IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2017.DOI

Bayesian Cramér-Rao Lower Bound for Magnetic Field-based Localization

BENJAMIN SIEBLER[1], (Member, IEEE), STEPHAN SAND[1], (Senior Member, IEEE), AND UWE D. HANEBECK[2], (Fellow, IEEE)

[1]German Aerospace Center (DLR), Institute of Communications and Navigation, 82234 Oberpfaffenhofen, Germany
[2]Karlsruhe Institute of Technology (KIT), Intelligent Sensor-Actuator-Systems Laboratory, 76131 Karlsruhe, Germany

Corresponding author: Benjamin Siebler (e-mail: benjamin.siebler@dlr.de).

This work is carried out in the Ubiquitous Spatio-Temporal Learning for Future Mobility (ULearn4Mobility) project funded by the Helmholtz AI Cooperation Unit under the grant ZT-I-PF-5-49.

ABSTRACT In this paper, we show how to analyze the achievable position accuracy of magnetic localization based on Bayesian Cramér-Rao lower bounds and how to account for deterministic inputs in the bound. The derivation of the bound requires an analytical model, e.g., a map or database, that links the position that is to be estimated to the corresponding magnetic field value. Unfortunately, finding an analytical model from the laws of physics in not feasible due to the complexity of the involved differential equations and the required knowledge about the environment. In this paper, we therefore use a Gaussian process (GP) that approximates the true analytical model based on training data. The GP ensures a smooth, differentiable likelihood and allows a strict Bayesian treatment of the estimation problem.
Based on a novel set of measurements recorded in an indoor environment, the bound is evaluated for different sensor heights and is compared to the mean squared error of a particle filter. Furthermore, the bound is calculated for the case when only the magnetic magnitude is used for positioning and the case when the whole vector field is considered. For both cases, the resulting position bound is below 10 cm indicating an high potential accuracy of magnetic localization.

INDEX TERMS Bayesian Cramér-Rao lower bound, finger-printing, Gaussian process, indoor localization, magnetic field-based localization, particle filter

I. INTRODUCTION

MAGNETIC localization uses position-dependent distortions of the Earth magnetic field. Distortions are caused by ferromagnetic material in the environment and are persistent over time as long the environment remains the same. The existence and the usefulness of such distortions have been shown for different environments and use cases. The majority of the literature published in this area is concerned with indoor localization of pedestrians and robots [1]–[9], where the distortions are due to steel in the walls, floors, and in furniture. For indoor environments, magnetic localization is often combined with an inertial navigation system or an odometer to improve the accuracy and reliability. Besides indoor environments, magnetic distortions were shown to enable localization also along roads [10]–[12], railway tracks [13]–[15], and in the airspace [16]. While on roads and railway tracks, the distortions are well pronounced and can be measured with low cost magnetometers, in the airspace the magnetic variations have smaller amplitudes. Hence, high quality magnetometers in combination with an inertial navigation system are required in the airspace.

Position estimation based on magnetic distortions is a highly nonlinear estimation problem. The nonlinearity is introduced in the likelihood in which a map of the distortions is required as the link between the estimated position and the observed magnetometer measurements. Due to this nonlinearity, also nonlinear estimation algorithms are required. A popular approach for magnetic localization is the particle filter used, e.g., in [3], [4], [6], [10], [13], [16]. Besides the particle filter also fingerprinting methods based on convolutional neuronal networks [8], k-nearest neighbors [2] and dynamic time warping [1] are proposed in the literature that perform position estimation based on a batch of data that is fitted to a database of magnetic fingerprints. In this paper we focus on approaches based on Bayesian filters because they allow an easy integration of motion models and control inputs and provide the uncertainties of the estimated position. In particular particle filters are investigated, were the advan-

tage of the particle filter is its simple implementation and capability to handle also "strong" nonlinearities for which linearized filters like the unscented or extended Kalman filter might diverge or give unsatisfying results.

The sampling-based nature of the particle filter has the disadvantage of a high computationally cost compared to Kalman filters, particularly in large state spaces. Fortunately, when only position, speed, and acceleration have to be estimated, the complexity is small enough to carry out the calculation in real time with a reasonable update rate. Even though the particle filter was shown to achieve good results in different scenarios, it is still unclear if it is even close to optimality. In this paper, we therefore show how to obtain a lower bound on the mean squared error. The benefit of such a bound is that it allows to determine the minimal achievable position error for a given magnetic field and it can be used as a benchmark during the development of a localization system to check if a specific estimation algorithm is close to optimality. The bound can also be used to check if a given magnetic field, e.g., inside a room, enables localization with an accuracy that fulfills the requirements of the targeted application. If the requirements are not fulfilled, one has to think about using additional sensors in combination with the magnetometer to achieve the desired accuracy before developing the actual localization system.

The proposed lower bound is based on the Bayesian Cramér-Rao lower bound (BCRLB) introduced by Van Trees [17]. The BCRLB combines the Cramér-Rao bound defined by the inverse Fisher information matrix (FIM) with a state space model allowing to incorporate prior knowledge, e.g., a motion model into the bound. Furthermore, in [18] a recursive version of the bound was derived that reduces the computational complexity. The BCRLB requires a measurement model that is analytically known and that leads to a differentiable likelihood. For magnetic localization this causes some difficulties because the true magnetic field map and hence the likelihood is unknown in general. The magnetic map is only known (with some uncertainty) at positions at which it was measured. Using the laws of physics to describe the magnetic field in analytical form is not possible because this would require an exact knowledge of all materials in the environment, its physical properties, and dimensions. In addition, even when the environment is exactly known, the involved differential equations have to be solved with numerical methods. In this paper, we therefore follow the approach presented in [19], where Gaussian processes (GPs) are learned based on a training data set to describe the measurement model. In [19], the authors applied this idea to obtain a lower bound for localization with received signal strength fingerprints. In contrast to this, here we adapt this methodology to the problem of magnetic localization and we derive the equations required for implementation and we show how a deterministic input can be incorporated into the bound. To the best of our knowledge this has never be done before.

After an introduction to the topic in Section II and a derivation of the equations specific for magnetic localization in Section III, an evaluation on a novel data set is performed in Section IV. In the evaluation, we use magnetometer measurements recorded in our laboratory to investigate the achievable accuracy of magnetic localization. The bound derived from the measurements is compared to the mean squared error (MSE) of a particle filter to check if the filter is close to optimality. Furthermore, the bound for the magnetic field measured at different heights is investigated and we compare the bounds for the cases when the whole magnetic vector or only its magnitude is considered in the estimation.

II. BAYESIAN CRAMÉR-RAO LOWER BOUND

In this section, the BCRLB and GPs are introduced and it is shown how they can be combined. In addition, an efficient recursive version of the bound is derived.

A. BCRLB FOR ANALYTICAL LIKELIHOOD MODELS

The BCRLB is a Bayesian bound on the minimal achievable MSE. In contrast to the classical Cramér-Rao bound, which is a bound on the variance of an unbiased estimator, the BCRLB is also valid for biased estimators. The BCRLB is defined by the inverse of the Bayesian information matrix (BIM) $\mathcal{J}$ and fulfills the matrix inequality

$$\mathbf{M} = \mathrm{E}_{\mathbf{x},\mathbf{z}}\left[(\hat{\mathbf{x}}(\mathbf{z}) - \mathbf{x})(\hat{\mathbf{x}}(\mathbf{z}) - \mathbf{x})^T\right] \geq \mathcal{J}^{-1}, \tag{1}$$

where $\mathbf{M}$ is the MSE matrix for an estimator $\hat{\mathbf{x}}$ that estimates the state vector $\mathbf{x}$ given the observations $\mathbf{z}$. The index of the expectation operator $\mathrm{E}_{\mathbf{x},\mathbf{z}}[\cdot]$ indicates that the expectation is taken w.r.t. the joint probability density function (pdf) $p(\mathbf{x}, \mathbf{z})$. The inequality sign states that the difference $\mathbf{M} - \mathcal{J}^{-1}$ is a positive semi-definite matrix. The BCRLB and the BIM was first introduced by Van Trees in [17] and is given by

$$\mathcal{J} = -\mathrm{E}_{\mathbf{x},\mathbf{z}}\left[\Delta_{\mathbf{x}}^{\mathbf{x}} \ln\left(p(\mathbf{z}, \mathbf{x})\right)\right]. \tag{2}$$

The differential operator $\Delta_{\mathbf{b}}^{\mathbf{a}} = \nabla_{\mathbf{b}} \nabla_{\mathbf{a}}^T$ is an outer product of the nabla operator $\nabla_{\mathbf{a}} = \begin{bmatrix} \frac{\partial}{\partial a_1} & \cdots & \frac{\partial}{\partial a_N} \end{bmatrix}^T$ and when applied to a function gives its Hessian matrix. The regularity conditions for the bound are given in [17] and assume that for the bias

$$\mathbf{B}(\mathbf{x}) = \int_{-\infty}^{\infty} \left[\hat{\mathbf{x}}(\mathbf{z}) - \mathbf{x}\right] p(\mathbf{z}|\mathbf{x}) d\mathbf{z} \tag{3}$$

the following limits are fulfilled

$$\begin{aligned} \lim_{x_i \to \infty} &= \mathbf{B}(\mathbf{x}) p(\mathbf{x}) = 0 \\ \lim_{x_i \to -\infty} &= \mathbf{B}(\mathbf{x}) p(\mathbf{x}) = 0 \end{aligned} \tag{4}$$

for each element x_i of vector $\mathbf{x} = \begin{bmatrix} x_1 & \cdots & x_{n_\mathbf{x}} \end{bmatrix}^T$. In addition, the first and second partial derivatives of the joint density $p(\mathbf{z}, \mathbf{x})$ w.r.t. the state must be integrable to ensure the existence of the expectations in the bound. The BCRLB (2) can be also split into a data-dependent part and a part

depending on prior knowledge by applying the Bayes' rule on the joint density $p(\mathbf{z},\mathbf{x}) = p(\mathbf{z}|\mathbf{x})p(\mathbf{x})$

$$\begin{aligned}\mathcal{J} &= \mathrm{E}_{\mathbf{x}}\left[\mathrm{E}_{\mathbf{z}|\mathbf{x}}\left[-\Delta^{\mathbf{x}}_{\mathbf{x}}\ln p(\mathbf{z}|\mathbf{x})\right]\right] + \mathrm{E}_{\mathbf{x}}\left[-\Delta^{\mathbf{x}}_{\mathbf{x}}\ln p(\mathbf{x})\right] \\ &= \mathrm{E}_{\mathbf{x}}\left[\mathcal{F}(\mathbf{x})\right] + \mathrm{E}_{\mathbf{x}}\left[-\Delta^{\mathbf{x}}_{\mathbf{x}}\ln p(\mathbf{x})\right], \end{aligned} \tag{5}$$

where $\mathcal{F}(\mathbf{x})$ is the FIM of the likelihood $p(\mathbf{z}|\mathbf{x})$. The BIM therefore is the sum of the expectation of the FIM over the state distribution $p(\mathbf{x})$ and the information obtained from the prior distribution itself. Connecting the BIM to the FIM is useful in practice because for many likelihoods the FIM can be found in the literature. In the BCRLB it is assumed that the measurement model and the likelihood $p(\mathbf{z}|\mathbf{x})$ is analytically known and the corresponding $\mathcal{F}(\mathbf{x})$ can be calculated. This is the case for many estimation and localization problems. For example, for global navigation satellite systems the range measurement model is given by the Euclidean distance between the receiver and satellite antenna and some additive white Gaussian noise (AWGN). For such models the FIM is well defined and easy to calculate. As we will see in the next section, this is not the case for magnetic localization.

