2) & \cdots & \kappa(\underline{x}_*, \underline{x}_*^N)\end{bmatrix} \ , \\ \kappa(\boldsymbol{X}, \underline{x}_*) &= \begin{bmatrix}\kappa(\underline{x}_*^1, \underline{x}_*) & \kappa(\underline{x}_*^2, \underline{x}_*) & \cdots & \kappa(\underline{x}_*^N, \underline{x}_*)\end{bmatrix}^T \ , \text{and} \\ \boldsymbol{K}(\boldsymbol{X}, \boldsymbol{X}) &= \begin{bmatrix}\kappa(\boldsymbol{X}, \underline{x}_*^1) & \kappa(\boldsymbol{X}, \underline{x}_*^2) & \cdots & \kappa(\boldsymbol{X}, \underline{x}_*^N)\end{bmatrix} \ .\end{aligned}$$

Our goal now is to determine the mean $\hat{h}_*^t$ and variance $\sigma_{h,t}^2$ of the underlying function h_*^t that corresponds to an arbitrary test input $\underline{x}_*^t$. The joint distribution of the measurement vector $\underline{b}_*$ and the function value h_*^t is given by

$$\begin{bmatrix}\underline{b}_* \\ h_*^t\end{bmatrix} \sim \mathcal{N}\left(\underline{0}, \begin{bmatrix}\kappa(\boldsymbol{X}, \boldsymbol{X}) + \sigma_n^2 \boldsymbol{I} & \kappa(\boldsymbol{X}, \underline{x}_*^t) \\ \kappa(\underline{x}_*^t, \boldsymbol{X}) & \kappa(\underline{x}_*^t, \underline{x}_*^t)\end{bmatrix}\right) \ , \tag{8}$$

assuming the mean of the measurements to be zero. Then, from this joint distribution, a prediction for h_*^t can be given as

$$\hat{h}_*^t = \kappa(\underline{x}_*^t, \boldsymbol{X}) \cdot \boldsymbol{K'}^{-1} \cdot \underline{b}_* \ , \tag{9}$$

$$\sigma_{h,t}^2 = \kappa(\underline{x}_*^t, \underline{x}_*^t) - \kappa(\underline{x}_*^t, \boldsymbol{X}) \cdot \boldsymbol{K'}^{-1} \cdot \kappa(\boldsymbol{X}, \underline{x}_*^t) \ , \tag{10}$$

using the helper term $\boldsymbol{K'} := \boldsymbol{K}(\boldsymbol{X}, \boldsymbol{X}) + \sigma_n^2 \boldsymbol{I}$.

As a final step, the hyperparameters $\theta = [\sigma_n, \sigma_\kappa, l_\theta, l_\omega]^T$ are estimated maximizing the log marginal likelihood

$$\log p(\underline{b}_* | \boldsymbol{X}, \underline{\theta}) = -\frac{1}{2} \underline{b}_*^T \boldsymbol{K'}^{-1} \underline{b}_* - \frac{1}{2} \log |\boldsymbol{K'}| - \frac{n}{2} \log(2\pi) \tag{11}$$

as presented in Rasmussen and Williams (2006). The derivation of this estimation is omitted for brevity.

2.4 Deriving a Measurement Equation

We will now formulate the measurement function required by (1). The function h is provided by the predicted mean function $\hat{h}_*^t$. Hence we obtain from (9) that

$$h(\underline{x}_k) = \kappa(\underline{x}_k, \boldsymbol{X}) \cdot \boldsymbol{K'}^{-1} \cdot \underline{b}_* \ . \tag{12}$$

The measurement noise term $v_k \sim \mathcal{N}(0, \sigma_{v,k}^2)$ follows from

$$\sigma_{v,k}^2 = \kappa(\underline{x}_k, \underline{x}_k) - \kappa(\underline{x}_k, \boldsymbol{X}) \cdot \boldsymbol{K'}^{-1} \cdot \kappa(\boldsymbol{X}, \underline{x}_k) \ . \tag{13}$$

The derivative of the measurement function can be calculated analytically, allowing the use of efficient recursive estimators such as the EKF. For more detailed information on integrating a measurement function modeled by a Gaussian process into the EKF, we refer to Ko and Fox (2009).

