

## An Extended Kalman Filter for Magnetic Field SLAM Using Gaussian Process Regression

Viset, F.M.; Helmons, R.L.J.; Kok, M.

**DOI**

[10.3390/s22082833](https://doi.org/10.3390/s22082833)

**Publication date**

2022

**Document Version**

Final published version

**Published in**

Sensors

**Citation (APA)**

Viset, F. M., Helmons, R. L. J., & Kok, M. (2022). An Extended Kalman Filter for Magnetic Field SLAM Using Gaussian Process Regression. *Sensors*, 22(8), Article 2833. <https://doi.org/10.3390/s22082833>

**Important note**

To cite this publication, please use the final published version (if applicable). Please check the document version above.

**Copyright**


Other than for strictly personal use, it is not permitted to download, forward or distribute the text or part of it, without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license such as Creative Commons.

**Takedown policy**

Please contact us and provide details if you believe this document breaches copyrights. We will remove access to the work immediately and investigate your claim.

Article

# An Extended Kalman Filter for Magnetic Field SLAM Using Gaussian Process Regression

Frida Viset <sup>1,\*</sup>, Rudy Helmons <sup>2</sup> and Manon Kok <sup>1</sup> 

<sup>1</sup> Delft Center for Systems and Control, Delft University of Technology, 2628 CD Delft, The Netherlands; m.kok-1@tudelft.nl

<sup>2</sup> Maritime and Transport Technology, Delft University of Technology, 2628 CD Delft, The Netherlands; r.l.j.helmons@tudelft.nl

\* Correspondence: f.m.viset@tudelft.nl

**Abstract:** We present a computationally efficient algorithm for using variations in the ambient magnetic field to compensate for position drift in integrated odometry measurements (dead-reckoning estimates) through simultaneous localization and mapping (SLAM). When the magnetic field map is represented with a reduced-rank Gaussian process (GP) using Laplace basis functions defined in a cubical domain, analytic expressions of the gradient of the learned magnetic field become available. An existing approach for magnetic field SLAM with reduced-rank GP regression uses a Rao-Blackwellized particle filter (RBPF). For each incoming measurement, training of the magnetic field map using an RBPF has a computational complexity per time step of  $O(N_p N_m^2)$ , where  $N_p$  is the number of particles, and  $N_m$  is the number of basis functions used to approximate the Gaussian process. Contrary to the existing particle filter-based approach, we propose applying an extended Kalman filter based on the gradients of our learned magnetic field map for simultaneous localization and mapping. Our proposed algorithm only requires training a single map. It, therefore, has a computational complexity at each time step of  $O(N_m^2)$ . We demonstrate the workings of the extended Kalman filter for magnetic field SLAM on an open-source data set from a foot-mounted sensor and magnetic field measurements collected onboard a model ship in an indoor pool. We observe that the drift compensating abilities of our algorithm are comparable to what has previously been demonstrated for magnetic field SLAM with an RBPF.

**Keywords:** simultaneous localization and mapping; Kalman filtering; localization; magnetic field



**Citation:** Viset, F.; Helmons, R.; Kok, M. An Extended Kalman Filter for Magnetic Field SLAM Using Gaussian Process Regression. *Sensors* **2022**, *22*, 2833.

<https://doi.org/10.3390/s22082833>

Academic Editors: Antoni Perez-Navarro, Raúl Montoliu and Joaquín Torres-Sospedra

Received: 25 February 2022

Accepted: 5 April 2022

Published: 7 April 2022

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

Autonomous navigation using only onboard sensors is a desirable technology for various applications. There is no stable access to GNSS signals indoors, underground, or underwater, as they are blocked by building elements, earth and water, respectively [1–3]. In other environments, the use of GNSS signals for localization can also be challenging. Surface vehicles in harbours can be close to containers, bridges, or other industrial elements that can cause multi-path effects on the GNSS (Global navigation satellite system) measurements [4]. Autonomous navigation using only onboard sensors is challenging because of the drift in the position estimate obtained from sensor measurements that are independent of pre-deployed infrastructure [5]. Drift occurs when noisy measurements are integrated to estimate position without any absolute position measurements [6]. We will refer to the position estimates and orientation estimates obtained when integrating such noisy measurements as odometry. A range of other possible sensor readings may be available in these scenarios. The scope of our research is limited to the investigation of autonomous navigation using onboard odometry and magnetic field measurements.

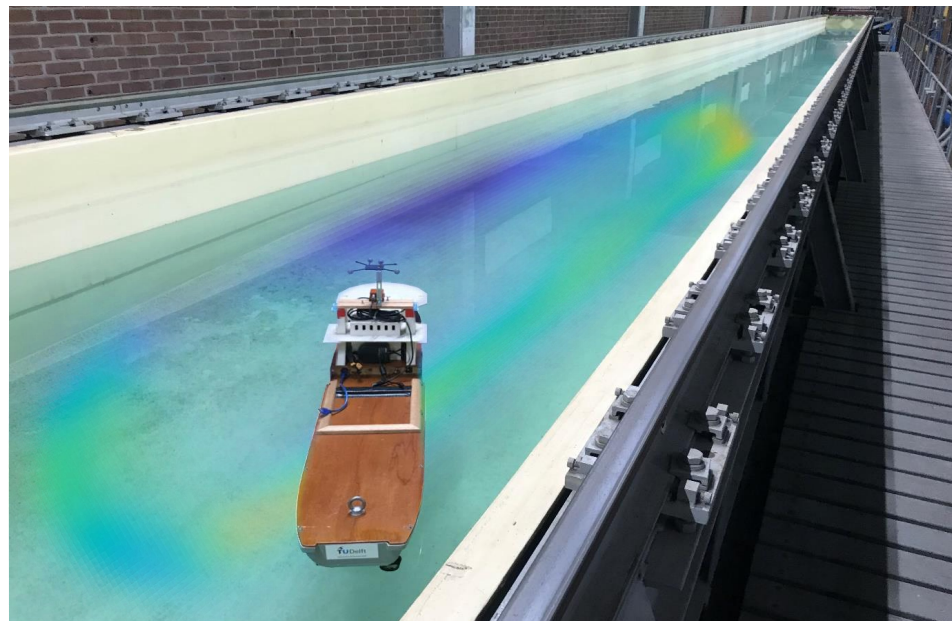
Magnetic field simultaneous localization and mapping (SLAM) has been proposed to compensate for odometry drift when there is access to magnetic field measurements

in a magnetic field with stationary spatial variations [7]. It has been demonstrated for indoor localization that magnetic field SLAM can be used to improve position estimates [8]. In environments with ferromagnetic structures, as for example in indoor environments, the magnetic field has rich spatial variation due to the magnetization of the metal [9]. Navigation using nonlinear variations in the ambient magnetic field has been proposed for a range of applications, such as indoor localization [1,8,10–22], underwater localization [23–31], and surface and aerial navigation [32,33]. Although [32] uses an extended Kalman filter (EKF) for localization in a learned magnetic field, the majority use a particle filter [1,10–13,15–18,22,25,27]. A comparative study has demonstrated that the particle filter is more accurate for underwater geomagnetic navigation in the case where the initial position is not known, while the EKF is more computationally efficient [34].

Computationally tractable magnetic field SLAM in three dimensions was proposed in [8], using a Rao-Blackwellized particle filter (RBPF) to simultaneously estimate the position and orientation of a pedestrian as well as the ambient magnetic field. A set of  $N_p$  particles are used to represent the position and orientation [8]. The RBPF for magnetic field SLAM proposed by [8] uses Gaussian process (GP) regression to combine knowledge about the nature of the magnetic field from Maxwell's equations with measurements of the magnetic field to create a magnetic field map. To this end, they build the magnetic field map for each particle using reduced-rank Gaussian process regression, which represents the magnetic field map as a linear combination of  $N_m$  Laplace basis functions on hexagonal domains, and which represents the magnetic field map uncertainty as a matrix with  $N_m^2$  entries. As each magnetic field map is represented with the weights of  $N_m$  basis functions and the corresponding covariance of these weights, all of which require updating at each time step, the computational cost of updating the magnetic field map is  $O(N_p N_m^2)$  [8]. In the case where the particle filter is run on a parallelized architecture, such as FPGAs, the computation time dependence on the number of particles can be reduced dramatically [35,36]. The scope of our research is limited to improving the speed of Magnetic field SLAM on non-parallelized architectures. Magnetic field SLAM has also been demonstrated feasible using an RBPF with Laplace basis functions defined on a single, cubic domain [22].

The contribution of this paper is an approach to magnetic field SLAM that is faster and requires less storage compared to the approach proposed in [8], inspired by the goal to run magnetic field SLAM in real-time on cheap carry-on units with low processing power. A property of using reduced-rank Gaussian process regression for magnetic field SLAM in a cubic domain is that the magnetic field map is given as a linear combination of analytically described basis functions [37]. We can therefore use the spatial derivatives of the closed-form solutions of the Laplacian to find the Jacobian of the magnetic field map with respect to the position estimate. To reduce computational expenses, we propose utilising the availability of analytical Jacobians of the reduced-rank Gaussian process magnetic field maps to perform magnetic field SLAM using an extended Kalman filter. This only requires building and updating a single copy of the magnetic field map at each time instance. Figure 1 shows the learned magnetic field map and estimated trajectory from our EKF algorithm for magnetic field SLAM, tested on magnetic field measurements collected onboard a model ship. The resulting computational cost is  $O(N_m^2)$  at each time step, instead of  $O(N_m^2 N_p)$ . The use of the EKF is possible if the dynamic model and measurement model are close to linear [38]. In the case of simultaneous localization and mapping, the world frame coordinate system is defined relative to the initial body frame coordinate system [8]. As there is no uncertainty in the initial position estimate due to this definition [8], the position estimate initially has zero covariance. In cases where the growth of the uncertainty of the pose estimate that comes from odometry drift is limited by frequent enough visitations of previous areas, magnetic field SLAM remedies drift in the position estimate, which means that the position estimation error no longer grows without bounds, but stays limited [8]. When the estimated position is close to the actual position, the magnetic field linearized about the estimated position provides a good local approximation to the magnetic field itself, as the magnetic field even in environments with

strong stationary disturbances can be assumed to have limited spatial variability [8,11,20,39]. A key assumption for several implementations of estimating the magnetic field with GP regression is that the magnetic field in locations close to each other have a higher correlation than the magnetic field in locations that are further away [8,11,20,39]. In these implementations, how rapidly the correlation diminishes with increasing distance is encoded in a hyperparameter in the GP prior describing the length scale of the spatial variations in the magnetic field [39].