B. BCRLB FOR MEASUREMENT-BASED LIKELIHOODS

In contrast to the previous section, now the case is addressed in which the measurement model itself is not known analytically but noisy measurements are available. In the following, we differentiate between the true but unknown measurement model

$$z_k = \tilde{\mathrm{h}}\left(\mathbf{x}_k\right) + n_k \tag{6}$$

where $n_k \sim \mathcal{N}(0,\sigma_n^2)$ is AWGN and $\tilde{\mathrm{h}}\left(\cdot\right)$ is the function that relates the state $\mathbf{x}_k \in \mathbb{R}^{n_\mathbf{x}}$ to the measurements z_k at time step k. Because $\tilde{\mathrm{h}}\left(\cdot\right)$ is unknown, an approximation $\tilde{\mathrm{h}}\left(\mathbf{x}\right) \approx \mathrm{h}\left(\mathbf{x},\mathcal{D}\right)$ based on some measured training data set $\mathcal{D} = \{\mathbf{X}_\mathcal{D},\mathbf{Z}_\mathcal{D}\}$ is required, where $\mathbf{Z}_\mathcal{D}$ is a set of noisy measurements of the true measurement model and $\mathbf{X}_\mathcal{D}$ are the corresponding known inputs. Here only functions from an input space with dimension $n_\mathbf{x}$ to a scalar output $\tilde{\mathrm{h}} : \mathbb{R}^{n_\mathbf{x}} \mapsto \mathbb{R}$ are considered. For vector-valued functions, e.g., the magnetic vector field, the measurement model is considered a combination of multiple scalar functions.

In principal, there are many ways to approximate a function but for the use in the BCRLB an approximation should be chosen that accounts for its stochastic nature introduced by the training data set. This paper follows the approach proposed in [19] where the measurement function is approximated with a GP. The advantage of the GP approximation is that the resulting function will be differentiable and for each input, here the state vector $\mathbf{x}$, the function value is Gaussian distributed allowing for a straightforward calculation of the FIM and BIM. To properly derive the bound for GP based measurement models the BCRLB is conditioned on the training data. This is achieved by conditioning all pdfs on the training data set $\mathcal{D}$. In contrast to the bound in [19], here we condition the bound also on a deterministic control sequence $\mathcal{U}$ which allows us to better control the state trajectories over which the bound is calculated, more details on this are presented in Section III. For brevity, in the following the training data set and the control input is combined in the set $\mathcal{D}_u = \{\mathcal{D},\mathcal{U}\}$. The bound conditioned on $\mathcal{D}_u$ is given by

$$\mathbf{M} = \mathrm{E}_{\mathbf{x},\mathbf{z}|\mathcal{D}_u}\left[(\hat{\mathbf{x}}(\mathbf{z}) - \mathbf{x})(\hat{\mathbf{x}}(\mathbf{z}) - \mathbf{x})^T\right] \geq \mathcal{J}^{-1} \tag{7}$$

with the BIM

$$\mathcal{J} = -\mathrm{E}_{\mathbf{x},\mathbf{z}|\mathcal{D}_u}\left[\Delta^{\mathbf{x}}_{\mathbf{x}}\ln\left(p(\mathbf{z},\mathbf{x}|\mathcal{D}_u)\right)\right]. \tag{8}$$

The proof for the conditional bound is exactly the same as the proof for the normal BCRLB in (1) which can be found in [17]. The only difference is that the pdf $p(\mathbf{z},\mathbf{x})$ is exchanged with the conditional pdf $p(\mathbf{z},\mathbf{x}|\mathcal{D}_u)$. For the regularity condition this results in

$$\mathbf{B}(\mathbf{x}) = \int_{-\infty}^{\infty}\left[\hat{\mathbf{x}}(\mathbf{z}) - \mathbf{x}\right]p(\mathbf{z}|\mathbf{x},\mathcal{D}_u)d\mathbf{z} \tag{9}$$

with the limits

$$\begin{aligned}\lim_{x_i\to\infty} &= \mathbf{B}(\mathbf{x})p(\mathbf{x}|\mathcal{D}_u) = 0 \\ \lim_{x_i\to-\infty} &= \mathbf{B}(\mathbf{x})p(\mathbf{x}|\mathcal{D}_u) = 0.\end{aligned} \tag{10}$$

As before in (5), the bound consist of a data dependent part and a part accounting for the prior information

$$\begin{aligned}\mathcal{J} &= \mathrm{E}_{\mathbf{x}|\mathcal{D}_u}\left[\mathrm{E}_{\mathbf{z}|\mathcal{D}_u,\mathbf{x}}\left[-\Delta^{\mathbf{x}}_{\mathbf{x}}\ln p(\mathbf{z}|\mathbf{x},\mathcal{D}_u)\right]\right] \\ &\quad + \mathrm{E}_{\mathbf{x}|\mathcal{D}_u}\left[-\Delta^{\mathbf{x}}_{\mathbf{x}}\ln p(\mathbf{x}|\mathcal{D}_u)\right] \\ &= \mathrm{E}_{\mathbf{x}|\mathcal{D}_u}\left[\mathcal{F}(\mathbf{x})\right] + \mathrm{E}_{\mathbf{x}|\mathcal{D}_u}\left[-\Delta^{\mathbf{x}}_{\mathbf{x}}\ln p(\mathbf{x}|\mathcal{D}_u)\right].\end{aligned} \tag{11}$$

The main step in obtaining the BCRLB is the calculation of the FIM

$$\mathcal{F}(\mathbf{x}) = \mathrm{E}_{\mathbf{z}|\mathcal{D}_u,\mathbf{x}}\left[-\Delta^{\mathbf{x}}_{\mathbf{x}}\ln p(\mathbf{z}|\mathbf{x},\mathcal{D}_u)\right], \tag{12}$$

where the likelihood $p(\mathbf{z}|\mathbf{x},\mathcal{D}_u)$ conditioned on the training data is required.

1) Likelihood of GP-based Measurement Models

In this section, we give a short overview on the GP equations that are required for the derivation of the GP based BCRLB. For a more detailed introduction about GPs and a detailed derivation of the following equations we recommend reading [20].

For the calculation of the BCRLB, the approximation of $\mathrm{h}\left(\cdot\right)$ is considered to be a GP. Therefore, the output of the function itself is a Gaussian random variable with pdf

$$p(\mathrm{h}\left(\mathbf{x}_k\right)|\mathbf{x}_k,\mathcal{D}_u) = \mathcal{N}\left(\mathrm{h}\left(\mathbf{x}_k\right);\mu(\mathbf{x}_k,\mathcal{D}_u),r(\mathbf{x}_k,\mathcal{D}_u)\right), \tag{13}$$

where $\mu(\mathbf{x}_k,\mathcal{D}_u)$ is the posterior mean and $r(\mathbf{x}_k,\mathcal{D}_u)$ the posterior covariance. The posterior mean and covariance of a GP are functions of the training data and the input for which

the measurement function has to be evaluated

$$\begin{aligned}\mu(\mathbf{x}_k,\mathcal{D}) &= \mathrm{m}\left(\mathbf{x}_k\right) \\ &\quad + \mathbf{K}_{\mathbf{x}_k\mathcal{D}}\left[\mathbf{K}_{\mathcal{D}\mathcal{D}}+\sigma_{\mathcal{D}}^2\mathbf{I}\right]^{-1}\left(\mathbf{Z}_{\mathcal{D}}-\mathrm{m}\left(\mathbf{X}_{\mathcal{D}}\right)\right), \\ r(\mathbf{x}_k,\mathcal{D}) &= \mathbf{K}_{\mathbf{x}_k,\mathbf{x}_k}-\mathbf{K}_{\mathbf{x}_k\mathcal{D}}\left[\mathbf{K}_{\mathcal{D}\mathcal{D}}+\sigma_{\mathcal{D}}^2\mathbf{I}\right]^{-1}\mathbf{K}_{\mathcal{D}\mathbf{x}_k}, \end{aligned} \quad (14)$$

where the shorthand notation $\mathbf{K}_{\mathcal{D}\mathcal{D}} = \mathbf{K}(\mathbf{X}_{\mathcal{D}},\mathbf{X}_{\mathcal{D}})$ and $\mathbf{K}_{\mathbf{x}_k\mathcal{D}} = \mathbf{K}(\mathbf{x}_k,\mathbf{X}_{\mathcal{D}})$ is used to describe the covariance matrices between the training points with themselves and the covariance between the training points and the input for which the GP is evaluated. The covariance matrix $\mathbf{K}(\cdot,\cdot)$ for two sets $\mathbf{A}$ and $\mathbf{B}$ of input points has the form

$$\mathbf{K}(\mathbf{A},\mathbf{B}) = \begin{bmatrix} \mathrm{k}\left(\{\mathbf{A}\}_1,\{\mathbf{B}\}_1\right) & \cdots & \mathrm{k}\left(\{\mathbf{A}\}_1,\{\mathbf{B}\}_{|\mathbf{B}|}\right) \\ \vdots & \ddots & \vdots \\ \mathrm{k}\left(\{\mathbf{A}\}_{|\mathbf{A}|},\{\mathbf{B}\}_1\right) & \cdots & \mathrm{k}\left(\{\mathbf{A}\}_{|\mathbf{A}|},\{\mathbf{B}\}_{|\mathbf{B}|}\right) \end{bmatrix} \quad (15)$$

and is fully described by the covariance function $\mathrm{k}(\cdot,\cdot)$ evaluated at the different set elements $\{\mathbf{A}\}_i$ and $\{\mathbf{B}\}_j$. The noise of the training data is accounted for with the variance $\sigma_{\mathcal{D}}^2$.

The choice of the covariance function depends on the underlying true function that is approximated. In this paper, the GP approximates a function that maps each input, here the position of a magnetometer, to the magnetic field value at that input. From practical experience it is known that the magnetic field changes with the position and that the field at two positions becomes less correlated when the distance between the points increases. As proposed in [5], [7], this behavior can be captured with the well-known squared exponential kernel

$$\mathrm{k}\left(\mathbf{a},\mathbf{b}\right) = \sigma_{\mathrm{k}}^2 \exp\left(-\frac{(\mathbf{a}-\mathbf{b})^T(\mathbf{a}-\mathbf{b})}{2l^2}\right) \quad (16)$$

that decays with increasing distance between the input points. The parameter l is the length scale and controls how wide the kernel is and how rapid the covariance decays. The other parameter is the variance σ_{k}^2 that is a measure of the overall variance of the function that is approximated. The mean function is assumed to be constant because we expect the magnetic field to vary around the value of the Earth magnetic field that can be considered homogenous in small areas like a room.

To obtain the likelihood required for the FIM in (12), the measurement model (6) is combined with (13). The observation z_k is according to (6) the sum of two Gaussian random variables, one for the GP that describes the measurements function and one for the noise of the sensor. The sensor noise is considered white and independent from the GP. The likelihood of the observation is therefore the convolution of the pdf in (13) and the noise pdf

$$\begin{aligned} p(z_k|\mathcal{D}_u,\mathbf{x}_k) &= \mathcal{N}\left(z_k;\mu(\mathbf{x}_k,\mathcal{D}_u),r(\mathbf{x}_k,\mathcal{D}_u)+\sigma_n^2\right) \\ &= \mathcal{N}\left(z_k;\mu(\mathbf{x}_k,\mathcal{D}_u),r_{\mathrm{C}}(\mathbf{x}_k,\mathcal{D}_u)\right). \end{aligned} \quad (17)$$

The resulting likelihood is again Gaussian and the covariances of the GP and the noise are simply summed up. Intuitively this is plausible, the likelihood gets wider and less informative when the sensor noise, the uncertainty in the GP or both increase.