2.5 Implementing an Efficient Approximation

An important concern when evaluating the measurement equation from Sec. 2.4 is computational speed, especially as the number of training state inputs N increases. We propose the following speed-ups:

- The expression $\boldsymbol{K'}^{-1} \cdot \underline{b}_*$ in (12) is independent from the state and the measurement, and thus, can be precalculated. This allows the evaluation cost of the measurement function to increase linearly to N. This term can also be reused in the derivative.
- Another time consuming operation is the matrix multiplication calculating the variance of the prediction. In the border case of infinite number of evenly distributed training points the predicted variance of the Gaussian process will converge to a constant value (the variance of the training points) for all test points. Assuming that the training state inputs are approximately evenly distributed and cover the state space in a sufficient way, we approximate the predicted variance into a state-independent form as

$$\sigma_{v,k}^2 \approx \sigma_n^2 \ . \tag{14}$$

As a last resource, the number of training input states N can be reduced, but this reduces the total amount of information being used.

Alternatively, we apply an approach that can incorporate an arbitrary number of training states while keeping the runtime constant. The idea is to find an approximation of a given measurement function (12) by employing a linear combination of M basis functions, where $M \ll N$. Interpreting the predicted mean of the Gaussian process as a linear combination of kernel functions is presented in Rasmussen and Williams (2006). We first select a list of M representative samples $\boldsymbol{X}_+ = \begin{bmatrix}\underline{x}_+^1 & \cdots & \underline{x}_+^M\end{bmatrix}$ from the state space, for example by uniform sampling. Then, using an arbitrary point $\underline{x}_+ \in \mathbb{R}^2$, we define the basis functions as

$$\psi^i(\underline{x}_+) := \kappa(\underline{x}_+, \underline{x}_+^i) \ , \tag{15}$$

for $1 \leq i \leq M$, by reusing the same kernel implemented for Sec. 2.2. Given a list of scalar weights $\underline{a} := \begin{bmatrix}a^1 & \cdots & a^M\end{bmatrix}^T$, we obtain a new measurement function

$$h(\underline{x}_+) = \begin{bmatrix}\psi^1(\underline{x}_+) & \cdots & \psi^M(\underline{x}_+)\end{bmatrix} \cdot \underline{a} \ . \tag{16}$$

The task is now to determine the weights $\underline{a}$ that best approximate the GP from Sec. 2.3 with input states $\boldsymbol{X} = \begin{bmatrix}\underline{x}_*^1 & \cdots & \underline{x}_*^N\end{bmatrix}$ and magnetic field measurements $\underline{b}_*$. We define

$$\boldsymbol{\Phi} := \begin{bmatrix}\psi^1(\underline{x}_*^1) & \ldots & \psi^M(\underline{x}_*^1) \\ \vdots & \ddots & \vdots \\ \psi^1(\underline{x}_*^N) & \ldots & \psi^M(\underline{x}_*^N)\end{bmatrix} \in \mathbb{R}^{N \times M} \ . \tag{17}$$

An optimal solution for $\underline{a}$ can then be determined by a regularized least squares solution as presented in Bishop (2006), yielding

$$\underline{a} = \underset{\underline{a}' \in \mathbb{R}^M}{\operatorname{argmin}} \ \|\underline{b}_* - \boldsymbol{\Phi} \cdot \underline{a}'\|_2^2 + \frac{\lambda}{2} \|\underline{a}'\|_2^2 \ , \tag{18}$$

where the regularization coefficient λ is added to prevent over-fitting. Setting the gradient with respect to $\underline{a}'$ to $\underline{0}$ leads to the closed form solution

$$\underline{a} = \left(\boldsymbol{\Phi}^T \boldsymbol{\Phi} + \lambda \cdot \boldsymbol{I}\right)^{-1} \cdot \boldsymbol{\Phi}^T \cdot \underline{b}_* \ . \tag{19}$$

The hyperparameters are estimated in the same manner as in the Gaussian process based on the training data. We denote this approach as the *Linear Basis Function Model* (LBFM).