**Figure 1.** Learned magnetic field variations in test pool. The color corresponds to the estimated norm of the magnetic field map, while the opacity is inversely proportional with the variance of the estimate.

We illustrate with simulations that we can expect the EKF for a localization task to give accurate estimates when the max norm of the covariance from the predictive distribution is small relative to the length scale of the magnetic field anomalies. A requirement for using GP regression to represent the magnetic field map in magnetic field SLAM is the prior knowledge of hyperparameters describing the expected distribution of the magnetic field potential [8]. These hyperparameters contain information about the expected length scale of the magnetic field spatial variations [39]. Without adding any further assumptions to the magnetic field SLAM formulation presented in [8], we can therefore assume to have information available about how rapidly we can expect our learned magnetic field to vary spatially. In SLAM, the uncertainty of the position estimate will grow when the sensor moves through unexplored areas, as there is no map information available to correct the estimated pose [8]. When the sensor re-enters an area where it has already built a map of the local anomalies, our proposed algorithm can be expected to converge when the covariance of the position estimate is small compared to the length scale of the learned magnetic field map. We show with magnetic field data collected from a model ship in an indoor pool (see Figure 1) and simulated odometry that our proposed algorithm converges when the odometry noise is limited, for a trajectory when the time until the first revisitation of a mapped area is constant. We also show with magnetic field measurements and odometry obtained from an open-source implementation by [40] that our algorithm can compensate for drift in position estimates based on accelerometer and gyroscope measurements in a foot-mounted sensor.

The remainder of this paper is structured as follows. In Section 2, we define the model for our magnetic field SLAM estimation problem. In Section 3, we derive an EKF for magnetic field SLAM. In Section 4 we show the convergence properties of our

algorithms in a simulated navigation task, where we can control the ratio of our position estimate uncertainty over the length scale of the magnetic field variations. In Section 5, we demonstrate the drift-compensating abilities of the EKF-SLAM algorithm on a set of data we collected with a model ship and on an open-source data-set from a foot-mounted sensor. Finally, in Section 6 we summarise our findings and discuss possible directions for future work. Our Matlab-implementation producing all results found in this paper can be found on <https://github.com/fridaviset/EKFMagSLAM>.

## 2. Modeling

Our simultaneous localization and mapping algorithm estimates the filtering distribution

$$p(x_t | y_{1:t}^b, \Delta p_{1:t}^w, \Delta q_{1:t}^b), \quad (1)$$

where we denote the available magnetic field measurements by  $y_{1:t}^b$ , the odometry describing the change in position and orientation  $\Delta p_{1:t}^w, \Delta q_{1:t}^b$ , and the state we wish to estimate at each time step  $t$  as  $x_t$ . For a vector  $x_t$ , the set  $\{x_1, \dots, x_t\}$  is denoted  $x_{1:t}$  for brevity. We use the superscript  $b$  to denote the sensor's body-frame, which is aligned with its sensor axes. The superscript  $w$  refers to the world frame, which is defined as the inertial frame that shares its origin with the body frame at time zero. The gravity field in this position is aligned with the negative  $z$ -axis, and where the initial yaw-angle between the body and world-frame at  $t = 0$  is zero. As we wish to estimate both position, orientation and the magnetic field map, we model our state as

$$x_t = [(p_t^w)^\top \quad (q_t^{wb})^\top \quad m^\top]^\top, \quad (2)$$

where  $p_t^w$  denotes the position,  $q_t^{wb}$  denotes the orientation transformation from the world frame to the body frame, and  $m$  is a vector that describes our magnetic field map represented with reduced-rank GP regression.

### 2.1. Measurement Model

We consider the case where we have access to measurements of the magnetic field in a sensor attached to the object we aim to localize. The measurement equation is given by

$$y_t^b = R_t^{bw} \nabla_p \varphi(p_t^w) + e_{m,t}^b, \quad e_{m,t}^b \sim \mathcal{N}(0, \sigma_m^2 \mathcal{I}_3), \quad (3)$$

where  $y_t^b$  denotes the magnetic field measurement,  $e_{m,t}^b$  denotes the measurement noise and  $R_t^{bw}$  denotes the rotation from world to body frame, corresponding to the conjugate of the quaternion  $q_t^{wb}$ , expressed as a rotation matrix. See [41] for definitions of the quaternion conjugate, and definitions of transformation from a quaternion to a rotation matrix. The function  $\nabla_p \varphi(p_t^w)$  is our model of the magnetic field. We model the magnetic field as in [42] as the gradient of the function  $\varphi(p_t^w)$  with respect to the position  $p_t^w$ , where  $\varphi: \mathbb{R}^3 \rightarrow \mathbb{R}$  is a scalar potential distributed as a GP with prior

$$\varphi \sim \mathcal{N}(0, \kappa_{SE}(\cdot, \cdot) + \kappa_{lin}(\cdot, \cdot)), \quad (4)$$

and where the kernel is defined by the functions

$$\kappa_{SE}(p, p') = \sigma_{SE}^2 \exp\left(-\frac{\|p - p'\|_2^2}{2l_{SE}^2}\right), \quad (5a)$$

$$\kappa_{lin}(p, p') = \sigma_{lin}^2 p^\top p', \quad (5b)$$

with  $\sigma_{SE}$ ,  $\sigma_{lin}$ ,  $l_{SE}$  and  $\sigma_m$  being hyperparameters. The hyperparameter  $l_{SE}$  refers to the length scale of the spatial variations in the magnetic field potential that is represented by the kernel [42]. The parameters  $\sigma_{SE}$ ,  $\sigma_{lin}$  and  $\sigma_m$  define the presence of the nonlinear components, linear components and measurement noise in the magnetic field respectively [42].

The linear component modelled by the kernel component  $\kappa_{\text{lin}}$  represents of the constant underlying earth magnetic field, while the nonlinear disturbances caused by the modelled by the nearby ferromagnetic structures is modelled by the kernel component  $\kappa_{\text{SE}}$ . Modelling the magnetic field as the gradient of a scalar potential ensures that Maxwell's equations are satisfied, under the assumption that no current passes through the domain where we construct our magnetic field map [42]. We use a reduced-rank approximation to the GP similar to the one used in [8] for mapping the indoor magnetic field for localization purposes with the same kernel. Our approach is an application of the GP approximation presented in [37], which is based on conditioning the GP prediction on a set of basis functions corresponding to a subset of the eigenbasis of the negative Laplace operator in a finite domain, subject to Dirichlet boundary conditions [37]. The reduced-rank approximation models the magnetic field potential as a sum of basis functions defined as the solutions to the Laplace equations over a finite domain  $\Omega \subset \mathbb{R}^3$

$$\begin{cases} -\nabla_p^2 \phi_i(p) = \lambda_i \phi_i(p), & p \in \Omega, \\ \phi_i(p) = 0, & p \in \delta\Omega, \end{cases} \quad (6)$$

where  $\phi_i$  is the  $i$ 'th eigenfunction, and  $\lambda_i$  is the  $i$ 'th eigenvalue [37]. We approximate the GP with the first  $N_m$  basis functions solving the Laplace equations defined over a cubical domain  $\Omega = [L_{l,1}, L_{u,1}] \times [L_{l,2}, L_{u,2}] \times [L_{l,3}, L_{u,3}]$ . In this case, using the  $N_m$  first eigenfunctions to represent the potential gives the approximation

$$\varphi(p) \approx \Phi(p)m, \quad (7)$$

with  $\Phi(p)$  being the matrix

$$\Phi(p) = [p^\top \quad \phi_1(p) \quad \dots \quad \phi_{N_m}(p)], \quad (8)$$

where  $\phi_i$  is the  $i$ 'th eigenfunction of the Laplace basis, and  $m \in \mathbb{R}^{N_m+3}$  is a vector determining the contribution of each linear components as well as each basis function to the potential. Each eigenfunction  $\phi_i(p)$  has a closed form expression given by

$$\phi_i(p) = \prod_{d=1}^3 \frac{\sqrt{2}}{\sqrt{L_{u,d} - L_{l,d}}} \sin\left(\frac{\pi n_{i,d}(p_d + L_{l,d})}{L_{u,d} - L_{l,d}}\right), \quad (9)$$

where the set  $(n_{i,1}, n_{i,2}, n_{i,3})$  is the set of three natural numbers that is different from the sets  $(n_{j,1}, n_{j,2}, n_{j,3})$  defined for all  $j < i$ , that gives the corresponding eigenvalue

$$\lambda_i = \sum_{d=1}^D \left( \frac{\pi n_{i,d}}{L_{u,d} - L_{l,d}} \right)^2, \quad (10)$$

as large as possible. The basis functions in (9) and eigenvalues in (10) are identical to those used in [22]. The vector  $m$  has a prior distribution given by

$$m \sim \mathcal{N}(0, \Lambda), \quad (11)$$

where  $\Lambda$  is defined as

$$\Lambda = \text{diag} \left[ \sigma_{\text{lin}}^2 \mathcal{I}_3, \quad S_{\text{SE}}(\sqrt{\lambda_1}), \quad \dots, \quad S_{\text{SE}}(\sqrt{\lambda_{N_m}}) \right], \quad (12)$$

with  $S_{\text{SE}}(\cdot)$  being the spectral density of the squared exponential kernel, as defined in [7]. This corresponds to the magnetic field potential  $\varphi(p) \approx \Phi(p)m$  having a prior distribution given by (4) as  $N_m$  goes to infinity, and the size of the domain goes to infinity [37]. Inserting this approximation to the magnetic field model gives the measurement model

$$y_t^b \approx R_t^{\text{bw}} \nabla_p \Phi(p_t^w) m + e_{m,t}^b \quad e_{m,t}^b \sim \mathcal{N}(0, \sigma_m^2 \mathcal{I}_3), \quad (13)$$

with the closed form expressions for  $\nabla_p \Phi(p_t^w)$  given in Appendix A. This measurement model is identical to the measurement model used in [8], with the exception of the basis functions  $\Phi(p_t^w)$ , which are different as they are defined with respect to different domains.