2) Fisher Information Matrix of GP-based Likelihoods

In this section, the FIM for the likelihood in (17) is presented. The equations for the FIM are obtained in a straightforward manner from the equations of the general Gaussian case [21]

$$\begin{aligned}\left[\mathcal{F}(\mathbf{x}_k)\right]_{ij} &= \frac{\partial\mu(\mathbf{x}_k,\mathcal{D})}{\partial[\mathbf{x}_k]_i} r_{\mathrm{C}}^{-1}(\mathbf{x}_k,\mathcal{D}) \frac{\partial\mu(\mathbf{x}_k,\mathcal{D})}{\partial[\mathbf{x}_k]_j} \\ &\quad + \frac{1}{2} r_{\mathrm{C}}^{-2}(\mathbf{x}_k,\mathcal{D}) \frac{\partial r_{\mathrm{C}}(\mathbf{x}_k,\mathcal{D})}{\partial[\mathbf{x}_k]_i} \frac{\partial r_{\mathrm{C}}(\mathbf{x}_k,\mathcal{D})}{\partial[\mathbf{x}_k]_j}. \end{aligned} \quad (18)$$

The FIM in (18) is defined element-wise, where $\left[\mathcal{F}(\mathbf{x}_k)\right]_{ij}$ is the element in i-th row and j-th column. To get the FIM for the GP-based measurement model therefore the derivatives of the mean and covariance in (17) w.r.t. the state variables are required. For any differentiable covariance function $\mathrm{k}(\cdot,\cdot)$ and mean function $\mathrm{m}(\cdot)$ the derivatives of the posterior mean are

$$\frac{\partial\mu(\mathbf{x}_k,\mathcal{D})}{\partial[\mathbf{x}_k]_i} = \frac{\partial}{\partial[\mathbf{x}_k]_i}\mathrm{m}\left(\mathbf{x}_k\right) + \sum_{l=1}^{M}[\mathbf{S}^{-1}\tilde{\mathbf{z}}]_l \frac{\partial\mathrm{k}\left(\mathbf{x}_k,\{\mathbf{X}_{\mathcal{D}}\}_l\right)}{\partial[\mathbf{x}_k]_i}. \quad (19)$$

For the posterior covariance the derivatives are given by

$$\begin{aligned}\frac{\partial r_{\mathrm{C}}(\mathbf{x}_k,\mathcal{D})}{\partial[\mathbf{x}_k]_i} &= \frac{\partial}{\partial[\mathbf{x}_k]_i}\mathrm{k}\left(\mathbf{x}_k,\mathbf{x}_k\right) - \\ &\quad 2\left(\frac{\partial}{\partial[\mathbf{x}_k]_i}\mathbf{K}_{\mathbf{x}_k\mathcal{D}}\right)\mathbf{S}^{-1}\mathbf{K}_{\mathcal{D}\mathbf{x}_k}, \end{aligned} \quad (20)$$

where we used the notation $\mathbf{S} = \mathbf{K}(\mathbf{X}_{\mathcal{D}},\mathbf{X}_{\mathcal{D}}) + \sigma_{\mathcal{D}}^2\mathbf{I}$ and $\tilde{\mathbf{z}} = \mathbf{z}_{\mathcal{D}} - \mathrm{m}\left(\mathbf{X}_{\mathcal{D}}\right)$. For a specific GP model the only thing left to do when calculating the FIM is to plug in the derivatives of the mean and covariance function into (19) and (20). For the GP in this paper those are the derivatives of the squared exponential kernel (16)

$$\begin{aligned}\frac{\partial\mathrm{k}\left(\mathbf{x}_k,\mathbf{a}\right)}{\partial[\mathbf{x}_k]_i} &= \frac{\partial}{\partial[\mathbf{x}_k]_i}\sigma_{\mathrm{k}}^2\exp\left(-\frac{(\mathbf{x}_k-\mathbf{a})^T(\mathbf{x}_k-\mathbf{a})}{2l^2}\right) \\ &= -\mathrm{k}\left(\mathbf{x}_k,\mathbf{a}\right)\frac{([\mathbf{x}_k]_i-[\mathbf{a}]_i)}{l^2} = \frac{\partial\mathrm{k}\left(\mathbf{a},\mathbf{x}_k\right)}{\partial[\mathbf{x}_k]_i} \end{aligned} \quad (21)$$

and the constant mean function

$$\frac{\partial}{\partial[\mathbf{x}_k]_i}\mathrm{m}\left(\mathbf{x}_k\right) = \frac{\partial}{\partial[\mathbf{x}_k]_i}\bar{m} = 0 \quad (22)$$

where $\mathbf{a}$ is an arbitrary vector with appropriate dimension.

C. BCRLB FOR NONLINEAR FILTERING

Magnetic localization is a dynamic estimation problem because the position will change over time. Therefore, the BCRLB for the state $\mathbf{x}_k$ at a specific time step k introduced in the previous section has to be extended to state sequences. Furthermore, during localization the state at the current time step typically depends on the previous one. This dependency

can be described in form of a motion model as it is used, e.g., in Kalman and particle filters. For the following nonlinear motion models of the form

$$\mathbf{x}_k = \mathrm{f}\left(\mathbf{x}_{k-1}, \mathcal{U}\right) + \mathbf{w}_{k-1} \tag{23}$$

are considered. The process noise $\mathbf{w}_{k-1}$ is assumed to be AWGN. In the derivation of the BCRLB for nonlinear filtering this is not required but for the implementation of the BCRLB the AWGN case allows for some simplifications. For the filtering bound let $\mathbf{X}_{0:k+1} = \begin{bmatrix}\mathbf{x}_0^T \cdots \mathbf{x}_{k+1}^T\end{bmatrix}^T$ be a vector in which the state sequence from time step 0 to $k+1$ is stacked. Similarly $\mathbf{Z}_{1:k+1} = \begin{bmatrix}\mathbf{z}_1^T \cdots \mathbf{z}_{k+1}^T\end{bmatrix}^T$ is the vector containing the sequence of measurements. According to (8), for $\mathbf{X}_{0:k+1}$ the BIM is

$$\mathcal{J}_{0:k+1} = -\mathrm{E}_{\mathbf{X}_{0:k+1},\mathbf{Z}_{1:k+1}|\mathcal{D}_u}\left[\Delta^{\mathbf{X}_{0:k+1}}_{\mathbf{X}_{0:k+1}} \ln p_{k+1}\right], \tag{24}$$

where the index of $\mathcal{J}$ indicates for which state sequence it is valid and p_{k+1} is the joint pdf

$$p_{k+1} = p(\mathbf{X}_{0:k+1}, \mathbf{Z}_{1:k+1}|\mathcal{D}_u). \tag{25}$$

The dimension of $\mathcal{J}_{0:k+1}$ is $(k+2)n_\mathbf{x} \times n_\mathbf{x}(k+2)$ and hence for long sequences the BIM becomes hard to invert. In [18], the authors address this issue with a recursive version of the bound in which only matrices with dimension $n_\mathbf{x} \times n_\mathbf{x}$ are inverted. In the next paragraph, we follow the proof given in [18] and point out where some assumptions have to be made about the GP in the measurement model in order to obtain an efficient recursive form of the bound. The basis for the recursive calculation is to split $\mathbf{X}_{0:k+1} = \begin{bmatrix}\mathbf{X}_{0:k}^T & \mathbf{x}_{k+1}^T\end{bmatrix}^T$, which allows us to write the BIM as a block matrix

$$\mathcal{J}_{0:k+1} = \begin{bmatrix}\mathbf{A}_{k+1} & \mathbf{B}_{k+1} \\ \mathbf{B}_{k+1}^T & \mathbf{C}_{k+1}\end{bmatrix}, \tag{26}$$

where we used that the differential operator in (24) can be written in block form

$$\begin{aligned}\Delta^{\mathbf{X}_{0:k+1}}_{\mathbf{X}_{0:k+1}} &= \begin{bmatrix}\nabla_{\mathbf{X}_{0:k}} \\ \nabla_{\mathbf{x}_{k+1}}\end{bmatrix}\begin{bmatrix}\nabla^T_{\mathbf{X}_{0:k}} & \nabla^T_{\mathbf{x}_{k+1}}\end{bmatrix} \\ &= \begin{bmatrix}\Delta^{\mathbf{X}_{0:k}}_{\mathbf{X}_{0:k}} & \Delta^{\mathbf{x}_{k+1}}_{\mathbf{X}_{0:k}} \\ (\Delta^{\mathbf{x}_{k+1}}_{\mathbf{X}_{0:k}})^T & \Delta^{\mathbf{x}_{k+1}}_{\mathbf{x}_{k+1}}\end{bmatrix}.\end{aligned} \tag{27}$$

Furthermore, if $\mathcal{J}_{0:k+1}$ is the BIM for the sequence $\mathbf{X}_{0:k+1}$, the BCRLB for the single state $\mathbf{x}_{k+1}$ is given by the lower right block of $\mathcal{J}^{-1}_{0:k+1}$. With block matrix (26) and the Schur complement we obtain

$$\mathbf{M}_{k+1} \geq \left(\mathbf{C}_{k+1} - \mathbf{B}_{k+1}^T \mathbf{A}_{k+1}^{-1} \mathbf{B}_{k+1}\right)^{-1} = \mathcal{J}^{-1}_{k+1}. \tag{28}$$

In (28) only the upper left block matrix $\mathbf{A}_{k+1}$ has to be inverted but this still has the dimension $(k+1)n_\mathbf{x} \times n_\mathbf{x}(k+1)$. To further reduce the dimension of the involved matrices the different blocks in (26) need to be broken down themselves into blocks. In the following the expectations are always taken w.r.t. the joint density $p(\mathbf{X}_{0:k+1}, \mathbf{Z}_{1:k+1}|\mathcal{D}_u)$ when no other density is explicitly mentioned in the index of the expectation operator. For the joint density it is also useful to define the identity

$$\begin{aligned}p_{k+1} &= p(\mathbf{x}_{k+1}, \mathbf{z}_{k+1}|\mathcal{D}_u, \mathbf{X}_{0:k}, \mathbf{Z}_{1:k}) \underbrace{p(\mathbf{X}_{0:k}, \mathbf{Z}_{1:k}|\mathcal{D}_u)}_{=p_k} \\ &= p(\mathbf{z}_{k+1}|\mathcal{D}_u, \mathbf{X}_{0:k+1}, \mathbf{Z}_{1:k}) p(\mathbf{x}_{k+1}|\mathcal{D}_u, \mathbf{X}_{0:k}, \mathbf{Z}_{1:k}) p_k.\end{aligned} \tag{29}$$

The matrix $\mathbf{A}_{k+1}$ is split into four blocks

$$\begin{aligned}\mathbf{A}_{k+1} &= \begin{bmatrix}\mathbf{A}^{(11)}_{k+1} & \mathbf{A}^{(12)}_{k+1} \\ (\mathbf{A}^{(12)}_{k+1})^T & \mathbf{A}^{(22)}_{k+1}\end{bmatrix} \\ &= -\mathrm{E}\left[\begin{bmatrix}\Delta^{\mathbf{X}_{0:k-1}}_{\mathbf{X}_{0:k-1}} & \Delta^{\mathbf{x}_k}_{\mathbf{X}_{0:k-1}} \\ \Delta^{\mathbf{X}_{0:k-1}}_{\mathbf{x}_k} & \Delta^{\mathbf{x}_k}_{\mathbf{x}_k}\end{bmatrix} \ln p_{k+1}\right].\end{aligned} \tag{30}$$

Matrix $\mathbf{B}_{k+1}$ is split only into two blocks

$$\begin{aligned}\mathbf{B}_{k+1} &= \begin{pmatrix}\mathbf{B}^{(1)}_{k+1} \\ \mathbf{B}^{(2)}_{k+1}\end{pmatrix} \\ &= -\mathrm{E}\left[\begin{bmatrix}\Delta^{\mathbf{x}_{k+1}}_{\mathbf{X}_{0:k-1}} \\ \Delta^{\mathbf{x}_{k+1}}_{\mathbf{x}_k}\end{bmatrix} \ln p_{k+1}\right].\end{aligned} \tag{31}$$

With (30) and (31) the BIM in (28) becomes

$$\begin{aligned}\mathcal{J}_{k+1} = \mathbf{C}_{k+1} - \\ \left[(\mathbf{B}^{(1)}_{k+1})^T (\mathbf{B}^{(2)}_{k+1})^T\right]\begin{bmatrix}\mathbf{A}^{(11)}_{k+1} & \mathbf{A}^{(12)}_{k+1} \\ (\mathbf{A}^{(12)}_{k+1})^T & \mathbf{A}^{(22)}_{k+1}\end{bmatrix}^{-1}\begin{bmatrix}\mathbf{B}^{(1)}_{k+1} \\ \mathbf{B}^{(2)}_{k+1}\end{bmatrix}.\end{aligned} \tag{32}$$

The efficient form presented in [18] requires the matrix $\mathbf{B}^{(1)}_{k+1}$ to be zero because this reduces (32) to

$$\mathcal{J}_{k+1} = \mathbf{C}_{k+1} - (\mathbf{B}^{(2)}_{k+1})^T (\mathbf{A}^{-1}_{k+1})^{(22)} \mathbf{B}^{(2)}_{k+1}. \tag{33}$$

where the lower right block of $\mathbf{A}^{-1}_{k+1}$ is obtained from (30) and the Schur complement

$$(\mathbf{A}^{-1}_{k+1})^{(22)} = \left(\mathbf{A}^{(22)}_{k+1} - (\mathbf{A}^{(12)}_{k+1})^T (\mathbf{A}^{(11)}_{k+1})^{-1} \mathbf{A}^{(12)}_{k+1}\right)^{-1}. \tag{34}$$

