# Distributed multi-agent magnetic field norm SLAM with Gaussian processes

Viset, Frida; Helmons, Rudy; Kok, Manon

# Distributed multi-agent magnetic field norm SLAM with Gaussian processes

Frida Viset[*], Rudy Helmons[†] and Manon Kok[*]

[*]Delft Center for Systems and Control,[†]Maritime and Transport Technology, and Institute for Geoscience and Petroleum

[* †]Delft University of Technology, and [†]Norwegian University of Science and Technology

Email: {F.M.Viset, R.L.J.Helmons, M.Kok-1}@tudelft.nl

*Abstract*—**Accurately estimating the positions of multi-agent systems in indoor environments is challenging due to the lack of Global Navigation Satelite System (GNSS) signals. Noisy measurements of position and orientation can cause the integrated position estimate to drift without bound. Previous research has proposed using magnetic field simultaneous localization and mapping (SLAM) to compensate for position drift in a single agent. Here, we propose two novel algorithms that allow multiple agents to apply magnetic field SLAM using their own and other agents' measurements.**

**Our first algorithm is a centralized approach that uses all measurements collected by all agents in a single extended Kalman filter. This algorithm simultaneously estimates the agents' position and orientation and the magnetic field norm in a central unit that can communicate with all agents at all times. In cases where a central unit is not available, and there are communication drop-outs between agents, our second algorithm is a distributed approach that can be employed.**

**We tested both algorithms by estimating the position of magnetometers carried by three people in an optical motion capture lab with simulated odometry and simulated communication dropouts between agents. We show that both algorithms are able to compensate for drift in a case where single-agent SLAM is not. We also discuss the conditions for the estimate from our distributed algorithm to converge to the estimate from the centralized algorithm, both theoretically and experimentally.**

**Our experiments show that, for a communication drop-out rate of $80\%$, our proposed distributed algorithm, on average, provides a more accurate position estimate than single-agent SLAM. Finally, we demonstrate the drift-compensating abilities of our centralized algorithm on a real-life pedestrian localization problem with multiple agents moving inside a building.**

*Index Terms*—**Multi-agent, SLAM, Gaussian processes, Distributed Kalman filters.**

## I. INTRODUCTION

A wide range of research is being performed on multi-agent motion control and path planning algorithms [1]. For most motion control algorithms, it is crucial for each agent to know its own position [2], [3]. Collaborative pedestrian navigation can be useful for example for rescue missions or law enforcement applications [4]. Indoors, Global Navigation Satellite System (GNSS) signal availability is limited and prone to

errors [5]. Current indoor navigation systems therefore often rely on integrating measurements of the change in position and orientation. For autonomous navigation in GNSS-denied environments where there are no previously deployed beacons or other structure supporting navigation, measurements of the change in position and orientation are often available from for example inertial sensors, wheel encoders or visual-inertial odometry [6]. Integrating measurements of change in position and orientation (odometry) gives accumulated position estimation errors (drift) that can increase without an upper bound [7].

To compensate for odometry drift, multi-agent simultaneous localization and mapping (SLAM) algorithms for navigation in GNSS-denied environments based on visual information have been widely studied [8]. Visual SLAM can in some applications be infeasible or prone to error due to privacy concerns, varying light conditions, or lack of distinguishable features or landmarks [9].

For several single-agent navigation tasks, magnetic field SLAM has been proposed and demonstrated to compensate for drift in the position estimate [10]–[17]. The magnetic field indoors is affected by structural metallic elements [18]. In Figure 1, an example of the magnetic field norm variations that can be found indoors is displayed. The indoor magnetic field typically has significant spatial variations and stays constant over time [19], [20]. To simultaneously create and use a map of the magnetic field, most approaches use a nonlinear stochastic interpolation scheme to learn the magnetic field online based on measurements. A stochastic interpolation scheme that also gives an uncertainty measure on the predictions in every location of the map is Gaussian process regression. Several of the previous works into magnetic field SLAM use reduced-rank Gaussian process regression approximated with Hilbert space basis functions so the computational complexity does not scale with the number of measurements [10], [11], [14], [21].

The contribution of this paper is twofold. The first contribution is an algorithm that uses all information measured by multiple agents to perform magnetic field norm SLAM online with an extended Kalman filter (EKF). This EKF is obtained by augmenting the state-space of the EKF for magnetic field SLAM in [21] to contain the poses of multiple agents as opposed to just a single agent. We denote this as the centralized algorithm, as it is an algorithm that can be executed in a centralized station that receives all measurements
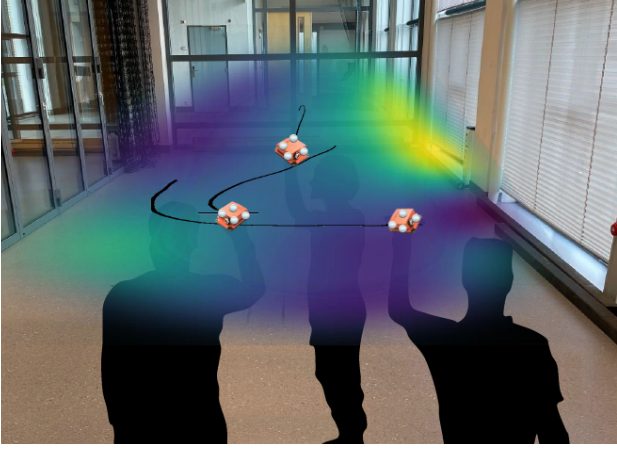
Fig. 1: Estimated magnetic field map and trajectories of multiple agents based on measurements from magnetometers carried by three people. The position of the magnetometer was recorded in an optical motion capture lab. The color of the map reflects the magnitude of the magnetic field norm, while the opacity of the overlaid map is inversely proportional to the marginal variance of the estimate.

made by all agents. Multi-agent systems do not always have access to a centralized control unit. Our second contribution is therefore a distributed version of the algorithm, where each agent uses information shared in communication between the agents to collaboratively approximate the output of the centralized algorithm. To implement the centralized EKF as a decentralized EKF, we use an approach closely related to the decentralized Kalman filter described in [22]. To the best of the author's knowledge, this is the first proposed algorithm for distributed multi-agent magnetic field SLAM with Gaussian process regression.