## 2.2. Dynamic Model

We assume access to noisy odometry measurements  $\Delta p_t^w$  and  $\Delta q_t^b$  of the change in position and orientation at each time step. We model the change in position and orientation according to the dynamic model

$$p_{t+1}^w = p_t^w + \Delta p_t^w + e_{p,t}^w, \quad e_{p,t}^w \sim \mathcal{N}(0, \sigma_p^2 \mathcal{I}_3), \quad (14a)$$

$$q_{t+1}^{wb} = q_t^{wb} \odot \Delta q_t^b \odot \exp_q(e_{q,t}^b), \quad e_{q,t}^b \sim \mathcal{N}(0, \sigma_q^2 \mathcal{I}_3), \quad (14b)$$

where  $e_{p,t}^w$  and  $e_{q,t}^b$  denote the position and orientation odometry measurement noise respectively,  $\odot$  is the quaternion product and  $\exp_q$  is the operator that maps an axis-angle orientation deviation to a quaternion (see [41] for details on quaternion algebra).

## 3. EKF for Magnetic Field Slam

We estimate our state with an EKF applied to the dynamic model in (14a,b) and the measurement model defined in (13), with predictive and filtered estimates denoted  $\hat{p}_{t|t-1}^w$ ,  $\hat{m}_{t|t-1}$ ,  $\hat{q}_{t|t-1}^{wb}$  and  $\hat{p}_{t|t}^w$ ,  $\hat{m}_{t|t}$ ,  $\hat{q}_{t|t}^{wb}$  respectively. We initialise the magnetic field state estimate as  $\hat{m}_{0|0} = 0_{N_m \times 1}$  according to the reduced-rank GP prior in (11). We initialise the orientation estimate according to the initial rotation  $\hat{q}_{0|0}^{wb} = q_0^{wb}$  between the world and body frame as defined in Section 2. We initialise the position estimate as  $\hat{p}_{0|0}^w = 0_{3 \times 1}$ , also according to our definition of the world frame relative to the initial body frame from Section 2.

We represent the deviation between the true and estimated predictive state by an error state  $\zeta_t$  defined as

$$\zeta_t = [(\delta_t^w)^\top \quad (\eta_t^w)^\top \quad v_t^\top]^\top, \quad (15)$$

where  $\delta_t^w = p_t^w - \hat{p}_{t|t-1}^w$  denotes the position estimation error,  $v_t = m - \hat{m}_{t|t-1}$  denotes the magnetic field state estimation error, and where  $\eta_t^w$  denotes the orientation estimation error parametrised as an axis-angle deviation according to

$$q_t^{wb} = \exp_q(\eta_t^w) \odot \hat{q}_{t|t-1}^{wb}. \quad (16)$$

Similarly, we represent the deviation between the true and estimated filtered state by an error state  $\tilde{\zeta}_t$  defined as

$$\tilde{\zeta}_t = [(\tilde{\delta}_t^w)^\top \quad (\tilde{\eta}_t^w)^\top \quad \tilde{v}_t^\top]^\top, \quad (17)$$

where  $\tilde{\delta}_t^w = p_t^w - \hat{p}_{t|t}^w$ ,  $\tilde{v}_t = m - \hat{m}_{t|t}$ , and where  $\tilde{\eta}_t^w$  denotes the filtered orientation estimation error according to

$$q_t^{wb} = \exp_q(\tilde{\eta}_t^w) \odot \hat{q}_{t|t}^{wb}. \quad (18)$$

Since we build our map relative to our initial position and orientation, the covariance of our initial position and orientation estimates is zero. The covariance of the initial magnetic field estimate is defined in (11) as the magnetic field map prior  $\Lambda$ . Hence, our initial error state  $\zeta_0$  has a covariance

$$P_{0|0} = \begin{bmatrix} 0_{6 \times 6} & 0_{6 \times (N_m+3)} \\ 0_{(N_m+3) \times 6} & \Lambda \end{bmatrix}. \quad (19)$$

To perform the dynamic update, we propagate our filtered state estimate through the nonlinear dynamic model (14a,b), giving the predictive updates as described in (26a,b). As the magnetic field is assumed stationary, its estimate is unchanged by the dynamic update defined in (26c). We derive the covariance update in the EKF by linearising about the filtered state estimate from the previous time step, with respect to the error state  $\tilde{\zeta}_t$ . Inserting (15), (17) and (26a,b) into the dynamic model (14a,b) gives

$$\hat{p}_{t|t-1}^w + \delta_t^w = \hat{p}_{t-1|t-1}^w + \Delta p_t^w + \tilde{\delta}_{t-1}^w + e_{p,t}^w \quad (20a)$$

$$\exp_q(\eta_t^w) = \exp_q(\tilde{\eta}_{t-1}^w) \odot \hat{q}_{t|t-1}^{wb} \odot \exp_q(e_{q,t}^b) \odot (\hat{q}_{t|t-1}^{wb})^C, \quad (20b)$$

where  $(\hat{q}_{t|t-1}^{wb})^C$  denotes the conjugate of the quaternion  $\hat{q}_{t|t-1}^{wb}$ . The linearization of the dynamic model with respect to the error states gives the following propagation of the error states

$$\delta_t^w = \tilde{\delta}_{t-1}^w + e_{p,t}^w, \quad (21a)$$

$$\eta_t^w \approx \tilde{\eta}_{t-1}^w + \hat{R}_{t|t-1}^{wb} e_{q,t}^b, \quad (21b)$$

$$v_t = \tilde{v}_{t-1}, \quad (21c)$$

where  $\hat{R}_{t|t-1}^{wb}$  denotes the rotation matrix corresponding to the rotation represented by the quaternion  $\hat{q}_{t|t-1}^{wb}$ . This linearization is exact for the position, and it is a good approximation for the orientation error state in the cases where the orientation error is small [41]. Equation (21a,b) can equivalently be written as

$$\tilde{\zeta}_t \approx \tilde{\zeta}_{t-1} + e_{\text{dyn},t}, \quad e_{\text{dyn},t} \sim \mathcal{N}(0_{(N_m+9) \times 1}, Q), \quad (22)$$

where

$$Q = \begin{bmatrix} \sigma_p^2 \mathcal{I}_3 & 0_{3 \times 3} & 0_{3 \times (N_m+3)} \\ 0_{3 \times 3} & \sigma_q^2 \mathcal{I}_3 & 0_{3 \times (N_m+3)} \\ 0_{(N_m+3) \times 3} & 0_{(N_m+3) \times 3} & 0_{(N_m+3) \times (N_m+3)} \end{bmatrix}. \quad (23)$$

As the linearization of the error state propagation is given in (22), the covariance  $P_{t|t-1}$  of the predictive state error  $\tilde{\zeta}_t$  is given by (26d).

For the measurement update, we apply an EKF measurement update to the measurement model in (13). We linearise about the predictive state estimate, with respect to the error state  $\tilde{\zeta}_t$ . The linearized measurement model is given by

$$y_t^b = \hat{R}_{t|t-1}^{bw} \nabla_p \Phi(\hat{p}_{t|t-1}^w) \hat{m}_t + H_t \tilde{\zeta}_t + e_{m,t}^b, \quad e_{m,t}^b \sim \mathcal{N}(0_{3 \times 1}, \sigma_m^2 \mathcal{I}_3), \quad (24)$$

where

$$H_t = \begin{bmatrix} \left( \nabla_{pp} \Phi(\hat{p}_{t|t-1}^w) \hat{m}_t \right)^\top \\ \left[ \left( \nabla_p \Phi(\hat{p}_{t|t-1}^w) \hat{m}_t \right) \times \right]^\top \\ \left( \nabla_p \Phi(\hat{p}_{t|t-1}^w) \right)^\top \end{bmatrix}^\top \quad (25)$$

with  $[v \times]$  being the scew-symmetric matrix representing the cross product  $v \times u$  between two vectors  $v, u \in \mathbb{R}^3$  as a matrix multiplication  $[v \times]u$  (explicit expression for  $[v \times]$  is given in [41]), and  $\nabla_{pp} \Phi(\cdot)$  being the Jacobian of the basis functions, given in Appendix A. Note that this Jacobian is a matrix with  $3 \times 3 \times (N_m + 3)$  entries, and multiplying it with the  $(N_m + 3)$ -dimensional state vector  $m$  therefore gives a  $3 \times 3$  matrix. Applying the Kalman filter measurement update to this linearized measurement function gives the EKF measurement update in (27a–e). This gives an estimate  $\hat{\zeta}_t$  of the predictive state error  $\tilde{\zeta}_t$ , with a corresponding covariance. By concatenating the estimated predictive state error to the predictive state, we obtain the filtered state estimates as defined in (28a–c). The covariance of the filtered error state  $\hat{\zeta}_t$  then becomes the same as the covariance of the



estimated predictive error state  $\hat{\xi}_t$ . Recursively applying the dynamic update, measurement update and re-linearization step results in Algorithm 1.

---

**Algorithm 1:** EKF for magnetic field SLAM

---

Input:  $\{\Delta p_t^w, \Delta q_t^b, y_t^b\}_{t=1}^N$

Output:  $\{\hat{p}_{t|t}^w\}_{t=1}^N, \{\hat{q}_{t|t}^{wb}\}_{t=1}^N, \{\hat{m}_{t|t}\}_{t=1}^N$

Initialisation:  $\hat{p}_{0|0}^w = 0_{3 \times 1}, \hat{q}_{0|0}^{wb} = q_0^{wb}, \hat{m}_{0|0} = 0_{(N_m+3) \times 0}$ . (19)

1: **for**  $t = 1$  to  $N$  **do**

2: Dynamic update

$$\hat{p}_{t|t-1}^w = \hat{p}_{t-1|t-1}^w + \Delta p_t^w \quad (26a)$$

$$\hat{q}_{t|t-1}^{wb} = \hat{q}_{t-1|t-1}^{wb} \odot \Delta q_t^b \quad (26b)$$

$$\hat{m}_{t|t-1} = \hat{m}_{t-1|t-1} \quad (26c)$$

$$P_{t|t-1} = P_{t-1|t-1} + Q \quad (26d)$$

3: Measurement update

$$z_t = \hat{R}_{t|t-1}^{wb} y_t^b - \nabla \Phi_P(\hat{p}_{t|t-1}^w) \hat{m}_{t|t-1} \quad (27a)$$

$$S_t = H_t P_{t|t-1} H_t^\top + \sigma_m^2 \mathcal{I}_3 \quad (27b)$$

$$K_t = P_{t|t-1} H_t^\top S_t^{-1} \quad (27c)$$

$$\hat{\xi}_t = K_t z_t \quad (27d)$$

$$P_{t|t} = P_{t|t-1} - K_t S_t K_t^\top \quad (27e)$$

4: Relinearization

$$\hat{p}_{t|t}^w = \hat{p}_{t|t-1}^w + \hat{\delta}_t^w \quad (28a)$$

$$\hat{q}_{t|t}^{wb} = \exp_q(\hat{\eta}_t^w) \odot \hat{q}_{t|t-1}^{wb} \quad (28b)$$

$$\hat{m}_{t|t} = \hat{m}_{t|t-1} + \hat{v}_t \quad (28c)$$

5: **end for**

---

#### 4. Simulations

We study when Algorithm 1 for localization in a previously learned magnetic field gives a converging pose estimate in a known nonlinear field depending on the position uncertainty at time  $t$ . Since we assume the magnetic field map is known, we can replace  $\hat{m}_{t|t}$  with the known  $m_{\text{sim}}$  everywhere in our algorithm. As a consequence of this, we can also skip (26c) and (28c), use  $P_{0|0} = 0_{6 \times 6}$  and

$$H_t = \begin{bmatrix} \left( \nabla_{PP} \Phi(\hat{p}_{t|t-1}^w) m \right)^\top \\ \left[ \left( \nabla_P \Phi(\hat{p}_{t|t-1}^w) m \right) \times \right]^\top \end{bmatrix}^\top. \quad (29)$$