It can be seen from the definition of $\mathbf{B}^{(1)}_{k+1}$ in (31) that the matrix becomes zero when no part of the joint pdf in (29) depends on $\mathbf{x}_{k+1}$ and $\mathbf{X}_{0:k-1}$ at the same time due the differentiation w.r.t. this state vectors. This is fulfilled when both, the likelihood and the motion model have the Markov property

$$\begin{aligned}p(\mathbf{z}_{k+1}|\mathcal{D}_u, \mathbf{X}_{0:k+1}, \mathbf{Z}_{1:k}) &= p(\mathbf{z}_{k+1}|\mathcal{D}_u, \mathbf{x}_{k+1}) \\ p(\mathbf{x}_{k+1}|\mathcal{D}_u, \mathbf{X}_{0:k}, \mathbf{Z}_{1:k}) &= p(\mathbf{x}_{k+1}|\mathcal{D}_u, \mathbf{x}_k).\end{aligned} \tag{35}$$

The matrices in (33) all have the dimension $n_\mathbf{x} \times n_\mathbf{x}$ but the calculation of (34) still involves larger matrices. Fortunately, due to the Markov property it can be shown [18], that

$$\mathbf{A}_{k+1} = \begin{bmatrix}\mathbf{A}_k & \mathbf{B}_k \\ \mathbf{B}_k^T & \mathbf{C}_k + \mathbf{D}^{11}_k\end{bmatrix} \tag{36}$$

where the matrices $\mathbf{A}_k$, $\mathbf{B}_k$, $\mathbf{C}_k$ are the matrices in (26) for time step k and $\mathbf{D}^{11}_k = -\mathrm{E}\left[\Delta^{\mathbf{x}_k}_{\mathbf{x}_k} \ln p(\mathbf{x}_{k+1}|\mathcal{D}_u, \mathbf{x}_k)\right]$. Given (28), (34) and (36) the connection to the previous BIM is

established by

$$\begin{aligned}(\mathbf{A}_{k+1}^{-1})^{(22)} &= \left(\mathbf{C}_k + \mathbf{D}_k^{11} - \mathbf{B}_k^T \mathbf{A}_k^{-1} \mathbf{B}_k\right)^{-1} \\ &= (\mathbf{D}_k^{11} + \mathcal{J}_k)^{-1}. \end{aligned} \tag{37}$$

Inserting (37) into (33) results in the recursive bound for which all involved matrices have dimension $n_{\mathbf{x}} \times n_{\mathbf{x}}$

$$\mathcal{J}_{k+1} = \mathbf{C}_{k+1} - (\mathbf{B}_{k+1}^{(2)})^T (\mathbf{D}_k^{11} + \mathcal{J}_k)^{-1} \mathbf{B}_{k+1}^{(2)}. \tag{38}$$

For better readability and comparability in the following we will use the notation from [18] where the bound is given by

$$\mathcal{J}_{k+1} = \mathbf{D}_k^{22} - (\mathbf{D}_k^{12})^T \left(\mathbf{D}_k^{11} + \mathcal{J}_k\right)^{-1} \mathbf{D}_k^{12} \tag{39}$$

with $\mathbf{C}_{k+1} = \mathbf{D}_k^{22}$ and $\mathbf{B}_{k+1}^{(2)} = \mathbf{D}_k^{12}$. The Markov property required for the efficient recursive formulation of the filtering bound implies that the GP as a part of the likelihood must be learned offline. Theoretically, as it is the case for simultaneous localization and mapping, the GP can be also adapted online based on all available measurements. For the online learning case, the BCRLB has to be calculated by inverting the large BIM for the whole sequence and the derivation of the concrete equations for implementation becomes much more involved as pointed out also in [19]. Fortunately, the bound of interest here is for localization only and assuming an offline trained GP is reasonable in practice.

III. IMPLEMENTATION OF THE BOUND

In the following, the implementation of the recursive bound for motion models of the form (23) is discussed.

A. EQUATIONS FOR GENERAL NONLINEAR MOTION WITH AWGN AND GP-BASED LIKELIHOODS

For the calculation of (39), the matrices $\mathbf{D}_k^{11}$, $\mathbf{D}_k^{12}$ and $\mathbf{D}_k^{22}$ are required. In addition the recursion must be initialized with the BIM $\mathcal{J}_0$ at time step zero at which no measurement is available yet. From (30), (31), and (26) one can see, that the required matrices are

$$\begin{aligned}\mathbf{D}_k^{11} &= -\operatorname{E}_{\mathbf{X}_{k:k+1}|\mathcal{D}_u}\left[\Delta_{\mathbf{x}_k}^{\mathbf{x}_k} \ln p(\mathbf{x}_{k+1}|\mathcal{D}_u, \mathbf{x}_k)\right] \\ \mathbf{D}_k^{12} &= -\operatorname{E}_{\mathbf{X}_{k:k+1}|\mathcal{D}_u}\left[\Delta_{\mathbf{x}_k}^{\mathbf{x}_{k+1}} \ln p(\mathbf{x}_{k+1}|\mathcal{D}_u, \mathbf{x}_k)\right] \\ \mathbf{D}_k^{22} &= -\operatorname{E}_{\mathbf{x}_{k+1},\mathbf{z}_{k+1}|\mathcal{D}_u}\left[\Delta_{\mathbf{x}_{k+1}}^{\mathbf{x}_{k+1}} \ln p(\mathbf{z}_{k+1}|\mathcal{D}_u, \mathbf{x}_{k+1})\right] \\ &\quad - \operatorname{E}_{\mathbf{X}_{k:k+1}|\mathcal{D}_u}\left[\Delta_{\mathbf{x}_{k+1}}^{\mathbf{x}_{k+1}} \ln p(\mathbf{x}_{k+1}|\mathcal{D}_u, \mathbf{x}_k)\right]. \end{aligned} \tag{40}$$

Note, in the above equations the expectations are theoretically over the complete joint density but the variables not part of the argument are simply marginalized out of the joint density by taking the expectation. A connection to the FIM of the current measurement is obtained from rewriting $\mathbf{D}_k^{22}$

$$\begin{aligned}\mathbf{D}_k^{22} &= \operatorname{E}_{\mathbf{x}_{k+1}|\mathcal{D}_u}\left[-\operatorname{E}_{\mathbf{z}_{k+1}|\mathbf{x}_{k+1},\mathcal{D}_u}\left[\Delta_{\mathbf{x}_{k+1}}^{\mathbf{x}_{k+1}} \ln p(\mathbf{z}_{k+1}|\mathcal{D}_u, \mathbf{x}_{k+1})\right]\right] \\ &\quad - \operatorname{E}_{\mathbf{X}_{k:k+1}|\mathcal{D}_u}\left[\Delta_{\mathbf{x}_{k+1}}^{\mathbf{x}_{k+1}} \ln p(\mathbf{x}_{k+1}|\mathcal{D}_u, \mathbf{x}_k)\right] \\ &= \operatorname{E}_{\mathbf{x}_{k+1}|\mathcal{D}_u}\left[\mathcal{F}(\mathbf{x}_{k+1})\right] \\ &\quad - \operatorname{E}_{\mathbf{X}_{k:k+1}|\mathcal{D}_u}\left[\Delta_{\mathbf{x}_{k+1}}^{\mathbf{x}_{k+1}} \ln p(\mathbf{x}_{k+1}|\mathcal{D}_u, \mathbf{x}_k)\right]. \end{aligned} \tag{41}$$

For the system model (23) with the Gaussian pdf $p(\mathbf{x}_{k+1}|\mathcal{D}_u, \mathbf{x}_k) = \mathcal{N}\left(\mathrm{f}\left(\mathbf{x}_k, \mathcal{U}\right), \mathbf{Q}\right)$ this results in the following equations

$$\begin{aligned}\mathbf{D}_k^{11} &= \operatorname{E}_{\mathbf{x}_k|\mathcal{D}_u}\left[\mathbf{F}_k^T \mathbf{Q}^{-1} \mathbf{F}_k\right] \\ \mathbf{D}_k^{12} &= -\operatorname{E}_{\mathbf{x}_k|\mathcal{D}_u}\left[\mathbf{F}_k^T\right] \mathbf{Q}^{-1} = (\mathbf{D}_k^{21})^T \\ \mathbf{D}_k^{22} &= \operatorname{E}_{\mathbf{x}_{k+1}|\mathcal{D}_u}\left[\mathcal{F}(\mathbf{x}_{k+1})\right] + \mathbf{Q}^{-1}, \end{aligned} \tag{42}$$

where $\mathbf{F}_k^T = \nabla_{\mathbf{x}_k} \mathrm{f}\left(\mathbf{x}_k, \mathcal{U}\right)^T$ is the Jacobian matrix of the nonlinear motion model w.r.t. the state vector. The matrices $\mathbf{D}_k^{11}$ and $\mathbf{D}_k^{12}$ only contain information related to the movement model while $\mathbf{D}_k^{22}$ contains the information obtained from the newest measurement. For most cases the expectations have to be calculated numerically because the involved integrals are intractable. One exception is the linear case, for which closed -form solutions exist.

In (42), the expectations are all taken w.r.t. pdfs of the state $\mathbf{x}$ conditioned on the data set and the inputs. After some thought, this makes sense, since the bound assumes that the motion model describes for each time step the distribution of the state vector in the state space, e.g., how likely a robot is at a certain position in space. In which part of the state space the state vector ends up depends on the realization of the process noise and all these possibilities must be reflected in the bound. For the likely case that no closed-form solution exists for the expectation, it can be obtained by simulating multiple state trajectories, evaluating the FIM and the other terms for the different trajectories and then averaging over them in each time step. Due to the law of large numbers, the numerical solution converges to the true solution when enough trajectories are used. The calculation of the expected values was also the reason to include inputs into the bound. The input gives some control about the simulated trajectories. This allows us to simulate more realistic trajectories, e.g., a pedestrian in a building most likely will not perform a random walk. Also, the input enables the simulation of trajectories that stay within a certain part of the state space for which the GP in the likelihood is "well" defined by sufficiently dense measurements. Including the inputs also alters the estimation problem. The resulting bound holds only for filters that also have access to the inputs. For the case where no inputs are available, the equations for the bound remain the same. One simply has to remove the conditioning on $\mathcal{U}$ in the involved pdfs. The numerical evaluation of the expectations comes with considerable computational costs but is still much faster than evaluating the MSE of a filter that requires not only the averaging over different realization of the state vector but also over different realization of the measurement noise.

B. EQUATIONS FOR 2D MOTION OF A WHEELED ROBOT

With the above equations it is now possible to easily evaluated the bound for different types of state space models. For this paper the focus is on localization of a wheeled robot in a 2D plane. The state vector is assumed to contain the x and y

position and the heading φ

$$\mathbf{x}_k = \begin{bmatrix} x_k & y_k & \varphi_k \end{bmatrix}^T. \tag{43}$$

The motion model is

$$\begin{aligned} x_{k+1} &= x_k + \cos(\varphi_k) T v_k + w_{x,k} \\ y_{k+1} &= y_k + \sin(\varphi_k) T v_k + w_{y,k} \\ \varphi_{k+1} &= \varphi_k + T\omega_k + w_{\omega,k} \end{aligned} \tag{44}$$

with the time T between two time steps and the inputs speed v_k and turn rate ω_k. The inputs are considered deterministic but process noise accounts for imperfections in the actuators that can not translate the input perfectly into motion. The process noise is assumed to be Gaussian $\mathbf{w}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}_k)$ with a diagonal covariance matrix $\mathbf{Q}_k = \mathrm{diag}(\begin{bmatrix} \sigma_{w_x}^2 & \sigma_{w_y}^2 & \sigma_{w_\varphi}^2 \end{bmatrix})$.