## II. CONNECTIONS TO PREVIOUS WORK

Previous work has applied average consensus to achieve distributed reduced-rank Gaussian process regression using measurements from multiple agents [23], [24]. Recursive stochastic least squares correspond to applying repeated Kalman filter measurement updates [25]. Magnetic field SLAM with an extended Kalman filter uses both a dynamic update and a measurement update at each timestep to jointly estimate the magnetic field map and the pose of a single agent [21]. Previous work has also demonstrated that Kalman filters with both measurement updates and dynamic updates can be implemented for multiple agents distributively with embedded consensus filters [22]. The distributed implementation in [22] is implemented by solving two consensus problems at each time step, one in the dynamic update and one in the measurement update. We also implement the distributed EKF by solving these two consensus problems at each time step. For both our distributed EKF and for the distributed Kalman filter in [22], even if each average consensus problem has not converged, the intermittent result is an approximation of the centralized solution [26].

Unlike previous work into extended Kalman filtering for magnetic field SLAM, we execute the measurement update on the information form. This allows for the measurement update to be implemented distributively by executing the average consensus algorithm at each timestep. Performing the measurement update for magnetic field SLAM on information form is closely related to the execution of the measurement updates on information form for magnetic field mapping proposed by [24]. The main difference between our work and the estimation algorithm presented in [24] is that we jointly and distributively estimate the pose of the agents and the map, while [24] only estimates the map. The main difference between our work and [21] and [27] is that we perform magnetic field SLAM for several agents instead of just one and that we propose a distributed algorithm for doing so. An additional difference between our work and the work presented in [21] is that we for simplicity consider only the magnetic field norm instead of the three-component magnetic field.

## III. MODEL

We assume that each individual agent has access to noisy odometry measurements, according to a model we describe in Section III-A. We also assume that each agent carries a magnetometer capable of measuring the magnetic field norm. In Section III-B we give the measurement model for the magnetometer and the model of the magnetic field norm that we use to apply Gaussian process regression to learn the magnetic field map.

### A. Dynamic model

We estimate the position of a set of $m$ agents indexed as $i = 1, \ldots, m$. The position and orientation of each agent at each timestep $t$ are denoted by the vector $p_{i,t}$ and the unit quaternion $q_{i,t}$ respectively. The quaternion is defined as the orientation from the world frame to the body frame. The body frame has its origin in the IMU's center of mass, and its axes are aligned with the accelerometer sensor axes. The world frame is defined as the stationary inertial frame that shares its origin with the body frame at time $t = 0$, where the gravity field is aligned with the negative z-axis, and the initial yaw-angle between the body and world-frame at $t = 0$ is zero. The position is given in the world frame.

We assume that each agent has access to noisy measurements $\Delta p_{i,t}$ of the change in their position and $\Delta q_{i,t}$ of the change in their orientation from sensors mounted in the body frame. The noisy measurements are defined such that

$$p_{i,t+1} = p_{i,t} + R(q_{i,t})(\Delta p_{i,t} + e_{i,\mathrm{p},t}), \tag{1a}$$

$$q_{i,t+1} = q_{i,t} \odot \exp_\mathrm{q}(\Delta q_{i,t}) \odot \exp_\mathrm{q}(e_{i,\mathrm{q},t}), \tag{1b}$$

$$[e_{i,\mathrm{p},t}^\top, e_{i,\mathrm{q},t}^\top]^\top \sim \mathcal{N}(0, \Sigma), \tag{1c}$$

where $e_{i,\mathrm{p},t}$ is a measurement noise of the change in position, $e_{i,\mathrm{q},t}$ is a measurement noise of the change in orientation, and where $\Sigma$ is a known noise covariance, $\odot$ is the quaternion product, and $\exp_\mathrm{q}$ is the operator that maps an axis-angle orientation deviation to a quaternion, defined as in the odometry

model in [14], and where $R(\cdot)$ is an operator transforming a unit quaternion to a rotation, defined as in the odometry model in [27]. Note that we assume the odometry covariance is the same for all agents.

### B. Measurement model

We assume that each agent $i$ has access to a continuous stream of measurements from the magnetic field norm in their current position $p_{i,t}$, according to

$$y_{i,t} = f(p_{i,t}) + e_{i,t}, \qquad e_{i,t} \sim \mathcal{N}(0, \sigma_{\mathrm{y}}^2), \qquad (2)$$

where $y_{i,t}$ is the measurement from agent $i$ at time $t$, $f : \mathbb{R}^3 \to \mathbb{R}$ is a function that maps the position to the magnetic field norm, and $e_{i,t}$ is the measurement noise with a covariance $\sigma_{\mathrm{y}}^2$. We model the function $f$ as a stationary Gaussian process according to

$$f \sim \mathcal{GP}(0, \kappa_{\mathrm{SE}}(\cdot, \cdot)), \qquad (3)$$

with a squared exponential kernel

$$\kappa_{\mathrm{SE}}(x, x') = \sigma_{\mathrm{SE}}^2 \exp\left(-\frac{\|x - x'\|_2}{2 l_{\mathrm{SE}}^2}\right), \qquad (4)$$

where $\sigma_{\mathrm{SE}}^2$ and $l_{\mathrm{SE}}$ are hyperparameters denoting the variance and lengthscale of the magnetic field norm nonlinearities, respectively [27]. We use the same basis functions as [21] and [27] to approximate the Gaussian process regression. The basis functions are defined in Appendix A.

### C. Communication graph

We assume that the agents have the possibility to send and receive $N_c$ messages to all other agents two times at each timestep $t$, once for the dynamic update and once for the measurement update.

We model the communication graph at time $t$ and communication step $t_c$ as an undirected graph $\mathcal{G}(t, t_c) = (\mathcal{E}(t, t_c), \mathcal{V})$ where $\mathcal{E}(t, t_c) \subset \{\{i, j\} | i, j \in \mathcal{V}\}$ denote the set of active communication edges at time $t$ between the set $\mathcal{V} = \{1, \cdots, m\}$ of all agents. We assume the probability for two agents to be able to communicate at any timestep $t$ at communication step $t_c$ to be $1 - \alpha$, where $\alpha$ is the probability of communication failure. We will refer to $\alpha$ as the communication failure rate or the dropout rate in the remainder of this paper. We denote the communication step at each timestep by the index $t_c$, where $t_c = 1, \ldots, 2N_c$.