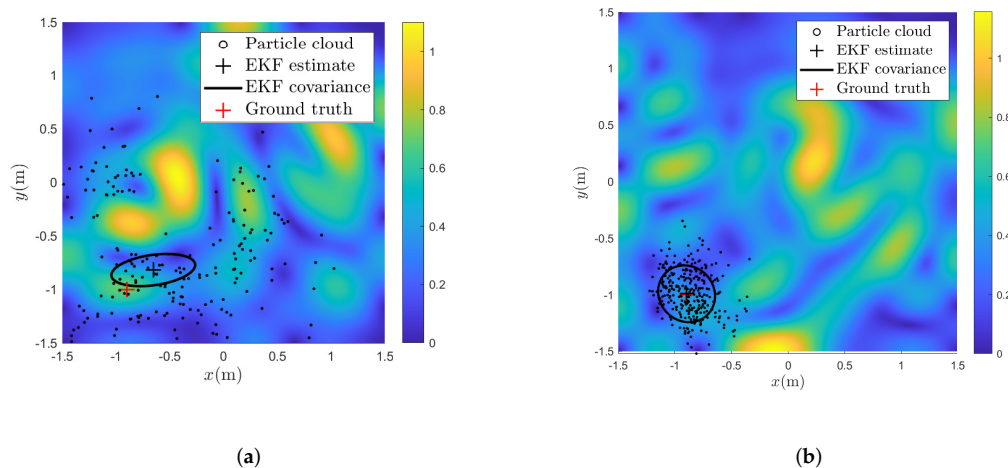
We start the simulation at time  $t$  with a varying predictive position estimation error, and set the standard deviation of our position uncertainty equal to the distance between the actual and estimated position at the beginning of the simulation. This artificially introduces a predictive position estimation error representing the estimation error that can accumulate over time in magnetic field SLAM. Position errors can, for example, accumulate when the

sensor is moved for a long time through areas with no information about the magnetic field available from previous measurements [8].

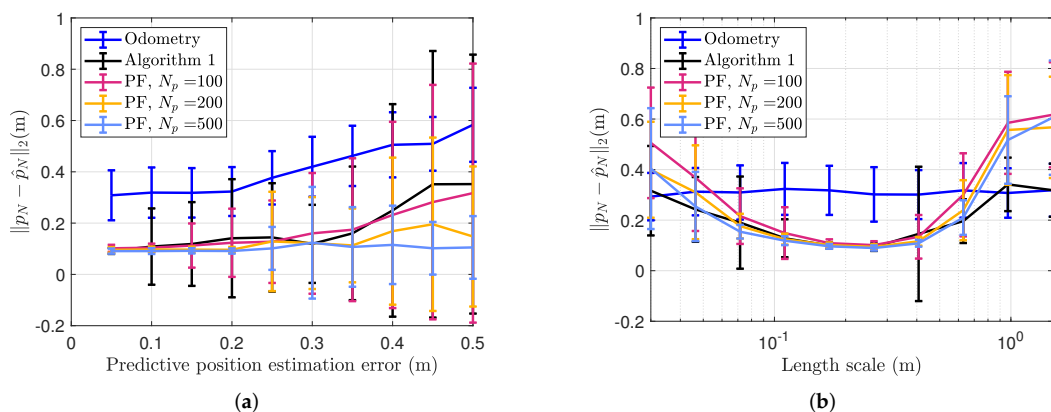
We simulate positions  $p_t$  along a square trajectory moving with constant velocity for four laps, and simulate the odometry by adding sampled realisations of the white noises  $e_{p,t}^w$  and  $e_{q,t}^b$  to (14a,b). We simulate a nonlinear field by drawing a sample from the reduced-rank GP prior  $m_{\text{sim}} \sim \mathcal{N}(0, \Lambda)$ , with  $\sigma_{\text{lin}} = 0$ ,  $\sigma_{\text{SE}} = 0.1$  and  $l_{\text{SE}} = 0.2$ . We used 50 basis functions to represent the magnetic field map. Using a domain that is  $3 \text{ m} \times 3 \text{ m} \times 1 \text{ m}$ . This number of basis functions ensures that for a GP trained with 2000 sampled measurements the root mean squared error (RMSE) between the approximate and the full GP predictions in 1000 randomly selected locations is below the measurement noise. To prevent ill effects from the boundary conditions we ensured that both the training and test data was at least 0.5 m away from the border. We then simulate magnetic field measurements by adding white noise to the gradient of the nonlinear field in the ground truth position according to (13), replacing  $\hat{m}_t$  with  $m_{\text{sim}}$ , and using  $\sigma_m = 0.03$ . We use the incoming magnetic field measurements and the odometry to estimate the position and orientation using an EKF and a particle filter. The EKF is implemented according to Algorithm 1, but reducing the state-space to only contain the position and orientation, and inserting  $m_{\text{sim}}$  in place of  $\hat{m}_t$  in (27a). We implement a particle filter for navigation in a magnetic field represented by a field learned with GP regression according to Algorithm 1 in [20], with the difference that we use the simulated reduced-rank map instead of a learnt full GP map, and that we perform the prediction step using our odometry model in (14a,b).

In Figure 2, we can see two examples of PFs and an EKF's estimate of the position filtered distribution, represented with a particle cloud and a mean and an uncertainty interval, respectively. In Figure 2a, the initial uncertainty of the position estimate is so large that the particle cloud becomes multi-modal, making it impossible for the EKF to correctly approximate the true nature of the filtered distribution. The estimated position is, therefore, far away from the true position. In addition, the uncertainty estimate of the EKF does not reflect this, as it relies upon a linearization of the nonlinear magnetic field about the predictive estimate. In Figure 2b, the position estimate at time  $t$  is still wrong, but close enough that the particle cloud representation of the filtered distribution appears uni-modal, and the EKF estimate of the filtered distribution is now closer to the estimate from the particle filter. As we see in Figure 3a displaying the position estimation error at the end of the trajectory estimates across the four laps, the position estimates from the EKF are accurate across the entire trajectory if the predictive position error is lower than 0.3 m. The particle filter on the other hand, is accurate even beyond these predictive position accuracies when using 500 particles, while it is only slightly improving the prediction accuracy over Algorithm 1 when using 100 or 200 particles. The accuracy is better for 200 particles than for 100 particles. For only 100 particles, the average prediction accuracy is worse than for Algorithm 1 for a predictive position accuracy of 0.2 m, likely due to the fact that the particle filter is a Monte-Carlo method, meaning that there is never a guarantee for convergence [43]. In Figure 3b, the average estimation accuracy for varying length scales of the simulated magnetic field is displayed. As in Figure 3a, the performance of the particle filter improves with increasing amount of particles. Algorithm 1 is able to compensate for odometry drift and achieve estimation error on average below 0.2 m for length scales between 0.1 and 0.4 m, using a constant predictive position error of 0.1 m. For length scales below 0.1 m, the linearization error becomes too big for the approximation accuracy of our linearized model to give a good result. For length scales higher than 0.4 m, the variations in the magnetic field are not rich enough to provide valuable information about the position of the sensor. Therefore, as the length scale of the field increases, even though the field becomes closer to linear and the linearization error continues to decrease, the estimation accuracy does not improve—because the signal-to-noise ratio from the magnetic field measurements also decreases. As we for the simulation results in Figure 3a use a simulated magnetic field map with spatial variations of 0.3 m, this indicates that as long as the covariance of the predictive distribution does not exceed the length scale of

the magnetic field, we can expect Algorithm 1 to have the same estimation accuracy as particle-based methods.



**Figure 2.** Comparison of approximations of the filtered position distribution given measurements from a simulated nonlinear field. The color indicates the norm of the simulated magnetic field. The covariance ellipsoids indicate the 68% confidence interval of the EKF estimate. (a) Estimates of the filtered distribution based on predictive estimates with error 0.40 m. (b) Estimates of the filtered distribution based on predictive estimates with error 0.05 m.



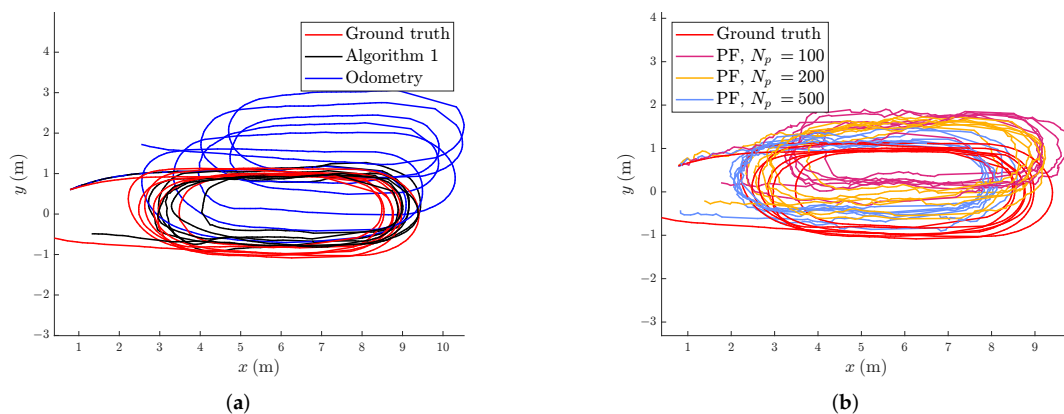
**Figure 3.** Simulation, investigating drift-compensating abilities given varying predictive position estimation errors. Comparison of position estimation error at the end of the trajectory between Algorithm 1 and a particle filter for localization in a known map with varying predictive position errors at the initialisation of the simulation. The lines connect the average results after 100 Monte Carlo repetitions with different realisations of the odometry noise, and the error bars represent one standard deviation. (a) Estimation accuracies with varying predictive position error. (b) Estimation accuracies with varying length scales  $l_{SE}$ .