The Jacobian matrix for (44) is

$$\begin{aligned} \mathbf{F}_k &= (\nabla_{\mathbf{x}_k} \mathrm{f}\left(\mathbf{x}_k, \mathcal{U}\right)^T)^T \\ &= \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ -\sin(\varphi) T v_K & \cos(\varphi) T v_K & 1 \end{bmatrix}^T. \end{aligned} \tag{45}$$

For the observation model we will consider two different cases, first only the magnitude of the magnetic field is used in the likelihood and second the vector field is used. Using only the magnitude simplifies the mapping process because the magnetometer attitude during map creation is not required but may reduce the achievable accuracy. If the accuracy is actually reduced and by how much will be further investigated in the evaluation part of this paper. The question how much the accuracy is affected by the choice of the measurement model is also a nice example for the usefulness of lower bounds.

The measurement model for the magnitude is scalar and of the form (6) with the likelihood defined by (17). The FIM for the calculation of the expected value in $\mathbf{D}_k^{22}$ is obtained from (18). For the vector case we follow the approch in [5] and assume that each component of the vector field is approximated by a separate GP. Hence the measurement model is given by

$$\mathbf{z}_k^b = \mathbf{R}_n^b(\varphi)\, \mathbf{h}\left(\mathbf{x}_k, \mathcal{D}_u\right) + \mathbf{n}_k, \tag{46}$$

with the magnetic map

$$\mathbf{h}\left(\mathbf{x}_k, \mathcal{D}_u\right) = \begin{bmatrix} \mathrm{h}_x^n(\mathbf{x}_k, \mathcal{D}_u) & \mathrm{h}_y^n(\mathbf{x}_k, \mathcal{D}_u) & \mathrm{h}_z^n(\mathbf{x}_k, \mathcal{D}_u) \end{bmatrix}^T, \tag{47}$$

where each element is a GP that is independent from the others. The superscript n indicates that the magnetic field in the map is given in the local navigation frame and the index stands for the different coordinate axis. The measurement is obtained in the body frame b of the magnetometer, to relate the measurement to the map therefore the rotation matrix from navigation to body frame is required. For the rotation matrix we assume the three magnetometer axis are aligned with the vehicle frame as shown in Fig. 1. With this frame

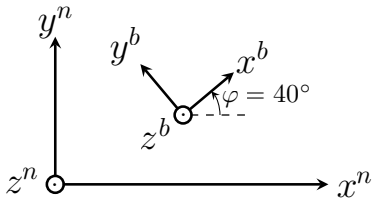


FIGURE 1. Definition of navigation and body frame for and heading angle of $\varphi = 40°$. The body frame is equal to the vehicle frame of the robot and hence the x axis of the body frame points into the driving direction of the robot when its speed is positive.

definitions the rotation matrix is

$$\mathbf{R}_n^b(\varphi) = \begin{bmatrix} \cos(\varphi_k) & \sin(\varphi_k) & 0 \\ -\sin(\varphi_k) & \cos(\varphi_k) & 0 \\ 0 & 0 & 1 \end{bmatrix}. \tag{48}$$

Based on the measurement model (46) the likelihood for the vector measurements can be calculated applying the rules for affine transformation of multivariate Gaussian random variables. When the random variable $\mathbf{a} \sim \mathcal{N}(\boldsymbol{\mu}_a, \boldsymbol{\Sigma}_a)$ is Gaussian distributed then the random variable $\mathbf{b} = \mathbf{A}\mathbf{a}$ follows the Gaussian distribution $\mathbf{b} \sim \mathcal{N}(\mathbf{A}\boldsymbol{\mu}_a, \mathbf{A}\boldsymbol{\Sigma}_a\mathbf{A}^T)$. Thus, the mean and the covariance of the likelihood

$$p(\mathbf{z}_k^b | \mathcal{D}_u, \mathbf{x}_k) \sim \mathcal{N}(\mathbf{z}_k^b; \boldsymbol{\mu}_{z^b}(\mathbf{x}_k, \mathcal{D}_u), \underbrace{\boldsymbol{\Sigma}_{z^b}(\mathbf{x}_k, \mathcal{D}_u) + \sigma_n^2 \mathbf{I}}_{\boldsymbol{\Sigma}_c(\mathbf{x}_k, \mathcal{D}_u)}). \tag{49}$$

are given by

$$\boldsymbol{\mu}_{z^b}(\mathbf{x}_k, \mathcal{D}_u) = \mathbf{R}_n^b(\varphi) \boldsymbol{\mu}_{z^n}(\mathbf{x}_k, \mathcal{D}_u) \tag{50}$$

and

$$\boldsymbol{\Sigma}_{z^b}(\mathbf{x}_k, \mathcal{D}_u) = \mathbf{R}_n^b(\varphi) \boldsymbol{\Sigma}_{z^n}(\mathbf{x}_k, \mathcal{D}_u) \mathbf{R}_n^b(\varphi)^T, \tag{51}$$

where $\boldsymbol{\mu}_{z^n}(\mathbf{x}_k, \mathcal{D}_u)$ is the mean vector in the navigation frame and $\boldsymbol{\Sigma}_{z^n}$ is the covariance in the navigation frame

$$\boldsymbol{\Sigma}_{z^n}(\mathbf{x}_k, \mathcal{D}_u) = \begin{bmatrix} r_x^n(\mathbf{x}_k, \mathcal{D}_u) & 0 & 0 \\ 0 & r_y^n(\mathbf{x}_k, \mathcal{D}_u) & 0 \\ 0 & 0 & r_z^n(\mathbf{x}_k, \mathcal{D}_u) \end{bmatrix}. \tag{52}$$

In (52) $r_i^n \quad i \in \{x, y, z\}$ are the GP variances of the different vector field components. Based on the Gaussian likelihood above, the FIM is defined by the vector version of (18) presented in [21]

$$\begin{aligned} [\mathcal{F}(\mathbf{x}_k)]_{ij} = &\left[\frac{\partial \boldsymbol{\mu}_{z^b}(\mathbf{x}_k)}{\partial [\mathbf{x}_k]_i}\right]^T \boldsymbol{\Sigma}_c^{-1}(\mathbf{x}_k) \left[\frac{\partial \boldsymbol{\mu}_{z^b}(\mathbf{x}_k)}{\partial [\mathbf{x}_k]_j}\right] \\ &+ \frac{1}{2} \mathrm{tr}\left[\boldsymbol{\Sigma}_c^{-1}(\mathbf{x}_k) \left[\frac{\partial \boldsymbol{\Sigma}_c(\mathbf{x}_k)}{\partial [\mathbf{x}_k]_i}\right] \boldsymbol{\Sigma}_c^{-1}(\mathbf{x}_k) \left[\frac{\partial \boldsymbol{\Sigma}_c(\mathbf{x}_k)}{\partial [\mathbf{x}_k]_j}\right]\right]. \end{aligned} \tag{53}$$

In the FIM, all derivatives are w.r.t. the state vector therefore the dependency on $\mathcal{D}_u$ can be dropped for brevity. The derivatives in (53) are element wise. The derivative of the

mean vector (50) therefore is a vector of the derivatives of each component. The derivative w.r.t. x_k is

$$\frac{\partial}{\partial x_k}\begin{bmatrix}\mu_x^b(\mathbf{x}_k)\\ \mu_y^b(\mathbf{x}_k)\\ \mu_z^b(\mathbf{x}_k)\end{bmatrix} = \begin{bmatrix}\cos(\varphi_k)\frac{\partial}{\partial x_k}\mu_x^n(\mathbf{x}_k) + \sin(\varphi_k)\frac{\partial}{\partial x_k}\mu_y^n(\mathbf{x}_k)\\ \cos(\varphi_k)\frac{\partial}{\partial x_k}\mu_y^n(\mathbf{x}_k) - \sin(\varphi_k)\frac{\partial}{\partial x_k}\mu_x^n(\mathbf{x}_k)\\ \frac{\partial}{\partial x_k}\mu_z^n(\mathbf{x}_k).\end{bmatrix}. \tag{54}$$

For y_k the derivative have the same form and only $\frac{\partial}{\partial x_k}$ has to be exchanged for $\frac{\partial}{\partial y_k}$. For the heading one obtains

$$\frac{\partial}{\partial \varphi_k}\begin{bmatrix}\mu_x^b(\mathbf{x}_k)\\ \mu_y^b(\mathbf{x}_k)\\ \mu_z^b(\mathbf{x}_k)\end{bmatrix} = \begin{bmatrix}-\sin(\varphi_k)\mu_x^n(\mathbf{x}_k) + \cos(\varphi_k)\mu_y^n(\mathbf{x}_k)\\ -\cos(\varphi_k)\mu_x^n(\mathbf{x}_k) - \sin(\varphi_k)\mu_y^n(\mathbf{x}_k)\\ 0\end{bmatrix}. \tag{55}$$

For the covariance matrix (51) the derivative is again a matrix containing the derivatives of each element. For x_k the non-zero elements of the derivative are

$$\begin{aligned}
\left[\frac{\partial \mathbf{\Sigma}_c(\mathbf{x}_k)}{\partial x}\right]_{11} &= \cos^2(\varphi_k)\frac{\partial r_x^n(\mathbf{x}_k)}{\partial x_k} + \sin^2(\varphi_k)\frac{\partial r_y^n(\mathbf{x}_k)}{\partial x_k}\\
\left[\frac{\partial \mathbf{\Sigma}_c(\mathbf{x}_k)}{\partial x}\right]_{12} &= \cos(\varphi_k)\sin(\varphi_k)\left(\frac{\partial r_y^n(\mathbf{x}_k)}{\partial x_k} - \frac{\partial r_x^n(\mathbf{x}_k)}{\partial x_k}\right)\\
\left[\frac{\partial \mathbf{\Sigma}_c(\mathbf{x}_k)}{\partial x}\right]_{21} &= \left[\frac{\partial \mathbf{\Sigma}_c(\mathbf{x}_k)}{\partial x}\right]_{12}\\
\left[\frac{\partial \mathbf{\Sigma}_c(\mathbf{x}_k)}{\partial x}\right]_{22} &= \sin^2(\varphi_k)\frac{\partial r_x^n(\mathbf{x}_k)}{\partial x_k} + \cos^2(\varphi_k)\frac{\partial r_y^n(\mathbf{x}_k)}{\partial x_k}\\
\left[\frac{\partial \mathbf{\Sigma}_c(\mathbf{x}_k)}{\partial x}\right]_{33} &= \frac{\partial r_z^n(\mathbf{x}_k)}{\partial x_k}.
\end{aligned} \tag{56}$$

As for the mean, the derivative of the covariance w.r.t. y_k has the same form as (56). For the heading derivative the non-zero elements are

$$\begin{aligned}
\left[\frac{\partial \mathbf{\Sigma}_c(\mathbf{x}_k)}{\partial \varphi_k}\right]_{11} &= 2\cos(\varphi_k)\sin(\varphi_k)(r_y^n(\mathbf{x}_k) - r_x^n(\mathbf{x}_k))\\
\left[\frac{\partial \mathbf{\Sigma}_c(\mathbf{x}_k)}{\partial \varphi_k}\right]_{12} &= (\cos^2(\varphi_k) - \sin^2(\varphi_k))(r_y^n(\mathbf{x}_k) - r_x^n(\mathbf{x}_k))\\
\left[\frac{\partial \mathbf{\Sigma}_c(\mathbf{x}_k)}{\partial \varphi_k}\right]_{21} &= \left[\frac{\partial \mathbf{\Sigma}_c(\mathbf{x}_k)}{\partial \varphi_k}\right]_{12}\\
\left[\frac{\partial \mathbf{\Sigma}_c(\mathbf{x}_k)}{\partial \varphi_k}\right]_{22} &= 2\cos(\varphi_k)\sin(\varphi_k)(r_x^n(\mathbf{x}_k) - r_y^n(\mathbf{x}_k)).
\end{aligned} \tag{57}$$

In the equations (54)–(57) for the FIM now the derivatives of the covariance (20) and mean (22) of the GP in the navigation frame can be inserted. With the FIM then the recursive BCRLB is obtained from (39) and (42).

IV. EVALUATION SETUP

In this section, the overall setup used for the evaluation is introduced. This includes the measurement setup, the hyper-parameter optimization, and an overview on the parameters of the Monte Carlo simulation.