## IV. CENTRALIZED EKF FOR MULTI-AGENT MAGNETIC FIELD SLAM

Following the approach of [21], we parameterize our system in terms of an error state $\xi_t$ linearised about the prior beliefs of the position of agents $i = 1, \ldots, m$ denoted $\tilde{p}_{i,t|t-1}$, the prior beliefs of the orientation of agent $i = 1, \ldots, m$ denoted $\tilde{q}_{i,t|t-1}$ and the prior belief of the map denoted $\tilde{w}_{i,t|t-1}$. The error state $\xi_t$ is defined as

$$\xi_t = [\delta_{1,t}^\top \quad \eta_{1,t}^\top \quad \cdots \quad \delta_{m,t}^\top \quad \eta_{m,t}^\top \quad \nu_t^\top]^\top, \qquad (5)$$

where $\delta_{i,t} = p_{i,t} - \tilde{p}_{i,t|t-1}$ denotes the position estimation error, $\nu_{t,i} = w - \tilde{w}_{i,t|t-1}$ denotes the magnetic field state

estimation error, and $\eta_{i,t}$ denotes the orientation estimation error parameterized as an axis-angle deviation according to

$$q_{i,t} = \exp_{\mathrm{q}}(\eta_{i,t}) \odot \tilde{q}_{i,t|t-1}. \qquad (6)$$

For simplicity, we assume that the initial position and orientation of all agents are known. The initial error state is then distributed as $\xi_0 \sim \mathcal{N}(0, P_{0|0})$, where $P_{0|0}$ is given by

$$P_{0|0} = \begin{bmatrix} 0 & \cdots & 0 & 0 \\ \vdots & \ddots & \vdots & \vdots \\ 0 & \cdots & 0 & 0 \\ 0 & \cdots & 0 & \Lambda, \end{bmatrix}. \qquad (7)$$

with $\Lambda$ defined in (28).

### A. Dynamic update

The posterior linearisation point is propagated to a prior linearisation point by applying the dynamic model in (1a)-(1c) through the update

$$\tilde{q}_{i,t+1|t} = \tilde{q}_{i,t|t} \odot \exp_{\mathrm{q}}(\Delta q_{i,t}), \qquad i = 1, \ldots, m \qquad (8a)$$

$$\tilde{p}_{i,t+1|t} = \tilde{p}_{i,t|t} + R(\tilde{q}_{i,t|t}) \Delta p_{i,t}, \qquad i = 1, \ldots, m \qquad (8b)$$

$$\tilde{w}_{t+1|t} = \tilde{w}_{t|t}. \qquad (8c)$$

The centralized dynamic update is defined as

$$P_{t+1|t} = F_t P_{t|t} F_t^\top + Q, \qquad (9)$$

where $Q$ is given by

$$Q = \begin{bmatrix} \Sigma & \cdots & 0 & 0 \\ \vdots & \ddots & \vdots & \vdots \\ 0 & \cdots & \Sigma & 0 \\ 0 & \cdots & 0 & 0 \end{bmatrix} \qquad (10)$$

and $F_t$ is defined as

$$F_t = \begin{bmatrix} F_{1,t} & \cdots & 0 & 0 \\ \vdots & \ddots & \vdots & \vdots \\ 0 & \cdots & F_{m,t} & 0 \\ 0 & \cdots & 0 & I \end{bmatrix}, \qquad (11)$$

where the matrix $F_{j,t}$ is given by

$$F_{j,t} = \begin{bmatrix} I & R(\tilde{q}_{j,t|t})[\Delta p_{j,t} \times] \\ 0 & I \end{bmatrix}, \qquad (12)$$

and where $[\Delta p_{i,t} \times]$ is defined as the skew-symmetric matrix such that $[\Delta p_{i,t} \times] u = \Delta p_{i,t} \times u$ gives the cross-product between $\Delta p_{i,t}$ and a vector $u \in \mathbb{R}^3$.

### B. Measurement update

The measurement update is performed by linearising the measurement model in (2) about the prior linearisation point with respect to the error state $\xi_t$. We let the information vector $\iota_{t|t-1}$ and information matrix $\mathcal{I}_{t|t-1}$ denote the information form of the state estimate $\hat{\xi}_{t|t-1}$ and the corresponding covariance $P_{t|t-1}^{-1}$, according to

$$\iota_{t|t-1} = P_{t|t-1}^{-1} \hat{\xi}_{t|t-1} = 0, \qquad (13)$$

$$\mathcal{I}_{t|t-1} = P_{t|t-1}^{-1}. \qquad (14)$$

**Algorithm 1** Centralized EKF for multi-agent magnetic field SLAM

Input: $\left\{\{\Delta p_{i,t}, \Delta q_{i,t}, y_{i,t}\}_{t=1}^{N}\right\}_{i=1}^{m}$

Output: $\left\{\{\tilde{p}_{i,t|t}, \tilde{q}_{i,t|t}, \tilde{w}_{t|t}\}_{t=1}^{N}\right\}_{i=1}^{m}$

    *Initialization*: $\tilde{p}_{i,0|0} = 0_{3\times1}$, $\tilde{q}_{i,0|0} = q_{0,i}$, $\tilde{w}_{0|0} = 0_{M\times1}$, (7)

1: **for** $t = 1$ to $N$ **do**
2:     Dynamic update according to (8a), (8b), (8c) and (9).
3:     Measurement update according to (14), (15a), (15b) and (17). Relinearization according to (18a), (18b) and (18c).
4: **end for**

---

The Kalman filter measurement update can then be expressed as an update of the information matrix and information vector as

$$\mathcal{I}_{t|t} = \mathcal{I}_{t|t-1} + \sum_{i=1}^{m} \frac{1}{\sigma_{\mathrm{y}}^2} H_{i,t} H_{i,t}^{\top}, \tag{15a}$$

$$\iota_{t|t} = \iota_{t|t-1} + \sum_{i=1}^{m} \frac{1}{\sigma_{\mathrm{y}}^2} H_{i,t}(y_{i,t} - \Phi(\tilde{p}_{i,t-1})^{\top} \tilde{w}_{t|t-1}), \tag{15b}$$

with

$$\begin{aligned} H_{i,t} = [&0_{1\times6(i-1)}, (\nabla\Phi(\tilde{p}_{i,t|t-1})\tilde{w}_{t|t-1})^{\top}, \\ &0_{1\times(3+6(m-i))}, (\Phi(\tilde{p}_{i,t|t-1}))^{\top}]^{\top}. \end{aligned} \tag{16}$$