## 5. Experimental Results

### 5.1. Model Ship Experiments

We performed experiments to test Algorithm 1 on a model ship in a pool. The magnetic field on the model ship was measured using an Xsens MTi-100 Inertial Measurement Unit (IMU). We recorded the ground truth position and orientation using a motion capture system with cameras and optical markers mounted on the model ship, as shown in Figure 1. The motion capture markers and the IMU were rigidly attached to the ship. The IMU measurements were collected on a computer onboard the ship. The magnetic field was disturbed by metal railings and building structures near the pool. We steered the model

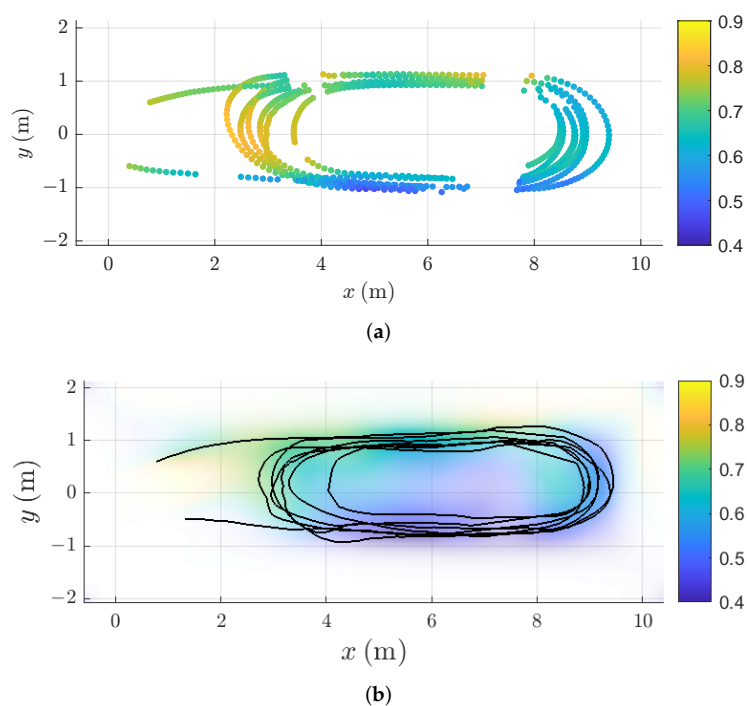
ship around in long loops in the pool, with a ground truth trajectory that is displayed in red in Figure 4. The magnetic field measurements were collected in the IMU at 200 Hz, and down-sampled to 5 Hz. The odometry was simulated based on the ground truth position and orientation, according to the odometry model in (14a,b), also at a frequency of 5 Hz (but with some motion capture measurement dropouts due to pool reflections). We simulate drifting odometry by computing the change in position and orientation at each time step from the ground truth and adding a simulated white noise with standard deviation  $\sigma_p = 0.01$ ,  $\sigma_q = 0.001$ . For these experiments, we use real magnetic field data and simulated odometry to investigate the effects of changing odometry noise on our algorithm. In addition, we simulate a constant position odometry bias of  $[0.003 \ 0.003 \ 0]$  m/time step. Our algorithm was not originally designed to compensate for constant position odometry biases. However, as this often occurs in practice (for example, when the odometry sensors are not perfectly calibrated), we chose to include it in our simulated odometry to test our algorithms' ability to compensate for a drift that consists both of integrated white noise and a constant disturbance.



**Figure 4.** Comparison of the model ship position trajectory estimates for a single realisation of simulated odometry noise from a birds-eye view. (a) Comparing Algorithm 1 and the odometry to the ground truth. (b) Comparison of the position estimates from the RBPF with 100, 200 and 500 particles respectively to the ground truth.

In magnetic field SLAM, there is usually no possibility of optimising the hyperparameters for GPs prior to estimating the map, as there is no magnetic field data available. This motivates us to choose hyperparameters prior to running Algorithm 1 [8]. The hyperparameters were set to  $l_{SE} = 0.8$ ,  $\sigma_{SE}^2 = 1$ ,  $\sigma_m^2 = 0.01$ ,  $\sigma_{lin}^2 = 1$  and 50 basis functions. The basis functions were defined with respect to a cubical domain  $\Omega$  which is as small as possible, and whose border is at least 1 m away from the closest ground truth position. We confirmed empirically that 50 basis functions is sufficient to ensure that the RMSE between the approximation and the full GP predictions in 1000 randomly selected locations in the domain is below the measurement noise. We based the predictions on 2000 samples from the full GP prior, sampled from randomly selected locations in the domain at least 1 meters away from the domain border. For the first set of experiments, we investigate and compare the position estimation error of Algorithm 1 with the particle filter-based approach to magnetic field SLAM [8] for odometry noise levels of  $\sigma_p = 0.01$ ,  $\sigma_q = 0.001$ . In Figure 5a, the norm of the magnetic field measurements in locations where there were no motion-capture dropouts are displayed. The norm of the measured magnetic field ranges between 0.46 and 0.83 (the Xsens MTi-100 provides unit-less measurements proportional to the magnetic field strength). The spatial variations in the measured magnetic field norm are visible in Figure 5a, and the magnetic field stays close to constant for position changes of less than 0.1 m, while it can change as much as from 0.46 to 0.83 when the position change is more than 1 meter. As the spatial variations in the magnetic field potential are the sources of the spatial variations in the magnetic field norm [39], we expect that our

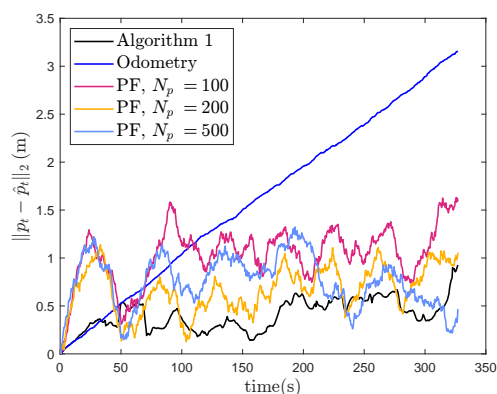
assumed length scale of 0.8 m is close enough to the actual length scale of the magnetic field variations to allow for Algorithm 1 to compensate for position estimation drift in the odometry. Figure 5b shows the norm of the magnetic field map learned by Algorithm 1, and comparing to the measured magnetic field norms in Figure 5a, we can see that the norm of the magnetic field estimates and the estimated trajectory are similar, with the learned magnetic field map prediction being more certain in and near the areas where there are more magnetic field measurements available. In Figure 4a, the estimated trajectory from Algorithm 1 is compared with the ground truth trajectory, as well as the dead reckoning position estimate from the simulated odometry. The position estimate from Algorithm 1 compensates visibly for the drift in the odometry.



**Figure 5.** Measured and estimated magnetic field and position trajectories for the model ship. The upper plot marks with circles the locations where magnetic field measurements were successfully collected and matched with a ground truth position in the model ship, and the colors of the circles correspond to the norm of the measured magnetic field. The lower plot displays the trajectory estimate from applying Algorithm 1 in black. It also shows the learned magnetic field map, where the color corresponds to the norm of the estimated magnetic field  $\|\nabla_p \Phi(p) \hat{m}_{N|N}\|_2$ , and the opacity is inversely proportional with the trace of the covariance matrix of the magnetic field map estimate in each location,  $\text{Tr}(\nabla_p \Phi(p) P_{N|N} (\nabla_p \Phi(p))^T)$ . (a) Measured magnetic field norm in ground truth positions. (b) The estimated magnetic field norm is displayed with the semi-transparent color map and the estimated trajectory is displayed with the black line.

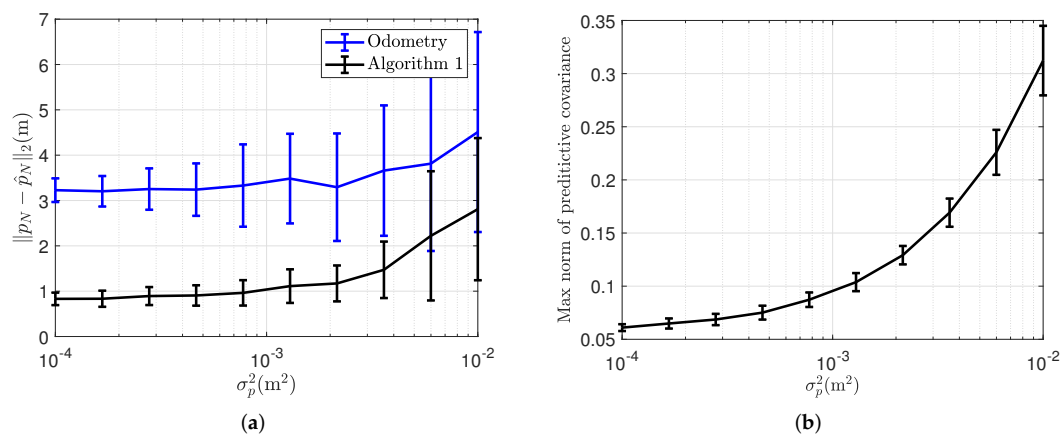
In Figure 6 the position estimation error of the odometry and Algorithm 1 can be seen to increase at the beginning of the trajectory. After around 31 s, Algorithm 1 can use the learned magnetic field map in combination with the incoming magnetic field measurements to compensate for drift in the estimated position and orientation. In Table 1, the position estimation RMSE values for 4 collected data sets of a similar shape as the one displayed in Figure 4a is shown after repeated experiments with the same odometry noise and the same constant drift in the xy-plane, showing that the reduction of RMSE is comparable also for repeated experiments. The position estimation error of the dead reckoning can increase potentially unbounded, while the position estimation error of Algorithm 1 remains bounded when the ship revisits previously mapped areas. However, it will only remain bounded if the quality of the map is good enough to provide information to the position

estimate. In Table 2, the runtime for each of the algorithms is displayed. The runtime of the PF methods grows proportionally with the number of particles used. As can be seen in Table 1, the PF performs on average worse than the EKF. The trajectory of the highest-weight particle from a single run of each of the PFs is shown in Figure 4b. From our simulation results, given enough particles, the particle filter compensates for drift at least as well as Algorithm 1. However, in contrast to our simulation results where we investigate how well the particle filter performs localization given a previously learned map, in practice, the particle filter has to rely upon magnetic field maps created conditionally on each particle. From these results a standard implementation of the PF in [8] with cubic domain basis functions and resampling at every timestep, with the same hyperparameter settings as the EKF performs worse on our collected model ship data, even given 500 particles. In general, the performance of the particle filter can depend on the resampling strategy, the measurement noise and the process noise [44]. Another possible explanation why the particle filter performs worse for the full SLAM scenario compared to the simulation case, is the fact that for long trajectories, the resampling step can cause loss of diversity amount the particles [45].