3. EXPERIMENT SETUP

In 4, a schematic illustration of the experiment setup used to evaluate our proposed approach is shown. The setup consists mainly of a PMSM/BrushLess DC (BLDC) motor with a corresponding controller board. The Anisotropic MagnetoResistive (AMR) sensor is placed at the back of motor at a suitable axial and radial distance, as described in Sec. 3.1, and measures the magnetic field of the permanent magnets mounted on the rotor in both x and y directions. A high-resolution optical encoder, employed only to obtain ground-truth values of the angular position for the evaluation, is mounted on the shaft in front of motor. Based on the measured angular positions the ground truth angular speed is calculated using a numerical differentiation presented in Merry et al. (2010). The optical encoder outputs its results using the ABZ interface, while the magnetic sensor outputs analog values.

The hardware components used for the experiment setup are given in Table 1. For the software, a program was written in the prototyping platform Arduino Due to capture both the analog signals and optical encoder data at a sampling rate of 450 Hz. The used hardware setup is shown in Fig. 5.

Table 1. Components used in the experiment setup.

Component	Hardware Used
Controller Board	Nucleo-F746ZG and IHM07M1
Magnetic Sensor	HMC1052 Honeywell (2019)
Optical Encoder	TMCS-28
PMSM Motor	24 V Brushless DC Motor

In the following, we will describe the approach we used to find a suitable position of the sensor, and discuss the characteristics of the obtained measurements.

3.1 Positioning of Magnetic Sensor

In principle, the magnetic sensor can be placed anywhere on the motor, but we aim for a position that maximizes the signal quality while reducing the measurement noise. A simulated cross-section view of the magnetic field distribution of a PMSM while in operation, presented by Ishak et al. (2010), is shown in Fig. 6. The magnetic field distribution towards the center is caused by the permanent magnets of the rotor, while the distribution at the outer part of the stator is caused by the electromagnets. However, we are interested in measuring the magnetic field caused by the permanent magnets mounted on the rotor. For this reason, a suitable radial distance r from the center, and a suitable axial z distance from the motor have to be found, to place the high-bandwidth magnetic sensor.

We employed the following heuristic approach. For the radial distance, the position of the sensor depends entirely on the location of the permanent magnets and its vicinity to the electromagnetic coils. Placing the sensor between the

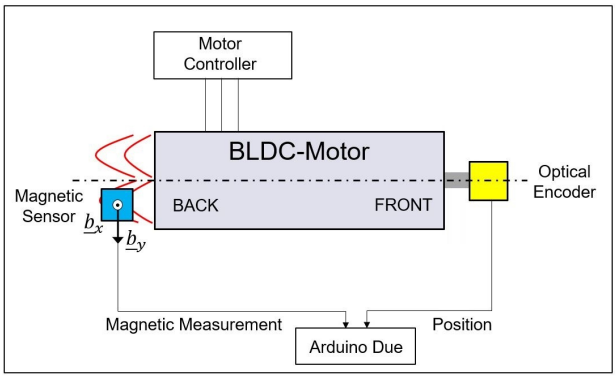


Fig. 4. Schematic illustration of the experiment setup.

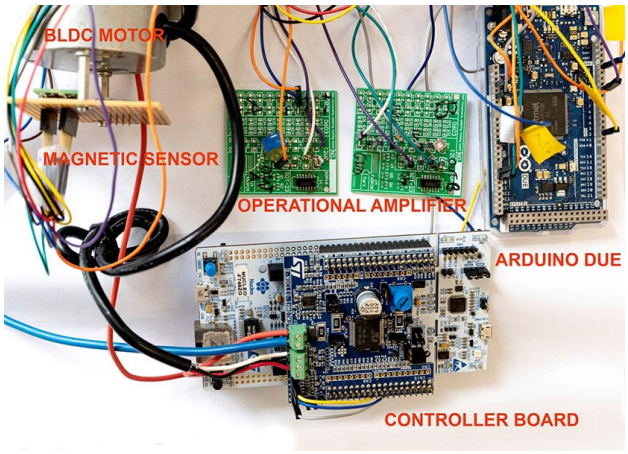


Fig. 5. Hardware setup used for magnetic field measurements of a PMSM.