The posterior error state estimate and covariance are given by

$$\hat{\xi}_{t|t} = \mathcal{I}_{t|t}^{-1} \iota_{t|t}, \qquad P_{t|t} = \mathcal{I}_{t|t}^{-1}. \tag{17}$$

The posterior linearisation point can then be calculated by propagating the estimated error state to the prior linearisation point according to

$$\tilde{p}_{i,t|t} = \tilde{p}_{i,t|t-1} + \hat{\delta}_{i,t|t}, \qquad i = 1, \dots, m, \tag{18a}$$

$$\tilde{q}_{i,t|t} = \exp_{\mathrm{q}}(\hat{\eta}_{i,t|t}) \odot \tilde{q}_{i,t|t-1}, \qquad i = 1, \dots, m, \tag{18b}$$

$$\tilde{w}_{t|t} = \tilde{w}_{t|t-1} + \hat{\nu}_{t|t}. \tag{18c}$$

Recursively applying the dynamic update and measurement update results in the centralized EKF for multi-agent magnetic field SLAM, as described in Algorithm 1.

## V. DISTRIBUTED MULTI-AGENT EKF FOR MAGNETIC FIELD SLAM

We denote agent $i$'s approximation of a centralized term by including a superscript $(i)$ on the approximated term. The initial posterior linearisation points are known and given as $\tilde{p}_{i,0|0}^{(i)} = p_{i,0}$, $\tilde{q}_{i,0|0}^{(i)} = q_{i,0}$ and $\tilde{w}_{0|0}^{(i)} = 0$. As in the centralized filter, we assume that the initial error centralized error state $\hat{\xi}_{t|t}^{(i)} = 0$ and the initial centralized covariance $P_{0|0}^{(i)} = P_{0|0}$ are both known.

## A. Dynamic update

In the case where each agent only has access to their own measurements, the posterior linearisation point of each agent can be propagated to a prior linearisation point through the dynamic model in the same way as for the centralized EKF, using (9). The matrix $F_t$ cannot be computed directly by any agent as each term $F_{j,t}$ contains the odometry measurement $\Delta p_{j,t}$ which is only available to agent $j$. The matrix $F_t$ can however be approximated by the network as a whole through average consensus, if each agent initializes their belief about the matrix $F_t^{(i)}$ according to

$$F_t^{(i)} = \begin{bmatrix} F_{1,t}^{(i)} & \cdots & 0 & 0 \\ \vdots & \ddots & \vdots & \vdots \\ 0 & \cdots & F_{m,t}^{(i)} & 0 \\ 0 & \cdots & 0 & I \end{bmatrix}, \tag{19}$$

where the term $F_{j,t}^{(i)}$ is defined according to

$$F_{j,t}^{(i)} = \begin{cases} mF_{j,t} - (m-1)I, & j = i \\ I, & j \neq i \end{cases} \tag{20}$$

The average of all the terms $\{F_t^{(i)}\}_{i=1}^{m}$ is $F_t$, so applying average consensus according to

$$F_t^{(i)} \leftarrow \sum_{j=1}^{m} W_{i,j}(t, t_c) F_t^{(j)}, \tag{21}$$

where the weights $W_{i,j}(t, t_c)$ are defined as in [26] as

$$W_{i,j}(t, t_c) = \begin{cases} \frac{1}{m}, & i, j \in \mathcal{E}(t, t_c) \\ 1 - \frac{d_i(t, t_c)}{m}, & i = j \\ 0, & \text{otherwise} \end{cases}, \tag{22}$$

and where $d_i(t, t_c)$ are the number of edges to node $i$ in the communication graph $\mathcal{E}(t, t_c)$ at timestep $t$, for $t_c = 1, \dots, N_c$ causes $F_t^{(i)}$ to converge to $F_t$ as $N_c \to \infty$ [26]. As we only apply a finite amount of average consensus steps $N_c$, we use $F_t^{(i)}$ at time $N_c$ as an approximation in the dynamic update of the covariance.

## B. Measurement update

The measurement update can be carried out in a distributed manner by first letting each agent update its belief about the information vector according to

$$\mathcal{I}_{t|t}^{(i)} = \mathcal{I}_{t|t-1}^{(i)} + m\frac{1}{\sigma_{\mathrm{y}}^2} H_{i,t} H_{i,t}^{\top}, \tag{23a}$$

$$\iota_{t|t}^{(i)} = \iota_{t|t-1}^{(i)} + m\frac{1}{\sigma_{\mathrm{y}}^2} H_{i,t}(y_{i,t} - \Phi(\tilde{p}_{i,t-1}^{(i)})^{\top} \tilde{w}_{t|t-1}^{(i)}), \tag{23b}$$

and then carry out average consensus across the network on the resulting information matrix and information vector, according to

$$\iota_{t|t}^{(i)} \leftarrow \sum_{j=1}^{m} W_{i,j}(t, t_c) \iota_{t|t}^{(j)} \tag{24a}$$

$$\mathcal{I}_{t|t}^{(i)} \leftarrow \sum_{j=1}^{m} W_{i,j}(t, t_c) \mathcal{I}_{t|t}^{(j)} \tag{24b}$$

**Algorithm 2** Distributed EKF for multi-agent magnetic field SLAM for agent $i$

---

**Input:** $\left\{\{\Delta p_{i,t}, \Delta q_{i,t}, y_{i,t}\}_{t=1}^N\right\}_{i=1}^m$

**Output:** $\left\{\left\{\tilde{p}_{i,t|t}^{(i)}, \tilde{q}_{i,t|t}^{(i)}, \tilde{w}_{t|t}^{(i)}\right\}_{t=1}^N\right\}_{i=1}^m$

    *Initialization*: $\tilde{p}_{i,0|0}^{(i)} = 0_{3\times 1}$, $\tilde{q}_{i,0|0}^{(i)} = q_0$, $\tilde{w}_{0|0}^{(i)} = 0_{M\times 1}$, (7)