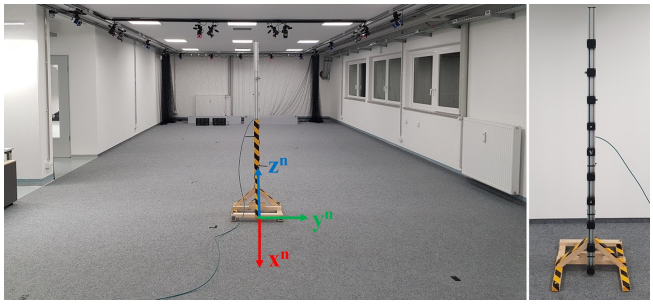


FIGURE 2. (left) Holodeck laboratory at the German Aerospace Center Institute of Communications and Navigation with the definition of the navigation frame. The x^n- and y^n-axis lie in the floor plane and the z^n-axis points upwards. On the ceiling the cameras of the Vicon reference system are visible. (right) Magnetometer array with ten sensors and a overall height og 1.84 m. The sensors are mounted in black aluminimu boxes and are separated by $\approx$ 0.2 m.

A. MEASUREMENT SETUP

The evaluation is based on measurements recorded in the Holodeck laboratory at the German Aerospace Center Institute of Communications and Navigation in Oberpfaffenhofen shown in the left part of Fig. 2. The magnetic field is measured with an array of magnetometers. The array shown in the right part of Fig. 2 consists of ten low-cost Kionix KMX62 magnetometers that are connected via CAN bus to a computer that records the data. The distance between two neighboring sensors is fixed to $\approx 0.2\,\mathrm{m}$. The first sensor is mounted $5.4\,\mathrm{cm}$ above the floor and the tenth sensor is at a height of $1.85\,\mathrm{m}$. For the recording of the magnetic vector field in the lab the array is moved around manually in a vertical position. The magnetic field therefore is measured in ten different planes with different heights. This enables to investigate the impact of the height on the achievable localization accuracy.

The magnetometer measurements are timestamped and synchronized with ground truth pose information of an optical tracking system from Vicon. In Fig. 2 the cameras of the system mounted on the ceiling can be seen. The recorded poses contain the position and the attitude. This enables to correct rotations of the array during the measurements. Prior to the Holodeck measurements, the magnetometers are calibrated with ellipsoid fitting, see e.g. [22], based on a data set recorded on a meadow at the German Aerospace Center site without any buildings close by. The true magnetic flux density of the Earth magnetic field for calibration is obtained from the international geomagnetic reference field model and has a value of $48.61\,\mu\mathrm{T}$. From the calibration data also the standard deviation of the sensor noise was estimated. The average value over all sensors was $\sigma_n \approx 0.23\,\mu\mathrm{T}$. The exact value varies from sensor to senor by $\pm 21.26\,\%$.

TABLE 1. Hyper parameters and dynamic range for the different sensor 1, 6 and 10.

Sensor	h / cm	Height Axis	Kernel σ_k/µT	Kernel l/m	Mean $\bar{m}$/µT	Likelihood $\sigma_{\mathcal{D}}$/µT	Dyn. range D / µT
1	5.40	x	5.92	0.31	9.94	0.33	48.24
1	5.40	y	9.07	0.34	20.06	0.30	69.66
1	5.40	z	9.07	0.30	-42.32	0.34	76.80
1	5.40	mag.	8.42	0.29	51.37	0.29	75.68
6	105.40	x	2.53	1.03	10.91	0.35	13.17
6	105.40	y	4.43	1.15	16.67	0.27	18.30
6	105.40	z	7.83	1.37	-41.28	0.34	17.65
6	105.40	mag.	7.39	1.36	45.32	0.23	19.64
10	185.40	x	2.86	0.92	10.56	0.24	16.08
10	185.40	y	6.07	1.00	16.68	0.24	30.53
10	185.40	z	6.15	0.98	-43.59	0.24	28.05
10	185.40	mag.	5.34	0.95	48.85	0.21	29.98

B. HYPERPARAMETER OPTIMIZATION AND TRAINING DATA

The hyperparameters of the GP are obtained by maximizing the logarithm of the marginal likelihood. The optimization is performed with the GPML Matlab toolbox presented in [23] and uses Gaussian hyper priors on the value of the constant mean function and the standard deviation $\sigma_{\mathcal{D}}$ of the noise of the training data. For each sensor the hyperparameters are optimized individually for each axis and the magnitude over all axes.

During the lab measurements, the magnetometers were moved freely while recording with an update rate of 200 Hz which resulted in a large amount of measurements. For training this amount is reduced by placing an equidistant grid in the measured area and then taking for each grid point only the measurement that was closest to it. When setting the distance between grid points there is a trade-off between computational complexity and the accuracy of the GP approximation. When the amount of points increases the GP is more likely to describe the approximated function accurately while the complexity will increase cubical with the number of grid points. Here the distance in the x- and y-direction between two neighboring point was set to $0.1\,\mathrm{m}$. To check if this spacing is sufficient for a good approximation of the magnetic field, cross validation was performed with measurements that are not part of the training data set $\mathcal{D}$.

In Fig. 3–Fig. 5 the result of the GP regression is shown for magnetometer 1, 6 and 10. The measurement locations for training and cross validation of sensor 1 are shown in Fig. 6. We only show here the GPs of three sensors because this already gives a good idea of the general behavior of the magnetic field at different heights. The hyperparameters obtained from optimization, the height and the dynamic range for the investigated sensors are listed in Table 1. The dynamic range is the difference between the maximum and minimum value of the magnetic field of the respective axis. The dynamic range and the hyperparameters confirm what is also apparent from Fig. 3, the magnetic field close to the floor has the highest spatial variability. The magnetic field

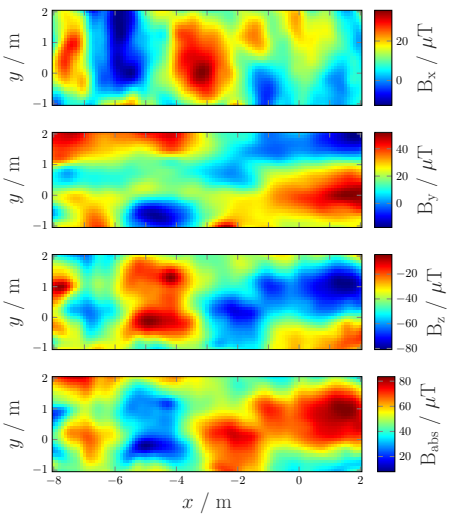


FIGURE 3. Magnetic field of sensor 1, mounted close to the floor, obtained from GP regression on an equidistant grid with a spacing of 0.1 m. From top to bottom the figures shows the x, y, z components and the magnitude of the magnetic field.

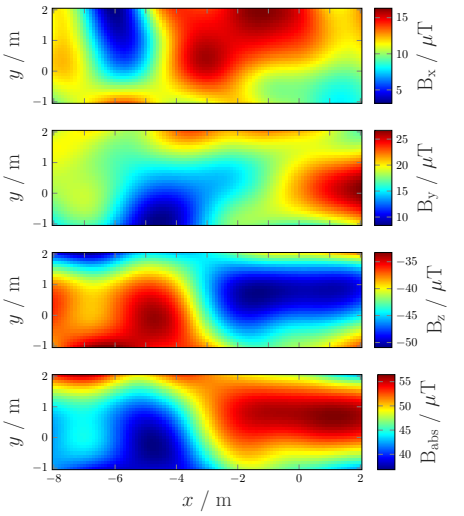


FIGURE 4. Magnetic field of sensor 6, mounted at a height of $\approx 1.05\,\mathrm{m}$, obtained from GP regression on an equidistant grid with a spacing of 0.1 m. From top to bottom the figures shows the x, y, z components and the magnitude of the magnetic field.

and parameters of the other sensors show, that when getting higher above the floor the range and variability first decreases and then increases again while the sensor becomes closer to the ceiling. These results comply with our expectations, when a sensor is close to the floor it is strongly affected by ferromagnetic material like rebar. With an increasing distance to the floor this effect is reduced until the sensor becomes closer to the ceiling. This general trend of the hyperparameters is also visible Fig. 7 where the length scale and the dynamic range is plotted over the ten different sensors heights. Figure 7 also shows that although the hyperparameters follow the same overall trend there are larger differences between the different field components, e.g., from 80-140 cm for the length scale. This could be expected because the influence of

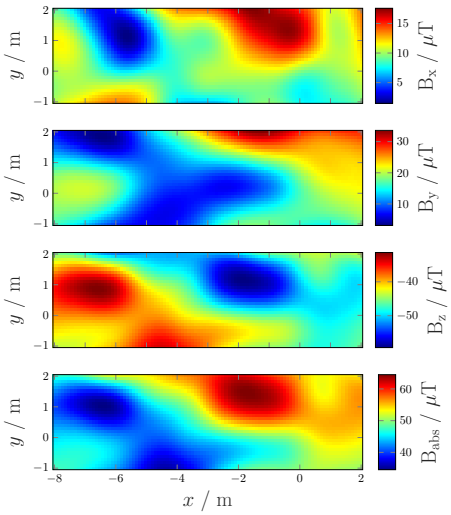


FIGURE 5. Magnetic field of sensor 10, mounted at a height of $\approx 1.85\,\mathrm{m}$, obtained from GP regression on an equidistant grid with a spacing of $0.1\,\mathrm{m}$. From top to bottom the figures shows the x, y, z components and the magnitude of the magnetic field.

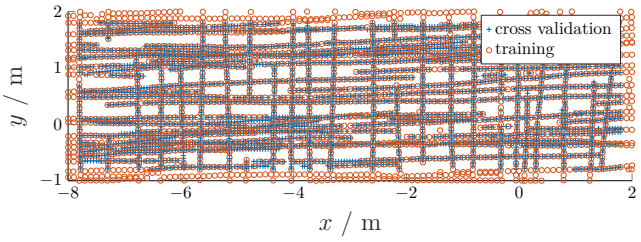


FIGURE 6. Locations of training (red circles) and cross validation (blue crosses) data of sensor 1. The array was moved first in meander pattern and the diagonally through the magnetic field which leads to straight patterns for the input locations.

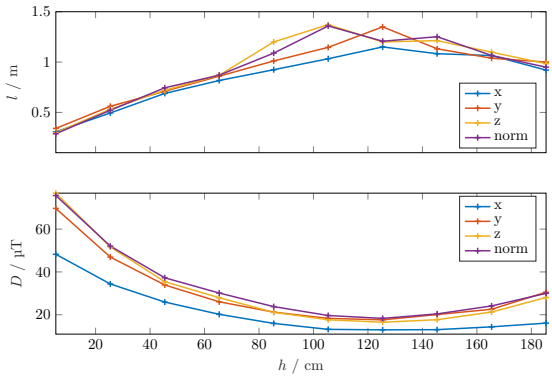


FIGURE 7. Length scale (top) and dynamic range (bottom) over the ten different sensor heights. Both parameters show a clear dependency on the height.

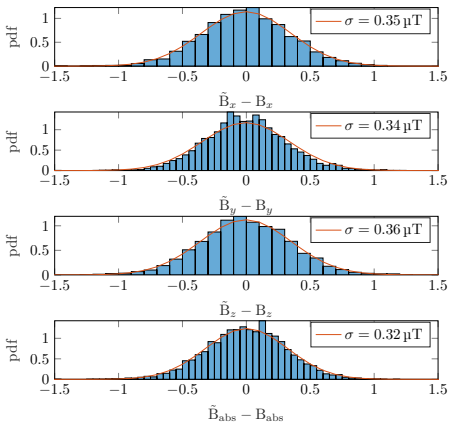


FIGURE 8. Histogram of the cross-validation residual for sensor 1. The redline shows a Gaussian pdf with the mean and standard deviation of the residuals.

ferromagnetic material on a magnetic field is not necessarily the same for all field components as can be observed also from Fig. 3-5.