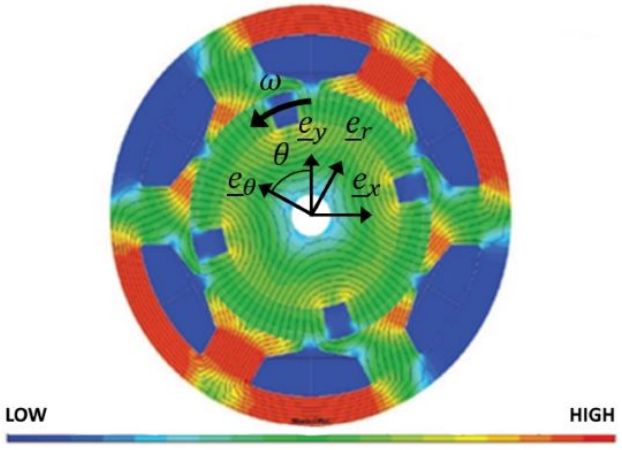


Fig. 6. The magnetic flux distribution of a typical 2-pole PMSM, taken from Ishak et al. (2010). The white center is the shaft, the green ring in the middle represents the permanent magnets, and around it are located the six electromagnets.

permanent magnets and the electromagnets (underneath the air gap of a PMSM) would increase the recorded noise. However, if the sensor is placed too close to the center of rotation axis, the signal amplitude would be smaller. Thus,

any radial position r between the permanent magnets and rotation axis along the radial distance would be satisfactory.

In case of the axial distance, the signal becomes stronger if the sensor is placed very close to the housing, but very weak and noisy if the sensor is placed far away from the housing. Therefore, the optimal axial distance depends upon the dynamic range of the chosen magnetic sensor. If the sensor has a higher dynamic range, a closer distance should be preferred.

For our setup, the determined values are $r = 10.6\,\text{mm}$ and $z = 13.0\,\text{mm}$. We adjusted the dynamic range of our magnetic sensor data using an operational amplifier to be compatible with the analog-to-digital converter of Arduino Due.

3.2 Magnetic Field Measurements

In the following, we will discuss the properties of the obtained magnetic field measurements. Taking measurements at different rotational speeds of the PMSM an unexpected behaviour of the measurements was observed as illustrated in Fig. 2. The magnetic field values show a speed dependent phase lag and shape deformation. In order to understand these artifacts, a simple setup was created with coils detached from the motor as shown in Fig. 7. In this case, the signal is approximately perfectly sinusoidal (Fig. 3), and there is no shape deformation. However, the phase shift is still observed. It could be due to the combined frequency response of the magnetic sensor and the amplifier board. The results obtained with the additional setup without coils, prove that shape deformation is attributed to the presence of coils and the housing. Fortunately, our approach does not need to model the cause of these artifacts explicitly, as both techniques introduced in Sec. 2 can describe the distorted shapes.

At every timestep, the Honeywell magnetic sensor provides two orthogonal measurements simultaneously, in the x and y directions. The values for a whole rotation of the motor are shown in Fig. 8. It is important to note that magnetic

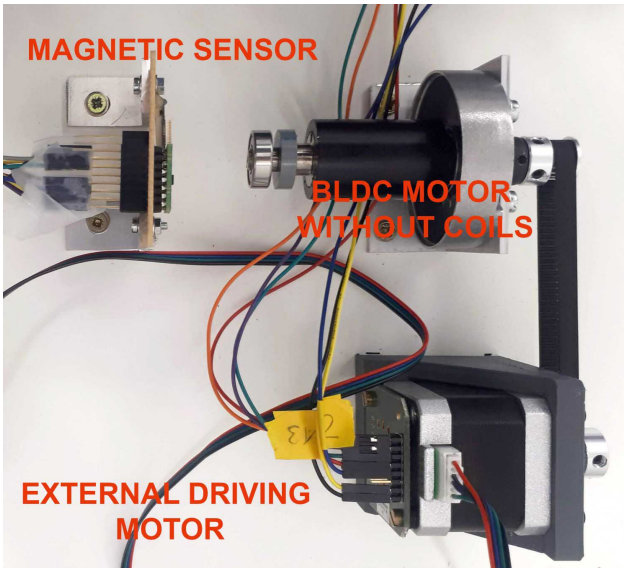


Fig. 7. Hardware setup used for magnetic field measurements of a PMSM without coils.