1: **for** $t = 1$ to $N$ **do**
2:    **Dynamic update:** Perform average consensus according to (19), (20) and (21). Then, propagate own belief of own state according to (8a), (8b), (8c) and (9), using output terms from average consensus.
3:    **Measurement update:** according to (23a), (23b). Average consensus according to (24a) and (24b). Relinearization according to (18a), (18b) and (18c).
4: **end for**

---

The result will then converge to the information matrix and information vector obtained by (15a)- (15b) as the number of communication steps goes to infinity. We use the output from the average consensus procedure as an approximation to the centralized information matrix in each agent. Each agent can therefore update their own linearization point locally by using the same update as the centralized EKF in (18a)-(18c). When there is no communication failure, the approximation will be exactly equal to the centralized solution even with $N_c = 1$. Recursively applying the dynamic update and the measurement update gives the Distributed EKF for multi-agent magnetic field SLAM described in Algorithm 2. If all $m$ agents are running Algorithm 2, the multi-agent system will collaboratively approximate the centralized estimate of Algorithm 1.

## VI. RESULTS

### A. Comparison of Algorithm 1 to Single-Agent SLAM

We test the ability of our algorithm to simultaneously estimate the locations of three handheld devices containing magnetometers, by testing on data collected by three test subjects in a motion capture lab. The experimental setup is illustrated in Figure 1. Each test subject held an Xsens MTi-100 IMU, which was used to collect magnetic field measurements. The ground truth position and orientation of the IMU were recorded with an optical motion capture system. The test subjects moved sequentially in the test area to ensure marker visibility for the optical motion capture system, but we test our algorithm on the three measured trajectories as if collected simultaneously.

The position measurements $\Delta p_{i,t}$ were simulated by first computing the difference of the recorded ground truth positions from each timestep to the next, and then adding noises of $e_{1,\mathrm{p},t} = [0.000, 0.001\ 0]$, $e_{2,\mathrm{p},t} = [-0.001, -0.0005, 0]$ and $e_{3,\mathrm{p},t} = [0.001, -0.0005, 0]$. These noise values were selected such that the position estimates of the agents would drift in different directions over a short timescale, making the

dead-reckoning position estimates to other agents particularly poor.

The differential orientation measurements $\Delta q_{i,t}$ were simulated by computing the difference in orientation from one timestep to the next, and then adding a simulated noise sampled from a normal distribution with standard deviation $\sigma_q = 1.0e - 5$. We then applied Algorithm 1 to the magnetic field measurements and the simulated odometry. The Gaussian process hyperparameters were set to $\sigma_{\mathrm{SE}} = 0.074$, $\sigma_y = 0.0042$, $l_{\mathrm{SE}} = 0.86$m. The hyperparameters were selected based on an optimization of the Gaussian process likelihood, using the recorded position for all the agents as the input locations and the magnetic field norm as the output. The parameter $\sigma_p$ used in the estimation was set to 0.022, which is two times as high as the maximum norm of the simulated noise, to make sure that the Kalman filter did not put too much trust in the odometry. To approximate the Gaussian process, 100 basis functions were used in a domain $\Omega$ defined as the smallest cube that was no closer than 3 meters to the closest recorded position. This is a sufficient amount of basis functions, as the approximation error between the reduced rank and the full GP in ten test points selected in random locations sampled from a uniform distribution inside the domain given all the collected measurements is lower than one measurement noise standard deviation $\sigma_y$.

The estimated trajectories using Algorithm 1 are displayed together with the learned magnetic field in Fig. 2. The results in Figure 3 show that the EKF for a single agent improves on the position estimate for all three agents. The end-point estimation error for Single agent SLAM is 87%, 65%, and 81% of the odometry error, respectively. Over time, the position estimates for Single agent SLAM are typically bounded [21], but for this example on this timescale, each agent does not have time to collect sufficient information about the magnetic field to compensate fully for the odometry drift. Even in this challenging case for magnetic field SLAM, multi-agent SLAM is able to compensate for the odometry drift. The end-point estimation errors of the position estimates from Algorithm 1 in Figure 3 are 37%, 8.2% and 7.9% compared to odometry error, for the three agents respectively.

### B. Testing Algorithm 2 on real magnetic field measurements with simulated odometry noise

We investigate the effects of varying communication failure rates $\alpha$ on the difference between the distributed estimate from Algorithm 2 and the centralized estimate from Algorithm 1. To study the most challenging case, we assume the agents have the possibility to communicate only once for each average consensus problem. By using the approximation obtained through one step of average consensus, we see in Figure 4 that the distributed algorithm is able to give an improved position estimate compared to single-agent magnetic field SLAM for failure rates up until 80%.

Each average consensus problem will give a solution that is exactly corresponding to the centralized solution when the communication failure rate is zero [22]. Otherwise, average