Before the BCRLB for the above shown magnetic fields is calculated, cross-validation is performed to verify that the learned GPs closely approximate the underlying magnetic field. The histogram of the residual between the cross-validation measurements and the GP for sensor 1 evaluated at the respective inputs is shown in Fig. 8. The residual for the different sensor axis is approximately Gaussian distributed and has a standard deviation slightly larger than the value $\sigma_{\mathcal{D}}$ obtained from hyperparameter optimization. This seems to be the case also for the magnitude. The magnitude measurements follow theoretically a non-central chi distribution. Fortunately, when the magnitude is large compared to the measurement noise variance the chi distribution approaches the shape of a Gaussian with a variance equal to the measurement noise and a mean close to the true mean.

For a perfect approximation the residual would contain only sensor noise with a standard deviation of $\sigma_n \approx 0.23\,\mu\mathrm{T}$. We suspect that the additional "noise" is partially caused by inaccuracies in the ground truth. An position error of only $5\,\mathrm{mm}$ and an magnetic field gradient of $35\,\mu\mathrm{T/m}$ already causes errors of $0.175\,\mu\mathrm{T}$. In addition, small attitude errors in the range of $0.2°$ and a magnetic field magnitude of $35\,\mu\mathrm{T}$ could lead to deviations of $0.12\,\mu\mathrm{T}$ in the measurements of the different sensor axis. From this we conclude that the residuals are close to what is to be expected and hence the GP seems to capture the underlying magnetic field with a high accuracy.

C. PARAMETERS FOR MONTE CARLO SIMULATION

The numerical approximation of the exceptions in (42) is performed over sample trajectories obtained from the motion model in (44). The quality of the approximation increases with the number N_{MC} of trajectories but using many trajectories also increases the computation time required to obtain the bound. To find a compromise between accuracy and computation time, the BCRLB for different values of N_{MC} was computed. In the computation the parameters for the process noise are $\sigma_{w_x} = \sigma_{w_y} = 0.01\,\mathrm{m}$ and $\sigma_{w_\varphi} = 0.025°$. The initial state distribution is Gaussian with mean $\mathbf{x}_0 = \begin{bmatrix}-7\,\mathrm{m} & 0.5\,\mathrm{m} & 0°\end{bmatrix}$ and covariance $\mathbf{P}_0 = \mathrm{diag}(\begin{bmatrix}(0.1\,\mathrm{m})^2 & (0.1\,\mathrm{m})^2 & (2°)^2\end{bmatrix})$. The input values are constant $v_k = 0.7\,\mathrm{m/s}$ and $\omega_k = 2.5\,°/\mathrm{s}$. Under the Gaus-

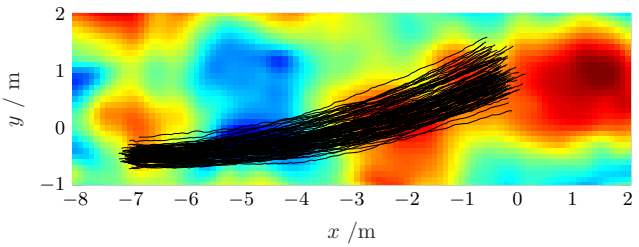


FIGURE 9. Magnitude of the magnetic field of sensor 1 and 100 trajectories generated from the motion model with the parameters set to the values described in section IV-C.

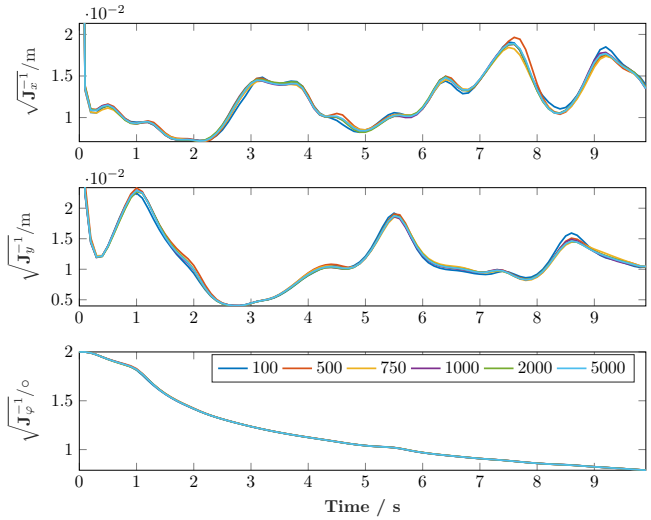


FIGURE 10. BCRLB for sensor 1 when using only the magnitude calculated over different values of N_{MC}. The first two plots (from bottom to top) show the square root of the bound on the x and y position and the third plot shows the bound on the heading φ.

sian assumption for the initial state distribution the BIM that starts the recursion in (39) is $\mathcal{J}_0 = \mathbf{P}_0^{-1}$. The measurement noise is set to the measured average value $\sigma_n = 0.23\,\mu\mathrm{T}$ of the KMX sensors. The duration of the simulation is 100 steps with a step size of $T = 0.1\,\mathrm{s}$. In Fig. 9, 100 example trajectories created with these parameters are plotted on top of the magnitude of the magnetic field of sensor 1.

The bound for different values of N_{MC} are shown in Fig. 10. The results indicate that even for 100 trajectories the bound is already close to the bound for $N_{\mathrm{MC}} = 5000$. To gain a little bit more accuracy in the calculation for the remainder of the paper $N_{\mathrm{MC}} = 750$ was selected.

V. RESULTS AND DISCUSSION

In this section, the achievable accuracy of magnetic localization is evaluated with the GP based BCRLB and system model (44). In the evaluation, the BCRLB is calculated for two cases. In the first case, only the magnetic magnitude is considered and in the second case, the whole vector is accounted for. For both cases, the bound for three sensor heights is calculated. Furthermore, the BCRLB is compared to the empirical root mean squared error (RMSE) of a sampling importance resampling particle filter. This comparison

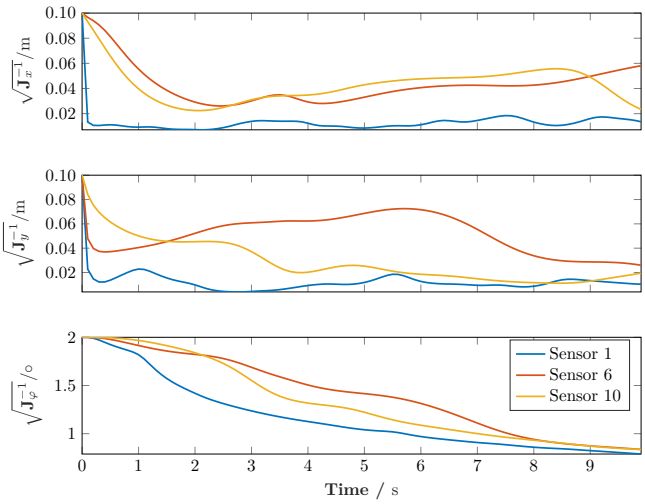


FIGURE 11. BCRLB over different sensor heights when only the **magnitude** of the magnetic field is considered. The first two plots (from bottom to top) show the square root of the bound on the x and y position and the third plot shows the bound on the heading φ.

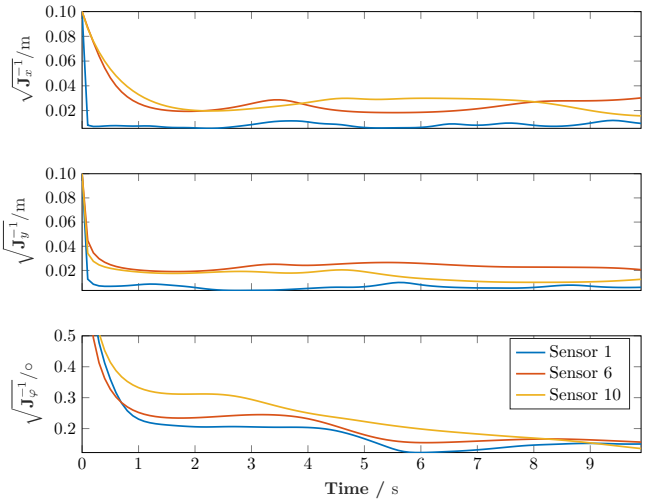


FIGURE 12. BCRLB over different sensor heights when the whole magnetic **vector field** is considered. The heading part is zoomed in to better show the difference between the different bounds. The first two plots (from bottom to top) show the square root of the bound on the x and y position and the third plot shows the bound on the heading φ.

shows how well the filter performs and is at the same time a plausibility check for the bound.

A. BCRLB FOR MAGNETIC LOCALIZATION

In Fig. 11 and Fig. 12 the square root of the BCRLB diagonal elements for sensor 1, 6 and 10 are shown. Overall the achievable position accuracy is well below $10\,\mathrm{cm}$. For the heading one can expect accuracies of a few degrees for the magnitude and below one degree for the vector case. Overall using the whole magnetic vector field results in a bound that is always below the one of the magnitude only case. In particular the heading is improved by using the whole vector, this is not only reflected in a higher accuracy but also in an significantly improved convergence time. This behavior

was to be expected, because for the magnitude only case the measurements are independent from the heading and hence the entries of the FIM associated with the heading become zero. That the heading is still observable, in the sense that the BCRLB gets reduced over time, is due to the motion model. Over the motion the heading is related to the position which is related to the measurements. The observability therefore is not direct. It should be mentioned, that in this particular example it is assumed that the magnetometer is mounted in the center of rotation. If the sensor would be mounted with a lever arm w.r.t. the center of rotation a rotation would lead to a change in the measured magnitude and the FIM for the heading would contain non-zero entries improving the observability. How well this works will likely depend on the length of the lever arm, is the length small compared to the length scale heading changes would only lead to small changes in the magnitude which can vanish in the sensor noise. In the vector case if the position is known the heading can be obtained from only one measurement. This is the principle on which compasses are based. This direct observability of the heading results in a fast convergence of the vector bound and heading errors below one degree after the first measurement. For the magnitude case the one degree threshold is only reached after almost $6\,\mathrm{s}$.

The comparison between the different sensors show, that overall sensor 1 close to the floor achieves most of the time the lowest values. Considering the definition of the FIM in (53) and the hyperparameters in Table 1 this is not surprising. Sensor 1 has the highest dynamic range and smallest length scale leading to more areas in the magnetic field with strong gradients that give rise to higher values in the FIM. Between Sensor 6 and Sensor 10 the differences are not that obvious anymore, e.g., the heading BCRLB of the vector case for Sensor 6 is smaller than for sensor 10 for most of the time. A similar behavior is observed for the x position between $4\,\mathrm{s}$–$8\,\mathrm{s}$ for the vector and the magnitude only case.

In summary the results show that in the given environment a sensor close to the floor is able to achieve localization accuracies below $1\,\mathrm{cm}$. For sensors higher above the floor the performance degrades due to a reduced spatial variability in the magnetic field. Nevertheless even for Sensor 6 which has the largest distance from the floor and the ceiling the position accuracy was below $10\,\mathrm{cm}$. If one can chose between the magnitude and the vector case clearly using the whole vector field is the better option from an accuracy point of few. This is not surprising, taking the magnitude of the vector field reduces the information and makes the magnetic field at different position less distinct. From a practical point of view the magnitude case is still interesting because it largely simplifies the mapping in which it is sufficient to know the sensor position in space but not its attitude.

B. PARTICLE FILTER PERFORMANCE

The BCRLB is a lower bound for the MSE of any filter. This means the bound can be used as optimality measure during filter development. In this section its shown how well

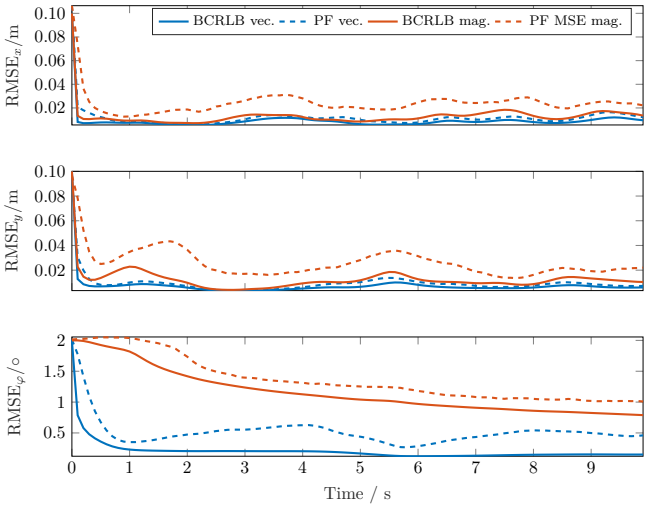


FIGURE 13. Comparison between the particle filter and the bound. The comparison shows the RMSE for magnitude only and vector measurements.

the particle filter performs when applied to magnetic localization and if there is room for improvement. In particular the sampling importance resampling filter [24] is evaluated. The MSE is obtained from running the particle filter over the 750 state trajectories and 25 realization of the sensor noise along these trajectories. This means in total the filter is evaluated 18750 times. From the result the MSE if obtained by averaging. This approach is needed due the definition of the MSE in (1) as the expected value of the estimation error w.r.t. the joint pdf of the states and measurements.