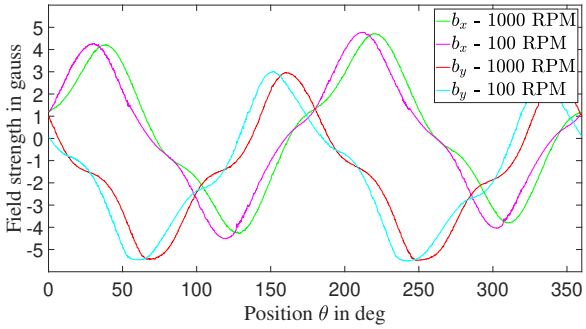


Fig. 8. Magnetic field in x and y directions from the two-axis magnetic sensor.

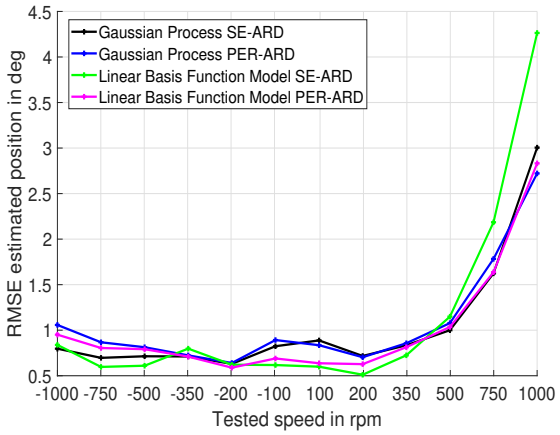
field values b_x and b_y are not exact shifted versions of each other. The reason for this is that the magnetic sensor is not placed at the center of the back surface of electric motor, but rather at a certain distance away from the center as shown in Fig 4.

Furthermore, each of the signals b_x and b_y have a period of 2π instead of π. This is clearly visible in both Fig 2 and Fig 3. This could be attributed to the asymmetric profile of the magnets being measured. Note that both characteristics are actually an advantage for the estimator, as they reduce ambiguities and provide more information for the calculation of the position and speed.

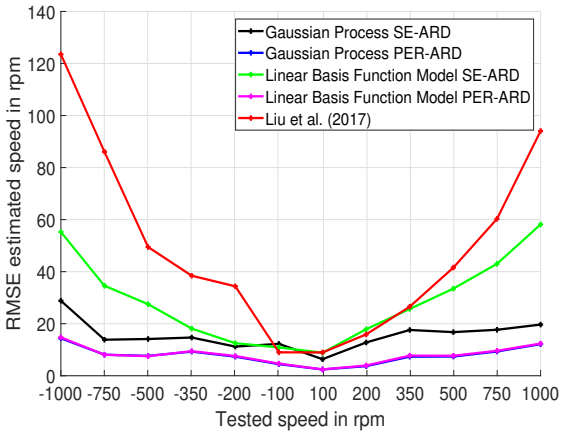
4. EVALUATION

We evaluate the proposed approach using the hardware setup introduced in Sec. 3. For the state estimation, we employ the EKF implementation provided in the Nonlinear Estimation Toolbox by J. Steinbring (2017), using ideas drawn from Ko and Fox (2009). To describe the evolution of the state, we apply a constant velocity model as introduced in (2). For the evaluation, we recorded two data sets, one for training and validation, and the other for testing. Each set contains angular position values and the corresponding magnetic field measurements consisting of two orthogonal components b_x and b_y. The angular speed is calculated based on the angular position measurements. In order to apply a zero mean Gaussian process, the magnetic field measurements are centered by subtracting the mean of the training data. The hyperparameters of the Gaussian process are optimized using the Gaussian Processes for Machine Learning (GPML) Toolbox presented in Rasmussen and Nickisch (2010).