(a) $t$ = 2 seconds       (b) $t$ = 7 seconds       (c) $t$ = 20 seconds       (d) $t$ = 80 seconds
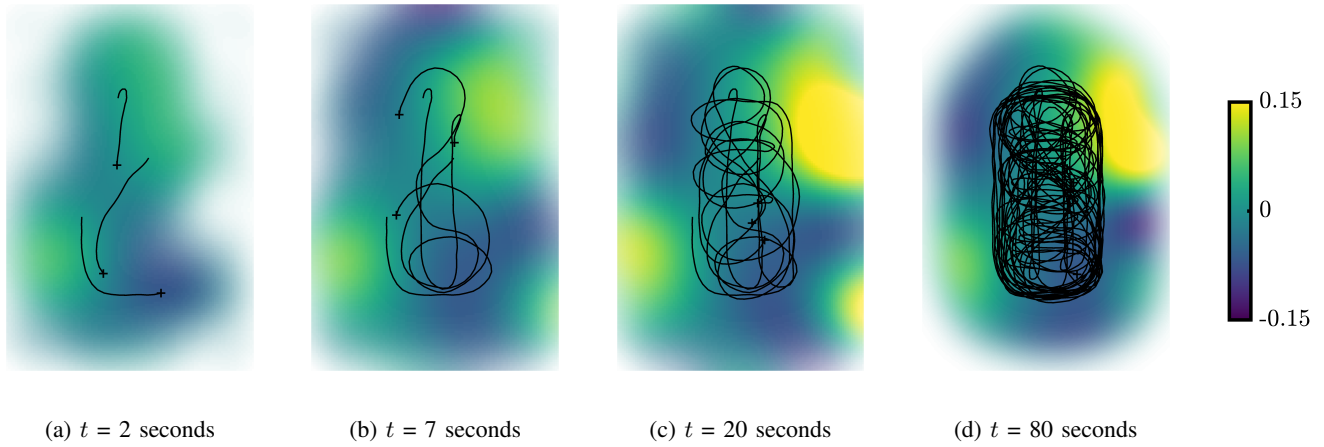
Fig. 2: Learned magnetic field map using Algorithm 1. The intensity of the learned magnetic field norm is indicated by the color, while the marginal variance of the magnetic field map is inversely proportional to the opacity. The estimated trajectories of the agents are indicated with black lines, and the current positions at each time are indicated with black crosses.
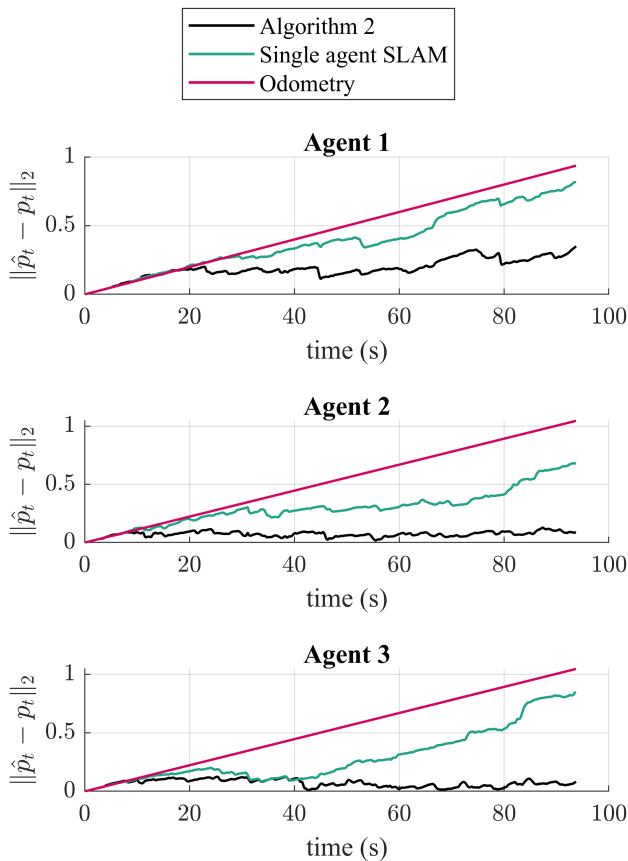


Fig. 3: Position estimation errors for three agents, relative to known ground truth position measured with optical motion capture system. The error is given as the euclidian distance between the true position $p_t$ and the estimated position $\hat{p}_t$ from Algorithm 2 with $N_c = 10$ and $\alpha = 0.2$, from Single agent SLAM and from integrating the pure odometry, respectively.
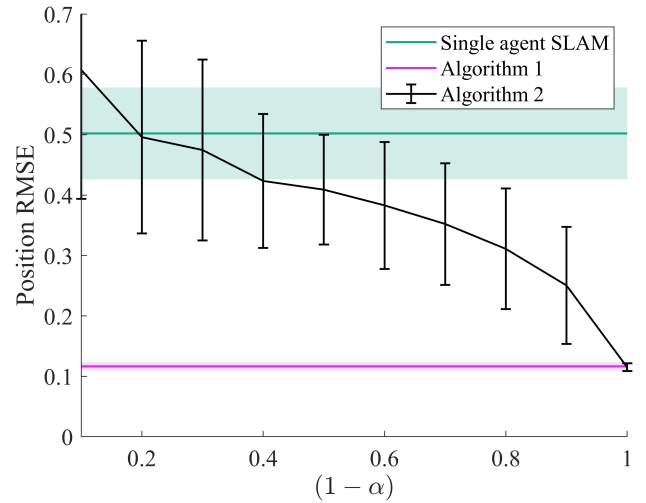


Fig. 4: RMSE of the full trajectory estimate using measurements from the motion capture lab, for a range of communication failure rates $\alpha$. The error bars indicate one standard deviation after 100 Monte-Carlo repetitions. The green line marks the average deviation in the position estimate between the single-agent SLAM solution and the centralized solution after 100 Monte-Carlo repetitions, and the light green area marks the range of one standard deviation. Algorithm 2 was run with $N_c = 1$.

consensus gives an approximation that converges to the true estimate as $N_c \to \infty$. The results in Fig. 4 confirm that the estimation error of Algorithm 1 is equivalent to the estimation error of Algorithm 2 when the dropout rate is zero. Furthermore, the results in Fig. 4 show that increasing the dropout rate $\alpha$, increases the estimation error of Algorithm 2. For all dropout rates of 80% or lower, the resulting position estimate from Algorithm 2 is closer to the centralized solution compared to the Single-agent SLAM. The results in Fig. 5 show that for higher $N_c$, the distributed estimate converges
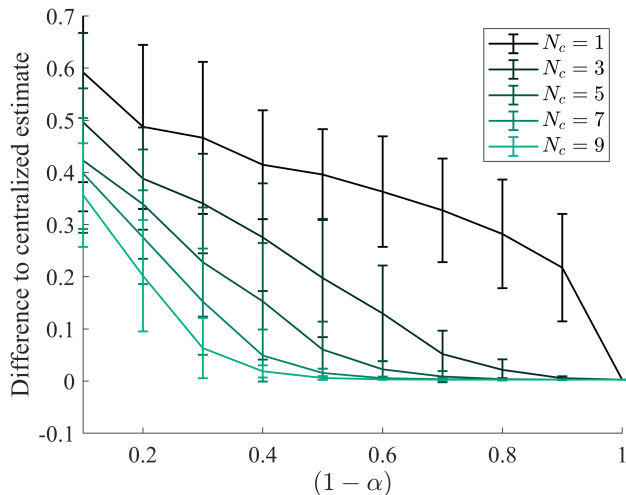
Fig. 5: Deviation between the estimate from Algorithm 1 and Algorithm 2 using measurements from the motion capture lab, for a range of communication rates $\alpha$, and a range of communication steps at each iteration $N_c$. The lines connect the average results after 100 MC repetitions, and the error bars indicate one standard deviation.
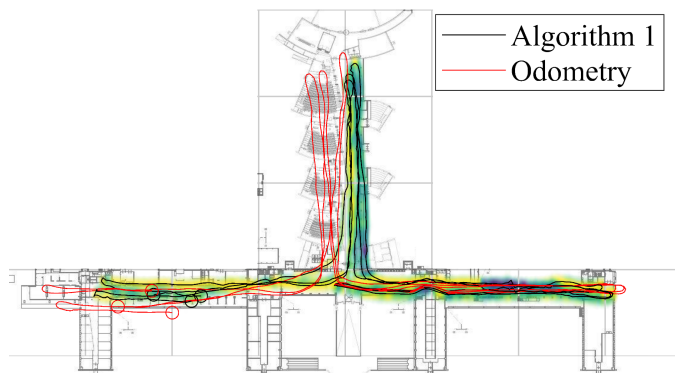