**Figure 6.** Comparison of model ship position estimation errors from Algorithm 1, drifting odometry and the PF with 100, 200 and 500 particles respectively for a single realisation of simulated odometry noise.

The linearization of the measurement function in (13) is performed around the predictive position estimate. The covariance of the estimate can grow when the sensor is moved through an area where the map is previously unknown. The growth rate will depend on the odometry noise. This is demonstrated in Figure 7b, where for 100 Monte Carlo simulations with different odometry noise realisations, the max norm of the predictive estimate from Algorithm 1 can be seen to increase with increasing odometry noise for the same trajectory. In Figure 7a, it can be seen that if we increase the simulated odometry noise above  $\sigma_p^2 = 0.002$ , the position estimation error of Algorithm 1 is no longer able to compensate for drift in the dead reckoning. A higher odometry noise means more drift is likely to accumulate to the predictive estimation error before revisiting a previously mapped area. The inability of Algorithm 1 to compensate for drift caused by odometry noises above  $\sigma_p^2 = 0.002$ , therefore, reflects how the assumptions of the measurement function being locally linear no longer hold when the covariance of the predictive distribution becomes large compared to the length scale of the magnetic field disturbances. For the experimental results, for odometry noises above  $\sigma_p^2 = 0.002$ , we observe a predictive covariance max norm of 0.15 m in the results in Figure 7b and an accumulated drift of 0.5 m, which is combination is comparable in magnitude to our length scale  $l_{SE}$  of 0.8 m. These results are comparable to our simulation results in Section 4, where Algorithm 1 for localization only converges when the position error is 0.3 m using a length scale of 0.2 m. In both cases, when the order of the prediction error goes beyond the length scale of the magnetic field variations, Algorithm 1 is no longer able to compensate for drift in the position estimate.



**Figure 7.** Investigation of the effect of varying odometry noise on the model ship position estimate. The lines connect the average results of the ship position estimation after 100 Monte Carlo repetitions with different realisations of the simulated odometry for varying amounts of odometry noise. (a) Model ship position estimation error at the end of the trajectory for varying amounts of odometry noise. (b) The max norm of the predictive covariance of the estimate from Algorithm 1 depending on varying odometry noise.

**Table 1.** Trajectory RMSE values in meters for the 4 collected data sets from the model ship. Values are given as averages  $\pm$  one standard deviation, after 100 Monte Carlo repetitions with different realisations of the simulated odometry noise.

Trajectory RMSEs	Data Set 1	Data Set 2	Data Set 3	Data Set 4
Algorithm 1	0.53 $\pm$ 0.15	0.58 $\pm$ 0.18	0.53 $\pm$ 0.24	0.98 $\pm$ 0.62
RBPF with 100 particles	0.85 $\pm$ 0.27	0.92 $\pm$ 0.26	0.95 $\pm$ 0.42	1.53 $\pm$ 0.53
RBPF with 200 particles	0.85 $\pm$ 0.22	0.89 $\pm$ 0.27	0.98 $\pm$ 0.47	1.48 $\pm$ 0.46
RBPF with 500 particles	0.87 $\pm$ 0.19	0.86 $\pm$ 0.24	1.00 $\pm$ 0.40	1.53 $\pm$ 0.50
Odometry	1.98 $\pm$ 0.54	1.52 $\pm$ 0.48	1.76 $\pm$ 0.52	1.65 $\pm$ 0.47

**Table 2.** Measured time to run the estimation algorithm (in seconds) for the 4 collected data sets from the model ship. Values are given as averages  $\pm$  one standard deviation, after 100 Monte Carlo repetitions with different realisations of the simulated odometry noise.

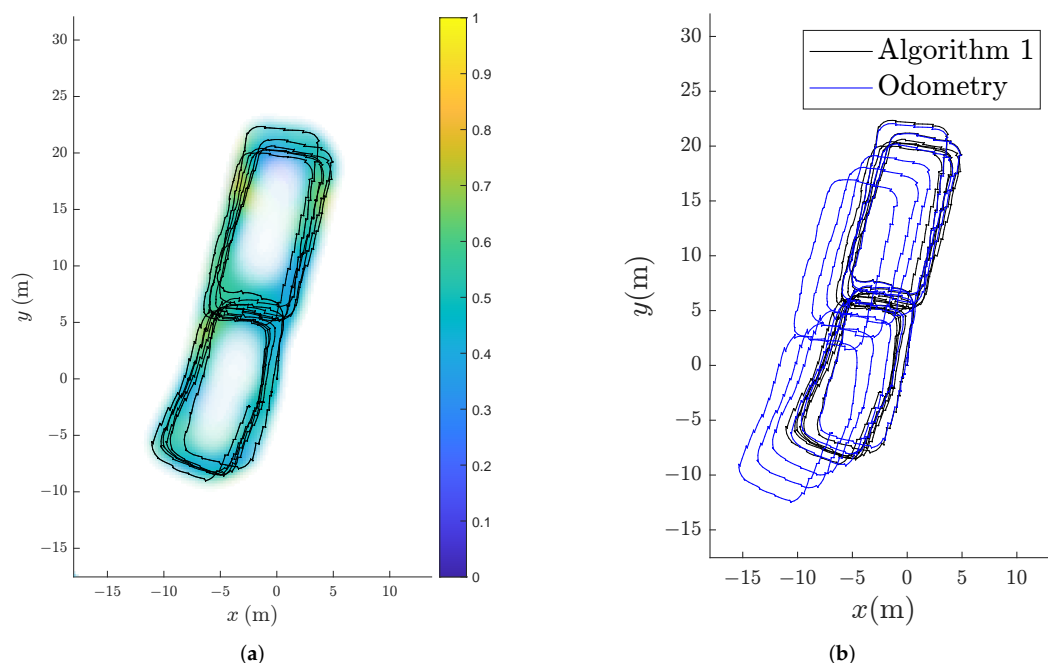
Runtimes	Data Set 1	Data Set 2	Data Set 3	Data Set 4
Algorithm 1	0.06 $\pm$ 0.01	0.05 $\pm$ 0.00	0.14 $\pm$ 0.01	0.05 $\pm$ 0.00
RBPF with 100 particles	12.85 $\pm$ 0.26	9.24 $\pm$ 0.10	21.55 $\pm$ 0.32	9.91 $\pm$ 0.14
RBPF with 200 particles	25.70 $\pm$ 0.49	18.44 $\pm$ 0.166	42.64 $\pm$ 0.27	19.72 $\pm$ 0.20
RBPF with 500 particles	64.24 $\pm$ 0.82	46.02 $\pm$ 0.20	106.93 $\pm$ 0.34	49.43 $\pm$ 1.29

## 5.2. Magnetic Field Slam for Pedestrians with Foot-Mounted Sensor

Using accelerometer and gyroscope measurements from an IMU mounted on the foot of a pedestrian, it is possible to estimate the position of the pedestrian with high accuracy on a short timescale using a zero-velocity-update (ZUPT) aided EKF [46]. The estimate is obtained by integrating the change in orientation and velocity. In addition, the assumption that when the foot is in the stationary part of the step, it has zero velocity is used to reduce the drift of the position and orientation estimates [46]. The position and orientation estimates are typically accurate at the beginning of a trajectory but can drift over time if biases and/or white noise affect the measurements [47]. This filter was implemented in an open-source implementation by [47].

To test the capabilities of Algorithm 1 on measurements from a foot-mounted sensor, we used magnetic field measurements present in the open-source data set used in [40], to remedy the drift present in the position estimates obtained from their ZUPT-aided EKF. The data set from [40] contains multiple measurement series from an IMU collected in the same hallway, walking the same trajectory. Although the implementation in [40] only used the accelerometer and gyroscope measurements from the IMU, magnetic field measurements were also collected, and are included in the published data. We first ran the open-source implementation from [40] of the ZUPT-aided EKF on the 12 available measurement sequences that were made by collected while walking in a similar trajectory. We ran the ZUPT-aided EKF independently on the 12 experiments and obtained 12 sets of position and orientation estimates. We then concatenated the 12 estimated trajectories by initialising each trajectory at the position and orientation where the previous trajectory ended. This gave a drifting odometry estimate of the position of the pedestrian. The drifting odometry is displayed in Figure 8b. This odometry has an increasing error in position and orientation over time partly because the ZUPT-aided EKF will have some position and orientation drift inherently and partly because of the assumption that the foot-mounted sensor ends in the same orientation at the end of each collected data set as the beginning of the next data set may not be exactly true. However, we can see that most drift accumulates in a constant direction, and drift caused by wrong orientation initialisation should cause twisting of the trajectory. It is, therefore, likely that most of the drift visible in Figure 8b is present due to inherent drift in the ZUPT-aided EKF. To use the odometry in Algorithm 1, we down-sampled the position and orientation estimates to 10 Hz and computed the change in position and orientation between each time step. We then used the changes in position and orientation as input odometry. As SLAM is performed in real-time, we cannot know the hyperparameters a priori to running the algorithm. The magnetic field measurements available in the open-source data set from [40] were without reported units but had a norm that ranged between 0.2429 and 0.8584. We therefore selected the expected nonlinear variations  $\sigma_{SE}^2 = 1$ . We assumed that the contribution from the constant earth magnetic field had approximately the same order of magnitude and so selected  $\sigma_{lin}^2 = 1$ . We set the length scale to  $l_{SE} = 2$  m, and we set the measurement noise to be  $\sigma_m^2 = 0.01$ . We used 1850 basis functions to approximate the magnetic field map. We selected a domain which was the smallest possible cube that was still at least 10 meters away from the first lap of the odometry. We found empirically that 1850 were a sufficient amount of basis functions using the same approach as in Sections 4 and 5. The resulting position estimate from Algorithm 1 compensates for drift in the odometry, as shown in Figure 8a.