We evaluate our approach with both regression techniques, the GP regression from Sec. 2.3, and the LBFM from Sec. 2.5. Each of them is also tested with two different kernels, namely the Squared Exponential Automatic Relevance Determination kernel (SE-ARD) presented in Duvenaud, D. (2014), and the Periodic Automatic Relevance Determination Kernel (PER-ARD) that we introduced in Sec. 2.2. In the GP, we use 1200 training points for the SE-ARD kernel, and 800 training points for the PER-ARD kernel. In the LBFM, we use 9800 training points with 1200 basis functions for the SE-ARD kernel. For the PER-ARD kernel, we use 10200 training points with 800 basis functions. The idea behind choosing a different number of training points was to make a fair comparison by keeping the execution



(a) Position estimation error



(b) Speed estimation error

Fig. 9. Position and speed estimation error for different operating motor speeds.

time of each of the model pretty much the same (in the range of 0.49 ms to 0.52 ms in MATLAB R2019a on an Intel(R) Core(TM) i7-8700 at 3.20 GHz under Windows 10). All models are trained with the data from the same training set. In the course of training, the measurement noise σ_n^2 is learned, as described in Sec. 2. After incorporating the validation data set, we also adjust the system noise matrix $\boldsymbol{R}_k$ and the regularization coefficient λ used for the LBFM. The values determined for λ, σ_n^2, and $\boldsymbol{R}_k$ are summarized in Table 2.

Table 2. Parameter values for the evaluation.

Model	σ_n^2	$\boldsymbol{R}_k$	λ
GP, SE-ARD	0.1	$\mathrm{diag}(0.008^2, 0.01^2)$	-
GP, PER-ARD	0.001	$\mathrm{diag}(0.0025^2, 0.009^2)$	-
LBFM, SE-ARD	0.1	$\mathrm{diag}(0.008^2, 0.01^2)$	10
LBFM, PER-ARD	0.001	$\mathrm{diag}(0.03^2, 0.1^2)$	1

The angular position measured with the optical encoder is utilized as ground truth. The ground truth speed is calculated by using the two previous angular positions and their corresponding time difference. For the evaluation, we determine the Root Mean Square Error (RMSE) between the estimated values (both speed and position) and the ground truth using the test data set. The results are presented in Fig. 9.

We run the EKF on a sequence with stepwise changing angular speed, and evaluate the performance separately for different speeds. For the position estimation, the LBFM with the PER-ARD kernel shows the best results across all speeds. All models perform more or less equally except for the LBFM with the SE-ARD kernel. This model and kernel combination has large position errors for higher positive speeds.

For the speed estimation, the models using the PER-ARD kernel perform clearly better compared to the models with the SE-ARD kernel. The speed estimation approach is also compared against the zero-crossings method given by Liu et al. (2017). The error of our speed estimation is bounded and it is less than 20 rpm for all the operating speeds of the motor. The error of the zero-crossing approach increases with the operating speed of the motor, and perform significantly worse at higher speeds. The reason might be the availability of few measurement points causing the zero-crossing detection to become less reliable.

Overall, GP and the LBFM with the PER-ARD kernel provided highly accurate position and speed estimation for all the covered operating speeds of the motor.

5. CONCLUSION AND FUTURE WORK

Electrical motors such as PMSMs contain permanent magnets attached to their rotors, which produce magnetic fields that can be exploited for absolute angular position and speed estimation. State-of-the-art position sensors, such as external magnetic resolvers or optical encoders, are either too bulky for setups with space limitations, or are too costly. Furthermore, they are unable to measure the speed directly. To address this, we proposed a new approach that only requires an external magnetic sensor to be placed outside of the motor, and provides highly accurate estimates by modeling the relationship between angular position, speed, and magnetic field strength using a Gaussian Process.

For our hardware setup, we used a low-cost high bandwidth anisotropic magnetoresistive sensor to record the magnetic field of the permanent magnets at different angles. However, the produced field values were not a simple sinusoidal shape, as expected, due to the inherent structure and construction of the motor itself. In order to describe these non-sinusoidal, distorted, and speed dependent signals, a GP regression technique was employed. Then, we introduced alternative model based on a linear combination of basis functions, the LBFM, to accommodate a large number of training points while maintaining low estimation runtime. Finally, in order to incorporate the periodic nature of the rotor angle, both the GP and the linear basis functions were trained with a combination of a periodic kernel for the positional angle, and a squared exponential kernel for the speed. The learned GP and LBFM measurement models were then used to estimate the angular position and speed of the motor, using an Extended Kalman Filter.