Fig. 6: Learned magnetic field map and estimated trajectories for three agents in a large building. The black circles indicate the estimated end positions of the agents using Algorithm 1, while the red circles indicate the estimated end positions of the agents using visual-inertial odometry. The color of the map is proportional to the learned intensity of the magnetic field norm, while the opacity is inversely proportional with the marginal variance.

more rapidly to the centralized estimate as $\alpha$ increases. When the communication failure rate is zero, so for $1 - \alpha = 1$, we can observe that the position estimate from the distributed EKF is equivalent to the position estimate from the centralized algorithm.

### C. Indoor experiment with three smartphone measurements

To test our algorithm on a larger scale experiment with real odometry and magnetic field norm measurements, we collected three sequences of visual-inertial odometry and magnetic field norm measurements inside a building using Google Pixel

smartphone. Google provides a platform primarily targeted at building augmented reality Apps called ARCore. Among other features, ARCore uses the phone's camera, accelerometer and gyroscope to compute a position and orientation estimate. Using a customized app, we simultaneously recorded this position and orientation estimate and the magnetometer measurements from the phone's built-in magnetometer at 200Hz. We subsequently computed the magnetic field norm using the three-component magnetic field measurements, and downsampled all measurements to 10 Hz. Algorithm 1 was applied to these three sequences as if they were collected by three separate agents simultaneously. The algorithm was applied with the following hyperparameters: $\sigma_{SE} = 7.2$, $l_{SE} = 1.2$m, $\sigma_y = 1.2$, $\sigma_p = 0.15$, $\sigma_q = 0.0001$ and with 500 basis functions in cubic tiles of size 38m×38m×38m. The tiles were placed with 8 meters of overlap at the borders. The resulting visual-inertial odometry estimate of the three trajectories is displayed in Fig. 6. The trajectories are illustrated with respect to the floor plan of the building where they were collected. The visual-inertial odometry is initially close to the real position, but over time, it drifts away from the hallways where the measurements were collected. In the same figure, the resulting position estimate of Algorithm 1 is displayed. These estimates are closer to the hallways where the measurements were collected, and therefore likely to have higher accuracy. The magnetic field map learned collaboratively by the three agents is displayed in Fig. 6.

## VII. CONCLUSION

For multiple agents navigating in a new environment, we proposed two algorithms that allow them to collaborate about solving the simultaneous mapping and localization task. The first algorithm can be employed when a central unit has access to all measurements from all agents. The second algorithm allows for multiple agents to collaboratively approximate the estimate of the first algorithm when there is no central station that can communicate with all agents at all times. Our proposed algorithms are capable of compensating for drift also in cases where single-agent SLAM fails to do so. We presented experimental results that confirm that the centralized multi-agent SLAM algorithm obtains a higher position accuracy compared to single-agent magnetic field SLAM. For our experimental results, the second algorithm was shown to give more accurate position estimates compared to single-agent SLAM for communication drop-out rates up until $80\%$.

### REFERENCES

[1] K.-K. Oh, M.-C. Park, and H.-S. Ahn, "A survey of multi-agent formation control," *Automatica*, vol. 53, pp. 424–440, Mar. 2015.
[2] M. Egerstedt and X. Hu, "Formation constrained multi-agent control," *IEEE Transactions on Robotics and Automation*, vol. 17, pp. 947–951, Dec. 2001.
[3] S. Sandeep, B. Fidan, and C. Yu, "Decentralized Cohesive Motion Control of Multi-Agent Formations," in *Proceedings of the 14th Mediterranean Conference on Control and Automation*, pp. 1–6, June 2006.

[4] P. Fankhauser, M. Bloesch, P. Krüsi, R. Diethelm, M. Wermelinger, T. Schneider, M. Dymczyk, M. Hutter, and R. Siegwart, "Collaborative navigation for flying and walking robots," in *Proceedings of the International Conference on Intelligent Robots and Systems (IROS)*, pp. 2859–2866, Oct. 2016.

[5] P. Puricer and P. Kovar, "Technical Limitations of GNSS Receivers in Indoor Positioning," in *Proceedings of the 17th International Conference Radioelektronika*, pp. 1–5, Apr. 2007.

[6] S. A. S. Mohamed, M.-H. Haghbayan, T. Westerlund, J. Heikkonen, H. Tenhunen, and J. Plosila, "A Survey on Odometry for Autonomous Navigation Systems," *IEEE Access*, vol. 7, pp. 97466–97486, 2019.

[7] O. J. Woodman, "An introduction to inertial navigation," 2007.

[8] D. Zou, P. Tan, and W. Yu, "Collaborative visual SLAM for multiple agents:A brief survey," *Virtual Reality & Intelligent Hardware*, vol. 1, pp. 461–482, Oct. 2019.

[9] A. Tourani, H. Bavle, J. L. Sanchez-Lopez, and H. Voos, "Visual SLAM: What Are the Current Trends and What to Expect?," *Sensors*, vol. 22, p. 9297, Jan. 2022.

[10] M. Osman, F. Viset, and M. Kok, "Indoor SLAM using a foot-mounted IMU and the local magnetic field," in *Proceedings of the 25th International Conference on Information Fusion (FUSION)*, pp. 1–7, July 2022.