**Figure 8.** Trajectory and magnetic field map estimate for the foot-mounted sensor data. The estimated trajectory obtained with Algorithm 1 is compared to odometry from the foot-mounted sensor data obtained via [40] implementation of the ZUPT-aided EKF using a foot-mounted accelerometer and gyroscope. The color of the magnetic field map corresponds to the norm of the estimated magnetic field, and the opacity is inversely proportional with the sum of the marginal variance for each of the three estimated magnetic field components. (a) Learned magnetic field displayed with the semi-transparent color map and estimated trajectory displayed with the black line with odometry from foot-mounted sensor from a birds eye view. (b) Trajectory estimate from Algorithm 1 compared to odometry from a birds eye view.

## 6. Conclusions and Future Work

We proposed using an EKF for magnetic field SLAM, which is computationally more efficient and requires less memory than previously proposed methods for magnetic field SLAM. Promisingly, we demonstrated that our proposed algorithm compensates for odometry drift in a way that is comparable to previously proposed, more computationally expensive methods. Using an experiment with magnetic field measurements collected onboard a model ship and using simulated odometry, we ran Monte-Carlo simulations investigating the capabilities of our algorithm to compensate for odometry drift for varying amounts of odometry noise, illustrating that when the uncertainty of the estimate is small compared to the length scale of the magnetic field variations, our proposed algorithm will give a position estimate that compensates for drift in odometry. We also demonstrated the abilities of our proposed algorithm to compensate for drift on an open-source data-set collected with a foot-mounted sensor.

To employ our proposed algorithm in real-life applications such as indoor, surface, underground or underwater navigation, it would be necessary to incorporate sources of odometry information that are available in real-life scenarios, such as inertial sensors or visual odometry from cameras. Another possible direction of future work could be to implement an iterated EKF or another extended Kalman-filter based estimation method that can handle larger non-linearities compared to the EKF [48], and investigate if this improves the convergence of the method. Future research could also look into further reducing the computational requirements associated with reduced-rank GP regression.

**Author Contributions:** Conceptualization, M.K. and F.V.; methodology, F.V.; software, F.V.; formal analysis, F.V.; data curation, and F.V.; writing—original draft preparation, M.K. and F.V.; writing—review and editing, M.K. and R.H. and F.V.; supervision, R.H. and M.K.; funding acquisition, R.H. and M.K. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was supported by ResearchLab Autonomous Shipping (RAS). This publication is part of the project “Sensor Fusion For Indoor localization Using The Magnetic Field” with project number 18213 of the research program Veni which is (partly) financed by the Dutch Research Council (NWO).

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** The data used to obtain the results in this paper can be found in <https://github.com/fridaviset/EKFMagSLAM>.

**Acknowledgments:** The experiments supporting this work were carried out in collaboration with Matt Hepworth, with ResearchLab Autonomous Shipping (RAS). We would like to thank him for his help in obtaining ground truth position and orientation estimates using a motion capture system, and for collecting IMU measurements onboard the model ship.

**Conflicts of Interest:** The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, or in the decision to publish the results.

## Abbreviations

The following abbreviations are used in this manuscript:

SLAM	Simultaneous localization and mapping
EKF	Extended Kalman filtering
RBPF	Rao-Blackwellized particle filter
GP	Gaussian process
IMU	Inertial measurement unit
ZUPT	Zero-velocity upate
RMSE	Root mean squared error
GNSS	Global navigation satellite system

## Appendix A. Analytical Jacobians

The gradient of the basis functions  $\nabla_p \Phi(p)$  used in (13) (the basis functions  $\Phi(p)$  are defined in (8)) is a  $3 \times (N_m + 3)$  matrix. The first three columns are given by an identity matrix

$$\{\nabla_p \Phi(p)\}_{1:3} = \mathcal{I}_{3 \times 3}, \quad (\text{A1})$$

and the  $j + 3$ rd column is given by the gradient of the  $j$ 'th basis function

$$\{\nabla_p \Phi(p)\}_{j+3} = \nabla_p \phi_j(p) = \begin{bmatrix} \frac{\pi n_j(1)}{L_{u,1} - L_{l,1}} c_1 s_2 s_3 \\ \frac{\pi n_j(2)}{L_{u,2} - L_{l,2}} s_1 c_2 s_3 \\ \frac{\pi n_j(3)}{L_{u,3} - L_{l,3}} s_1 s_2 c_3 \end{bmatrix}, \quad (\text{A2})$$

with  $s_d$  and  $c_d$  defined for the three spatial dimensions  $d = 1, 2, 3$  as

$$s_d = \sin\left(\pi n_j(d) \frac{(p_d - L_{l,d})}{(L_{u,d} - L_{l,d})}\right) \frac{1}{\sqrt{\frac{1}{2}(L_{u,d} - L_{l,d})}}, \quad (\text{A3a})$$

$$c_d = \cos\left(\pi n_j(d) \frac{(p_d - L_{l,d})}{(L_{u,d} - L_{l,d})}\right) \frac{1}{\sqrt{\frac{1}{2}(L_{u,d} - L_{l,d})}}. \quad (\text{A3b})$$

The Jacobians of the basis functions used in (29) is a  $3 \times 3 \times (N_m + 3)$  matrix. The first 3 entries along the third dimensions are all zero-matrices

$$\{\nabla_{pp}\Phi(p)\}_1 = \{\nabla_{pp}\Phi(p)\}_2 = \{\nabla_{pp}\Phi(p)\}_3 = 0_{3 \times 3}, \quad (\text{A4})$$

and the  $(j + 3)$ rd entry along the third dimension is given by

$$\{\nabla_{pp}\Phi(p)\}_j = \nabla_{pp}\phi_j(p) = \begin{bmatrix} -\left(\frac{\pi n_j(1)}{L_{u,1}-L_{l,1}}\right)^2 s_1 s_2 s_3 & \left(\frac{\pi n_j(1)}{L_{u,1}-L_{l,1}}\right)\left(\frac{\pi n_j(2)}{L_{u,2}-L_{l,2}}\right) c_1 c_2 s_3 \\ \left(\frac{\pi n_j(2)}{L_{u,2}-L_{l,2}}\right)\left(\frac{\pi n_j(1)}{L_{u,1}-L_{l,1}}\right) c_1 c_2 s_3 & -\left(\frac{\pi n_j(2)}{L_{u,2}-L_{l,2}}\right)^2 s_1 s_2 s_3 \\ \left(\frac{\pi n_j(3)}{L_{u,3}-L_{l,3}}\right)\left(\frac{\pi n_j(1)}{L_{u,1}-L_{l,1}}\right) c_1 s_2 c_3 & \left(\frac{\pi n_j(3)}{L_{u,3}-L_{l,3}}\right)\left(\frac{\pi n_j(2)}{L_{u,2}-L_{l,2}}\right) s_1 c_2 c_3 \\ \left(\frac{\pi n_j(1)}{L_{u,1}-L_{l,1}}\right)\left(\frac{\pi n_j(3)}{L_{u,3}-L_{l,3}}\right) c_1 s_2 c_3 \\ \dots \left(\frac{\pi n_j(2)}{L_{u,2}-L_{l,2}}\right)\left(\frac{\pi n_j(3)}{L_{u,3}-L_{l,3}}\right) s_1 c_2 c_3 \\ -\left(\frac{\pi n_j(3)}{L_{u,3}-L_{l,3}}\right)^2 s_1 s_2 s_3 \end{bmatrix}. \quad (\text{A5})$$

## References

1. Storms, W.; Shockley, J.; Raquet, J. Magnetic field navigation in an indoor environment. In Proceedings of the Ubiquitous Positioning Indoor Navigation and Location Based Service, Kirkkonummi, Finland, 14–15 October 2010; pp. 1–10.
2. Paull, L.; Saeedi, S.; Seto, M.; Li, H. AUV Navigation and Localization: A Review. *IEEE J. Ocean. Eng.* **2014**, *39*, 131–149. doi:10.1109/JOE.2013.2278891.
3. Dong, L.; Sun, D.; Han, G.; Li, X.; Hu, Q.; Shu, L. Velocity-Free Localization of Autonomous Driverless Vehicles in Underground Intelligent Mines. *IEEE Trans. Veh. Technol.* **2020**, *69*, 9292–9303. doi:10.1109/TVT.2020.2970842.
4. Mekik, C.; Can, O. Multipath effects in RTK GPS and a case study. *J. Aeronaut. Astronaut. Aviat. Ser. A* **2010**, *42*, 231–240.
5. Cadena, C.; Carlone, L.; Carrillo, H.; Latif, Y.; Scaramuzza, D.; Neira, J.; Reid, I.; Leonard, J.J. Past, Present, and Future of Simultaneous Localization and Mapping: Toward the Robust-Perception Age. *IEEE Trans. Robot.* **2016**, *32*, 1309–1332. doi:10.1109/TRO.2016.2624754.
6. Woodman, O.J. *An Introduction to Inertial Navigation*; Technical Report; Computer laboratory University of Cambridge: Cambridge, UK, 2007.
7. Solin, A.; Kok, M.; Wahlström, N.; Schön, T.B.; Särkkä, S. Modeling and Interpolation of the Ambient Magnetic Field by Gaussian Processes. *IEEE Trans. Robot.* **2018**, *34*, 1112–1127. doi:10.1109/TRO.2018.2830326.
8. Kok, M.; Solin, A. Scalable Magnetic Field SLAM in 3D Using Gaussian Process Maps. In Proceedings of the 21st International Conference on Information Fusion, Cambridge, UK, 10–13 July 2018; pp. 1353–1360. doi:10.23919/ICIF.2018.8455789.
9. Yamazaki, K.; Kato, K.; Ono, K.; Saegusa, H.; Tokunaga, K.; Iida, Y.; Yamamoto, S.; Ashiho, K.; Fujiwara, K.; Takahashi, N. Analysis of magnetic disturbance due to buildings. *IEEE Trans. Magn.* **2003**, *39*, 3226–3228. doi:10.1109/TMAG.2003.816729.
10. Haverinen, J.; Kemppainen, A. A global self-localization technique utilizing local anomalies of the ambient magnetic field. In Proceedings of the IEEE International Conference on Robotics and Automation, Kobe, Japan, 12–17 May 2009; pp. 3142–3147. doi:10.1109/ROBOT.2009.5152885.
11. Vallivaara, I.; Haverinen, J.; Kemppainen, A.; Röning, J. Simultaneous localization and mapping using ambient magnetic field. In Proceedings of the IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI), Karlsruhe, Germany, 23–25 September 2010; pp. 14–19. doi:10.1109/MFI.2010.5604465.
12. Kemppainen, A.; Haverinen, J.; Vallivaara, I.; Röning, J. Near-optimal SLAM exploration in Gaussian processes. In Proceedings of the IEEE Conference on Multisensor Fusion and Integration, Salt Lake City, UT, USA, 5–7 September 2010; pp. 7–13. doi:10.1109/MFI.2010.5604467.
13. Vallivaara, I.; Haverinen, J.; Kemppainen, A.; Röning, J. Magnetic field-based SLAM method for solving the localization problem in mobile robot floor-cleaning task. In Proceedings of the IEEE 15th International Conference on Advanced Robotics: New Boundaries for Robotics (ICAR 2011), Tallinn, Estonia, 20–23 June 2011; pp. 198–203. doi:10.1109/ICAR.2011.6088632.
14. Chung, J.; Donahoe, M.; Schmandt, C.; Kim, I.J.; Razavi, P.; Wiseman, M. Indoor location sensing using geo-magnetism. In Proceedings of the 9th International Conference on Mobile Systems, Applications, and Services, Bethesda, MA, USA, 28 June–1 July 2011; pp. 141–154. doi:10.1145/1999995.2000010.
15. Le Grand, E.; Thrun, S. 3-Axis magnetic field mapping and fusion for indoor localization. In Proceedings of the IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI), Hamburg, Germany, 13–15 September 2012; pp. 358–364. doi:10.1109/MFI.2012.6343024.