Evaluations showed that all of the models estimated the position with an accuracy of less than one degree for most of the speeds. Additionally, the models trained

with the periodic kernel performed better compared to models trained with the non-periodic kernel. The speed estimation was also better compared with the approach Liu et al. (2017). Overall, the position and speed estimation using linear basis function model with the periodic kernel fared well compared to the other implemented models in this paper. The evaluation showed that our contributions provided highly accurate estimation for all of the considered speed ranges.

As future work, we plan to test our approach under large and varying load conditions of the motor. Additionally, we plan to extend the existing constant velocity model to a constant acceleration model in order to move closer to the operating conditions of the motor.

ACKNOWLEDGEMENTS

We would like to thank Jonas Draeger, Alexander Riffel and Achim Langendoerfer for the experiment setup.

REFERENCES

Ams AG (2019). AMS AS5047U - High Resolution Rotary Position Sensor. URL `https://ams.com/as5047u`.

Bishop, C.M. (2006). *Pattern Recognition and Machine Learning*. Springer Science+Business Media.

Duvenaud, D. (2014). *Automatic Model Construction with Gaussian Processes*. Ph.D. thesis, University of Cambridge, Cambridge.

Gu, J., Ouyang, M., Li, J., Lu, D., Fang, C., and Ma, Y. (2013). Driving and Braking Control of PM Synchronous Motor Based on Low-resolution Hall Sensor for Battery Electric Vehicle. *Chinese Journal of Mechanical Engineering*, 26(1), 1–10.

Honeywell (2019). 1, 2 and 3 Axis Magnetic Sensors HMC1051/HMC1052/HMC1053. URL `https://www.sparkfun.com/datasheets/IC/HMC105X.pdf`.

Ishak, D., Manap, N.A.A., Ahmad, M.S., and Arshad, M.R. (2010). Electrically Actuated Thrusters for Autonomous Underwater Vehicle. In *Proceedings of the 2010 11th IEEE International Workshop on Advanced Motion Control (AMC)*. Nagaoka, Niigata.

J. Steinbring (2017). The Nonlinear Estimation Toolbox. URL `bitbucket.org/nonlinearestimation/toolbox`.

Ko, J. and Fox, D. (2009). GP-BayesFilters: Bayesian Filtering using Gaussian Process Prediction and Observation Models. *Autonomous Robots*, 27(1), 75–90.

Lee, S., Lemley, T., and Keohane, G. (2009). A comparison study of the commutation methods for the three-phase permanent magnet brushless dc motor. In *Proceedings of the Electrical Manufacturing Technical Conference 2009: Electrical Manufacturing and Coil Winding Expo, EMCWA*. Nashville, TN, United States.

Liu, X., Pong, P.W.T., and Liu, C. (2017). Velocity Measurement Method for PMSMs Through External Stray Magnetic Field Sensing. In *Proceedings of the 2017 IEEE SENSORS*. Glasgow, United Kingdom.

Melexis (2019). Triaxis Magnetic Resolvers. URL `https://www.melexis.com`.

Merry, R., van de Molengraft, M., and Steinbuch, M. (2010). Velocity and Acceleration Estimation for Optical Incremental Encoders. *Mechatronics*, 20(1), 20–26.

Microsemi (2012). Field Oriented Control of Permanent Magnet Synchronous Motors - User's Guide.

Pu, J.t. and Wang, H. (2012). A Novel Variable M/T Method for Speed Measurement with High Precision in a Wide Speed Range. In *Proceedings of 2nd International Conference on Electronic & Mechanical Engineering and Information Technology*. Shenyang, China.

Rasmussen, C.E. and Williams, C.K.I. (2006). *Gaussian Processes For Machine Learning*. MIT Press.

Rasmussen, C.E. and Nickisch, H. (2010). Gaussian Processes for Machine Learning (GPML) Toolbox. *Journal of Machine Learning Research*, 11, 3011–3015.

Trinamic (2019). Trinamic Optical Encoder TMCS-28. URL `https://www.trinamic.com`.

Volder, J.E. (1959). The CORDIC Trigonometric Computing Technique. *IRE Transactions on Electronic Computers*, EC-8(3), 330–334.