[11] I. Vallivaara, J. Haverinen, A. Kemppainen, and J. Röning, "Magnetic field-based SLAM method for solving the localization problem in mobile robot floor-cleaning task," in *Proceedings of the 15th International Conference on Advanced Robotics (ICAR)*, pp. 198–203, June 2011.

[12] I. Vallivaara, J. Haverinen, A. Kemppainen, and J. R"oning, "Simultaneous localization and mapping using ambient magnetic field," in *IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems*, pp. 14–19, 2010.

[13] J. Coulin, R. Guillemard, V. Gay-Bellile, C. Joly, and A. de La Fortelle, "Online Magnetometer Calibration in Indoor Environments for Magnetic field-based SLAM," in *Proceedings of the 12th International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, pp. 1–8, Sept. 2022.

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

[15] P. Robertson, M. Angermann, and B. Krach, "Simultaneous Localization and Mapping for Pedestrians using only Foot-Mounted Inertial Sensors," in *ACM International Conference Proceeding Series*, Sept. 2009.

[16] P. Robertson, M. Frassl, M. Angermann, M. Doniec, B. J. Julian, M. Garcia Puyol, M. Khider, M. Lichtenstern, and L. Bruno, "Simultaneous Localization and Mapping for pedestrians using distortions of the local magnetic field intensity in large indoor environments," in *Proceeings of the International Conference on Indoor Positioning and Indoor Navigation*, pp. 1–10, Oct. 2013.

[17] J. Jung, S.-M. Lee, and H. Myung, "Indoor Mobile Robot Localization and Mapping Based on Ambient Magnetic Fields and Aiding Radio Sources," *IEEE Transactions on Instrumentation and Measurement*, vol. 64, no. 7, pp. 1922–1934, 2015.

[18] W. Storms, J. Shockley, and J. Raquet, "Magnetic field navigation in an indoor environment," in *Ubiquitous Positioning Indoor Navigation and Location Based Service*, pp. 1–10, Oct. 2010.

[19] G. Ouyang and K. Abed-Meraim, "A Survey of Magnetic-Field-Based Indoor Localization," *MDPI Electronics*, vol. 11, p. 864, Jan. 2022.

[20] 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*, pp. 913–920, Nov. 2013.

[21] F. Viset, R. Helmons, and M. Kok, "An Extended Kalman Filter for Magnetic Field SLAM Using Gaussian Process Regression," *Sensors*, vol. 22, p. 2833, Jan. 2022.

[22] R. Olfati-Saber, "Distributed Kalman Filter with Embedded Consensus Filters," in *Proceedings of the 44th IEEE Conference on Decision and Control*, pp. 8179–8184, Dec. 2005.

[23] G. Pillonetto, L. Schenato, and D. Varagnolo, "Distributed Multi-Agent Gaussian Regression via Finite-Dimensional Approximations," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 41, pp. 2098–2111, Sept. 2019.

[24] D. Jang, J. Yoo, C. Y. Son, D. Kim, and H. J. Kim, "Multi-Robot Active Sensing and Environmental Model Learning With Distributed Gaussian

[25] Process," *IEEE Robotics and Automation Letters*, vol. 5, pp. 5905–5912, Oct. 2020.

[25] S. Särkkä, *Bayesian Filtering and Smoothing*. Institute of Mathematical Statistics Textbooks, Cambridge: Cambridge University Press, 2013.

[26] L. Xiao, S. Boyd, and S. Lall, "A scheme for robust distributed sensor fusion based on average consensus," in *Proceedings of the 4th International Symposium on Information Processing in Sensor Networks, 2005.*, pp. 63–70, Apr. 2005.

[27] F. Viset, J. T. Gravdahl, and M. Kok, "Magnetic field norm SLAM using Gaussian process regression in foot-mounted sensors," in *Proceedings of the European Control Conference (ECC)*, (Delft, Netherlands), pp. 392–398, IEEE, June 2021.

[28] N. Wahlström, M. Kok, T. B. Schön, and F. Gustafsson, "Modeling magnetic fields using Gaussian processes," in *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 3522–3526, May 2013.

[29] A. Solin and S. Särkkä, "Explicit Link Between Periodic Covariance Functions and State Space Models," in *Proceedings of the Seventeenth International Conference on Artificial Intelligence and Statistics*, pp. 904–912, PMLR, Apr. 2014.

## APPENDIX

### A. Basis function definitions

The basis functions are defined over a finite-support cubical domain $\Omega \subset \mathbb{R}^d$, defined as $\Omega = [L_{l,1}, L_{u,1}] \times [L_{l,2}, L_{u,2}] \times [L_{l,3}, L_{u,3}]$. The basis functions are given as

$$\phi_i(p) = \prod_{d=1}^{3} \frac{\sqrt{2}}{\sqrt{L_{u,d} - L_{l,d}}} \sin\left(\frac{\pi n_{i,d}(p_d + L_{l,d})}{L_{u,d} - L_{l,d}}\right), \quad (25)$$

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 value of a parameter $\lambda_i$ defined as

$$\lambda_i = \sum_{d=1}^{D} \left(\frac{\pi n_{i,d}}{L_{u,d} - L_{l,d}}\right)^2, \quad (26)$$

as large as possible. These basis functions are then used to approximate the Gaussian process prior with a parametric prior

$$f \approx \Phi^\top w, \qquad w \sim \mathcal{N}(0, \Lambda), \quad (27)$$

where $\Phi$ is a vector of $M$ basis functions $\phi_i : \mathbb{R}^d \to \mathbb{R}$, $w \in \mathbb{R}^M$ is a vector of weights, and $\Lambda$ is defined as

$$\Lambda = \text{diag}\left[S_{\text{SE}}(\sqrt{\lambda_1}), \quad \cdots, \quad S_{\text{SE}}(\sqrt{\lambda_{N_m}})\right], \quad (28)$$

with $S_{\text{SE}}(\cdot)$ being the spectral density of the squared exponential kernel, as defined in [28]. This means that the approximation of the magnetic field norm in (27) has a prior distribution that tends to (3) as $M$ goes to infinity, and the size of the domain goes to infinity [29].