16. Kim, S.E.; Kim, Y.; Yoon, J.; Kim, E.S. Indoor positioning system using geomagnetic anomalies for smartphones. In Proceedings of the International Conference on Indoor Positioning and Indoor Navigation (IPIN), Sydney, NSW, Australia, 13–15 November 2012; pp. 1–5.
17. Frassl, M.; Angermann, M.; Lichtenstern, M.; Robertson, P.; Julian, B.J.; Doniec, M. Magnetic maps of indoor environments for precise localization of legged and non-legged locomotion. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, Tokyo, Japan, 3–7 November 2013; pp. 913–920. doi:10.1109/IROS.2013.6696459.
18. Robertson, P.; Frassl, M.; Angermann, M.; Doniec, M.; Julian, B.; Puyol, M.; Khider, M.; Lichtenstern, M.; Bruno, L. Simultaneous Localization and Mapping for Pedestrians using Distortions of the Local Magnetic Field Intensity in Large Indoor Environments. In Proceedings of the International Conference on Indoor Positioning and Indoor Navigation (IPIN 2013), Montbeliard, France, 28–31 October 2013. doi:10.1109/IPIN.2013.6817910.
19. Jung, J.; Lee, S.M.; Myung, H. Indoor Mobile Robot Localization and Mapping Based on Ambient Magnetic Fields and Aiding Radio Sources. *IEEE Trans. Instrum. Meas.* **2015**, *64*, 1922–1934. doi:10.1109/TIM.2014.2366273.
20. Solin, A.; Särkkä, S.; Kannala, J.; Rahtu, E. Terrain navigation in the magnetic landscape: Particle filtering for indoor positioning. In Proceedings of the European Navigation Conference (ENC), Helsinki, Finland, 30 May–2 June 2016; pp. 1–9. doi:10.1109/EURONAV.2016.7530559.
21. Kim, H.S.; Seo, W.; Baek, K. Indoor Positioning System Using Magnetic Field Map Navigation and an Encoder System. *Sensors* **2017**, *17*, 651. doi:10.3390/s17030651.
22. Viset, F.; Gravidahl, J.; Kok, M. Magnetic field norm SLAM using Gaussian process regression in foot-mounted sensors. In Proceedings of the European Control Conference, Delft, The Netherlands, 29 June–2 July 2021; pp. 392–398. doi:10.23919/ECC54610.2021.9655230.
23. Zheng, H.; Wang, H.; Wu, L.; Chai, H.; Wang, Y. Simulation research on gravity-geomagnetism combined aided underwater navigation. *J. Navig.* **2013**, *66*, 83–98. doi:10.1017/S0373463312000343.
24. Nygren, I. Robust and efficient terrain navigation of underwater vehicles. In Proceedings of the IEEE/ION Position, Location and Navigation Symposium, Monterey, CA, USA, 5–8 May 2008; pp. 923–932. doi:10.1109/PLANS.2008.4570034.
25. Lager, M.; Topp, E.A.; Malec, J. Underwater Terrain Navigation Using Standard Sea Charts and Magnetic Field Maps. In Proceedings of the IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI), Daegu, Korea, 16–18 November 2017; pp. 78–84. doi:10.1109/MFI.2017.8170410.
26. Tal, A.; Klein, I.; Katz, R. Inertial Navigation System/Doppler Velocity Log (INS/DVL) Fusion with Partial DVL Measurements. *Sensors* **2017**, *17*, 415. doi:10.3390/s17020415.
27. Tyrén, C. Magnetic terrain navigation. In Proceedings of the 5th International Symposium on Unmanned Untethered Submersible Technology (UUST 1987), Durham, NH, USA, June 1987; pp. 245–256. doi:10.1109/UUST.1987.1158556.
28. Wu, Z.; Hu, X.; Wu, M.; mu, H.; Cao, J.; Zhang, K.; Tuo, Z. An experimental evaluation of autonomous underwater vehicle localization on geomagnetic map. *Appl. Phys. Lett.* **2013**, *103*, 104102. doi:10.1063/1.4819832.
29. Wu, Y.; Shi, W. On Calibration of Three-Axis Magnetometer. *IEEE Sens. J.* **2015**, *15*, 6424–6431. doi:10.1109/JSEN.2015.2459767.
30. Huang, Y.; Hao, Y. Method of separating dipole magnetic anomaly from geomagnetic field and application in underwater vehicle localization. In Proceedings of the IEEE International Conference on Information and Automation, Harbin, China, 20–23 June 2010; pp. 1357–1362. doi:10.1109/ICINFA.2010.5512104.
31. Teixeira, F.C.; Quintas, J.; Pascoal, A. Robust methods of magnetic navigation of marine robotic vehicles. *IFAC-PapersOnLine* **2017**, *50*, 3470–3475. doi:10.1016/j.ifacol.2017.08.851.
32. Mu, H.; Wu, M.; Hu, X.; Hongxu, M. Geomagnetic surface navigation using adaptive EKF. In Proceedings of the Second IEEE Conference on Industrial Electronics and Applications, Harbin, China, 23–25 May 2007; pp. 2821–2825. doi:10.1109/ICIEA.2007.4318926.
33. Canciani, A.; Raquet, J. Airborne Magnetic Anomaly Navigation. *IEEE Trans. Aerosp. Electron. Syst.* **2017**, *53*, 67–80. doi:10.1109/TAES.2017.2649238.
34. Quintas, J.; Cruz, J.; Pascoal, A.; Teixeira, F.C. A Comparison of Nonlinear Filters for Underwater Geomagnetic Navigation. In Proceedings of the IEEE/OES Autonomous Underwater Vehicles Symposium (AUV), St. Johns, NL, Canada, 30 September–2 October 2020; pp. 1–6. doi:10.1109/AUV50043.2020.9267915.
35. Liu, S.; Mingas, G.; Bouganis, C.S. Parallel resampling for particle filters on FPGAs. In Proceedings of the International Conference on Field-Programmable Technology (FPT), Shanghai, China, 10–12 December 2014; pp. 191–198. doi:10.1109/FPT.2014.7082775.
36. Varsi, A.; Maskell, S.; Spirakis, P.G. An  $O(\log 2N)$  Fully-Balanced Resampling Algorithm for Particle Filters on Distributed Memory Architectures. *Algorithms* **2021**, *14*, 342. doi:10.3390/a14120342.
37. Solin, A.; Särkkä, S. Hilbert Space Methods for Reduced-Rank Gaussian Process Regression. *Stat. Comput.* **2020**, *30*, 419–446. doi:10.1007/s11222-019-09886-w.
38. Kok, M.; Schön, T.B. Magnetometer Calibration Using Inertial Sensors. *IEEE Sens. J.* **2016**, *16*, 5679–5689. doi:10.1109/JSEN.2016.2569160.
39. Wahlström, N.; Gustafsson, F. Magnetometer Modeling and Validation for Tracking Metallic Targets. *IEEE Trans. Signal Process.* **2014**, *62*, 545–556.
40. Skog, I. OpenShoe Matlab Framework. In *OpenShoe—Foot-Mounted INS for Every Foot*; KTH Royal Institute of Technology, Stockholm, Sweden and Indian Institute of Science: Bangalore, India, 2011. Available online: [http://www.openshoe.org/?page\\_id=362](http://www.openshoe.org/?page_id=362) (accessed on 18 November 2021).

41. Kok, M.; Hol, J.D.; Schön, T.B. Using Inertial Sensors for Position and Orientation Estimation. *Found. Trends Signal Process.* **2017**, *11*, 1–153. doi:10.1561/20000000094.
42. Wahlström, N.; Kok, M.; Schön, T.B.; Gustafsson, F. Modeling magnetic fields using Gaussian processes. In Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, Vancouver, BC, Canada, 26–31 May 2013; pp. 3522–3526. doi:10.1109/ICASSP.2013.6638313.
43. Chen, Z. Bayesian Filtering: From Kalman Filters to Particle Filters, and Beyond. *Statistics* **2003**, *182*, 1–69. doi:10.1080/02331880309257.
44. Gustafsson, F. *Statistical Sensor Fusion*; Studentlitteratur AB: Lund, Sweden, 2013.
45. Arulampalam, M.; Maskell, S.; Gordon, N.; Clapp, T. A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking. *IEEE Trans. Signal Process.* **2002**, *50*, 174–188. doi:10.1109/78.978374.
46. Foxlin, E. Pedestrian tracking with shoe-mounted inertial sensors. *IEEE Comput. Graph. Appl.* **2005**, *25*, 38–46. doi:10.1109/MCG.2005.140.
47. Nilsson, J.; Skog, I.; Händel, P.; Hari, K.V.S. Foot-mounted INS for everybody—An open-source embedded implementation. In Proceedings of the IEEE/ION Position, Location and Navigation Symposium, Myrtle Beach, SC, USA, 23–26 April 2012; pp. 140–145. doi:10.1109/PLANS.2012.6236875.
48. Skoglund, M.A.; Hendeby, G.; Axehill, D. Extended Kalman filter modifications based on an optimization view point. In Proceedings of the 18th International Conference on Information Fusion (Fusion), Washington, DC, USA, 6–9 July 2015; pp. 1856–1861.