The used implementation of the particle filter is mostly a standard sampling importance resampling filter. In each time step first a prediction based on the motion model and the inputs is performed and then the weights are updated. If the effective sampling size is below $0.8 \cdot N_p$ resampling is performed. During the implementation of the filter it quickly became apparent that evaluating the GP for every particle slows down the filter significantly. Due to the large number of required filter evaluations this becomes quickly unfeasible. The complexity of the filter is also a problem in practice where for most applications a real time position solution is desired. Therefore, we decided to modify the filter by not evaluating the GP variance as one might do in practice. The variance evaluation is the bottleneck as it requires a matrix vector multiplication while the mean only requires the calculation of the scalar product between two vectors. This can be easily seen from (14). The variance contains the term $\mathbf{K}_{\mathbf{x}_k\mathcal{D}}\mathbf{S}^{-1}\mathbf{K}_{\mathcal{D}\mathbf{x}_k}$ that requires the left and right multiplication of a vector that depends on the current state $\mathbf{x}_k$ and the square matrix $\mathbf{S}^{-1}$ with a dimension equal to the amount of training data. For the mean only $\mathbf{K}_{\mathbf{x}_k\mathcal{D}}\mathbf{S}^{-1}(\mathbf{z}_\mathcal{D} - \mathrm{m}\,(\mathbf{X}_\mathcal{D}))$ has to be evaluated where $\mathbf{S}^{-1}(\mathbf{z}_\mathcal{D} - \mathrm{m}\,(\mathbf{X}_\mathcal{D}))$ is a vector that depends solely on the training data and hence can be calculated once offline. Neglecting the variance of the GP speeds up the particle filter significantly but also removes the information about the GP uncertainty. Therefore, to account

for the variance of the GP in the weight update the standard deviation of the likelihood was set to $1.2 \cdot \sigma_n$. To increase the stability of the filter also the process noise was increased by $20\,\%$ compared to the noise set in the Monte Carlo simulation. For this settings, Fig. 13 shows the RMSE of the particle filter in comparison to the square root of the BCRLB.

The result in the figure shows that in principle the filter RMSE follows the shape of the bound but cannot reach it. For the vector case, the filter RMSE of the 2D position is quite close to the bound with an difference in the order of millimeters after the filter has converged at around $1\,\mathrm{s}$. When only the magnitude is considered, this gap is larger and more in the range of $1\,\mathrm{cm}$. For position estimation it seems that including the whole magnetic vector improves the result significantly also for the filter. This seems reasonable, considering the whole magnetic vector field makes the field at a certain position more "unique" which helps the filter to converge toward the correct position. For the heading one can observe that overall the filter result for the vector field is well below the result and the bound for the magnitude only case. As before this was to be expected. Observing the full vector with each measurement enables the filter to directly observe the heading instead of having to wait until the sensor moves through the changing magnetic field as it the case for the magnitude only case. As already for the bound, this results in a faster decay in the heading uncertainty after initialization.

VI. CONCLUSIONS

In this paper, the BCRLB with deterministic inputs and GP-based likelihoods was adapted to the problem of magnetic field-based localization. For magnetic localization, two special cases were considered. In the first case, only the magnetic magnitude was assumed to be available to the estimator and in the second case, the whole magnetic vector field was considered. The bound for both cases was then evaluated based on measurements of an indoor environment with multiple sensors mounted in different heights above the floor. To the best of our knowledge this was never done before.

The first step of the evaluation was to optimize the GP hyperparameters individually for all sensors. The resulting hyperparameters show a clear height dependence. With an increasing distance to the floor and the ceiling, the length scale of the squared exponential kernel increases and the overall dynamic range of the magnetic field decreases. In the second step, the BCRLBs for three selected sensors were calculated. The bounds show that for the three sensor heights position accuracies below $10\,\mathrm{cm}$ are attainable, for the heading values below 1° are possible. The results clearly indicate that using the magnetic vector instead of its magnitude improves the position and heading accuracy. This comes with no surprise since taking the magnitude reduces the information contained in the magnetic field. The magnitude case is still quite important in practice since it allows to reduce the effort required for creating the magnetic map by avoiding the need for an attitude reference system. In the final step of the evaluation the BCRLB was compared to the RMSE of a particle filter. The position RMSE of the filter is close to the bound for the vector case but never attains it. For the magnitude case a larger difference is observed. Hence, in both cases the results show that the used particle filter is not optimal w.r.t. the RMSE.

REFERENCES

[1] P. Li, X. Yang, Y. Yin, S. Gao, and Q. Niu, "Smartphone-based indoor localization with integrated fingerprint signal," *IEEE Access*, vol. 8, pp. 33 178–33 187, 2020.

[2] Z. Wu, M. Wen, G. Peng, X. Tang, and D. Wang, "Magnetic-assisted initialization for infrastructure-free mobile robot localization," in *2019 IEEE International Conference on Cybernetics and Intelligent Systems (CIS) and IEEE Conference on Robotics, Automation and Mechatronics (RAM)*, 2019, pp. 518–523.

[3] M. Frassl, M. Angermann, M. Lichtenstern, P. Robertson, B. J. Julian, and M. Doniec, "Magnetic maps of indoor environments for precise localization of legged and non-legged locomotion," in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Nov. 2013, pp. 913–920.

[4] J. Haverinen and A. Kemppainen, "Global indoor self-localization based on the ambient magnetic field," *Robotics and Autonomous Systems*, vol. 57, no. 10, pp. 1028–1035, Oct. 2009.

[5] N. Akai and K. Ozaki, "Gaussian processes for magnetic map-based localization in large-scale indoor environments," in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Sep. 2015, pp. 4459–4464.

[6] M. Kok and A. Solin, "Scalable Magnetic Field SLAM in 3D Using Gaussian Process Maps," in *2018 21st International Conference on Information Fusion (FUSION)*, Jul. 2018, pp. 1353–1360.

[7] I. Vallivaara, J. Haverinen, A. Kemppainen, and J. Röning, "Simultaneous localization and mapping using ambient magnetic field," in *2010 IEEE Conference on Multisensor Fusion and Integration*, Sep. 2010, pp. 14–19.

[8] I. Ashraf, M. Kang, S. Hur, and Y. Park, "MINLOC: Magnetic Field Patterns-Based Indoor Localization Using Convolutional Neural Networks," *IEEE Access*, vol. 8, pp. 66 213–66 227, 2020.

[9] S. Lee, S. Chae, and D. Han, "ILoA: Indoor Localization Using Augmented Vector of Geomagnetic Field," *IEEE Access*, vol. 8, pp. 184 242–184 255, 2020.

[10] J. A. Shockley and J. F. Raquet, "Navigation of Ground Vehicles Using Magnetic Field Variations," *Navigation*, vol. 61, no. 4, pp. 237–252, 2014.

[11] D. Wei, X. Ji, W. Li, H. Yuan, and Y. Xu, "Vehicle localization based on odometry assisted magnetic matching," in *2017 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, Sep. 2017, pp. 1–6.

[12] S. B. Cruz and A. Aguiar, "MagLand: Magnetic Landmarks for Road Vehicle Localization," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 4, pp. 3654–3667, Apr. 2020.

[13] B. Siebler, O. Heirich, and S. Sand, "Train Localization with Particle Filter and Magnetic Field Measurements," in *2018 21st International Conference on Information Fusion (FUSION)*, Jul. 2018, pp. 1–5.

[14] ——, "Bounding ins positioning errors with magnetic-field-signatures in railway environments," in *Proceedings of the 30th International Technical Meeting of The Satellite Division of the Institute of Navigation (ION GNSS+ 2017)*, September 2017, pp. 3224 – 3230.

[15] B. Siebler, O. Heirich, S. Sand, and U. D. Hanebeck, "Joint Train Localization and Track Identification based on Earth Magnetic Field Distortions," in *2020 IEEE/ION Position, Location and Navigation Symposium (PLANS)*, Apr. 2020, pp. 941–948.

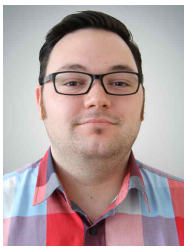
[16] A. Canciani and J. Raquet, "Airborne Magnetic Anomaly Navigation," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 53, no. 1, pp. 67–80, Feb. 2017.

[17] H. L. Van Trees, *Detection, estimation, and modulation theory, part I: detection, estimation, and linear modulation theory*. John Wiley & Sons, 2004.

[18] P. Tichavsky, C. Muravchik, and A. Nehorai, "Posterior Cramer-Rao bounds for discrete-time nonlinear filtering," *IEEE Transactions on Signal Processing*, vol. 46, no. 5, pp. 1386–1396, May 1998.

[19] Y. Zhao, C. Fritsche, G. Hendeby, F. Yin, T. Chen, and F. Gunnarsson, "Cramér–Rao Bounds for Filtering Based on Gaussian Process State-Space Models," *IEEE Transactions on Signal Processing*, vol. 67, no. 23, pp. 5936–5951, Dec. 2019.

[20] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning*, ser. Adaptive computation and machine learning. Cambridge, Massachusetts: MIT Press, 2006.
[21] S. M. Kay, *Fundamentals of statistical signal processing*. New Jersey: Prentice Hall PTR, 1993.
[22] J. Fang, H. Sun, J. Cao, X. Zhang, and Y. Tao, "A Novel Calibration Method of Magnetic Compass Based on Ellipsoid Fitting," *IEEE Transactions on Instrumentation and Measurement*, vol. 60, no. 6, pp. 2053–2061, Jun. 2011.
[23] C. E. Rasmussen and H. Nickisch, "Gaussian Processes for Machine Learning (GPML) Toolbox," *Journal of Machine Learning Research*, vol. 11, no. 100, pp. 3011–3015, 2010.
[24] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking," *IEEE Transactions on Signal Processing*, vol. 50, no. 2, pp. 174–188, Feb. 2002.



BENJAMIN SIEBLER (M'18) received his M.Sc. in electrical engineering and information technology from the Karlsruhe Institute of Technology (KIT), Germany, in 2014. He is currently pursuing his PhD at the Intelligent Sensor-Actuator-Systems Laboratory of the KIT. Since 2014 he is a researcher at the Institute of Communications and Navigation of the German Aerospace Center in Oberpfaffenhofen. His research focus is on multisensor fusion for localization applications with inertial sensors and magnetic field distortions.

STEPHAN SAND (SM'10) received his PhD from ETH Zurich, Switzerland in 2010. Since 2002, Stephan has been researching on wireless communications and multi-sensor navigation at the Institute of Communications and Navigation of the German Aerospace Center in Oberpfaffenhofen, Germany. Since 2014 he is leading the Vehicular Applications Group researching novel systems that combine robust navigation and wireless communications technologies for road users and railways. Stephan has authored and co-authored more than 100 publications in conferences and journal. He has obtained several patents and actively contributes to vehicular and railway communication standardization, e.g. in ETSI JTFIR and IEEE 802.11bd.

UWE D. HANEBECK (F'17) is a chaired professor of Computer Science at the Karlsruhe Institute of Technology (KIT) in Germany and director of the Intelligent Sensor-Actuator-Systems Laboratory (ISAS). He obtained his Ph.D. degree in 1997 and his habilitation degree in 2003, both in Electrical Engineering from the Technical University in Munich, Germany. His research interests are in the areas of information fusion, nonlinear state estimation, stochastic modeling, system identification, and control with a strong emphasis on theory-driven approaches based on stochastic system theory and uncertainty models. He is author and coauthor of more than 500 publications in various high-ranking journals and conferences and an IEEE Fellow.