Heading Estimation Using Ultra-Wideband Received Signal Strength and Gaussian Processes

Daniil Lisus [ID], Charles Champagne Cossette [ID], Mohammed Shalaby [ID], and James Richard Forbes [ID]

Abstract—It is essential that a robot has the ability to determine its position and orientation to execute tasks autonomously. Heading estimation is especially challenging in indoor environments where magnetic distortions make magnetometer-based heading estimation difficult. Ultra-wideband (UWB) transceivers are common in indoor localization problems. This letter experimentally demonstrates how to use UWB range and received signal strength (RSS) measurements to estimate robot heading. The RSS of a UWB antenna varies with its orientation. As such, a Gaussian process (GP) is used to learn a data-driven relationship from UWB range and RSS inputs to orientation outputs. Combined with a gyroscope in an invariant extended Kalman filter, this realizes a heading estimation method that uses only UWB and gyroscope measurements.

Index Terms—Localization, sensor fusion, machine learning, Gaussian process regression.

I. INTRODUCTION

EXECUTION of robotic tasks relies heavily on the ability to estimate the position and orientation of the robot. In outdoor environments, GPS is a reliable positioning solution. In indoor environments, ultra-wideband (UWB) transceivers are frequently used for positioning [1]. However, the task of attitude estimation, particularly in indoor applications, is still an ongoing problem. Traditional methods use a magnetometer to predict a robot's orientation based on the local magnetic field. However, this field can be greatly distorted in indoor settings by metallic objects. Another approach, referred to position-aided inertial navigation, involves the use of inertial measurement unit (IMU) and position measurements to solve for the robot's orientation [2, Chapter 11]. However, heading becomes unobservable and starts to drift during motion with constant linear velocity, including when the robot is stationary or rotating in a fixed position [3]. In this letter, it is experimentally shown that UWB transceivers, commonly used to estimate position in indoor applications, can also be used to provide heading estimates.

UWB transceivers are attractive due to their low power, small size, low cost, decimeter-level ranging accuracy, and ability
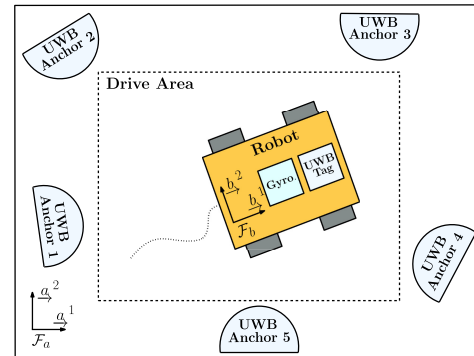
Fig. 1. A sample room setup of a robot equipped with a UWB tag and gyroscope and 5 stationary UWB anchors. The local frame $\mathcal{F}_a$ and robot body frame $\mathcal{F}_b$ are shown.

to act as a communication medium as well as a positioning device [4], when a system of static transceivers, called anchors, is present. The large bandwidth of UWB allows the transceivers to send nanosecond-duration radio impulses, resulting in an increased time resolution and resistance to noise caused by reflected signals [5].

Recently, several researchers have aimed to improve the quality of UWB range measurements by utilizing Gaussian Process (GP) models. These models have been used to correct for discrepancies caused by nonuniform radiation patterns of the UWB antenna and by non-line-of-sight communications [4], [6]–[8]. Since range measurements are used to estimate position, improved range measurements result in improved positioning accuracy. Existing literature focuses on the single-tag, multi-anchor approach where a single UWB module, called a tag, is mounted on a moving robot and measures the range to multiple stationary UWB modules, called anchors, with known positions. A depiction of this approach is shown in Fig. 1. Previous letters, such as [9], [10], have investigated predicting the angle of arrival of a received UWB signal. An approach for 2D pose estimation is shown in [9]. However, the method relies on extracting the channel impulse response, which is not possible with some off-the-shelf UWB modules. Attitude estimation based on GPS received signal strength (RSS) is shown in [11].

Compared to previous letters that use GP models to improve the accuracy of UWB range measurements, this letter explores how to use a GP model, trained using range and RSS inputs and ground truth heading outputs, to provide direct heading estimation of an indoor planar robot. The GP model's ability to "learn" a relationship between range, RSS, and heading is
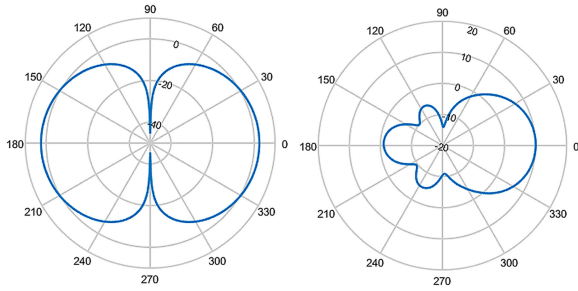
Fig. 2. A simulated radiation pattern for an ideal dipole antenna (left) and an ideal helical antenna (right). The directivity value, proportional to the RSS, in dBi is plotted as a function of the angle around the antenna.

Fig. 3. A high-level diagram of the proposed approach. The Gaussian Process (GP) block is pre-trained for a given anchor setup.

due to the heading-dependent RSS pattern associated with a UWB antenna [12]. This dependency is visualized in Fig. 2 for simulated ideal dipole and helical antennas. The pattern is unique for a given antenna and independent of the transceiver.

The GP model is capable of providing both a heading estimate and a corresponding uncertainty. However, to further enhance heading estimation, the GP model is used in the correction step of an invariant extended Kalman filter (IEKF) [13] that uses a gyroscope, another common sensor, within a prediction step. The proposed approach is capable of heading estimation using only a UWB module and a gyroscope, both of which are typically present in indoor setups that localize using an IMU and range measurements. A visualization of the proposed approach is shown in Fig. 3.

The rest of the letter is structured as follows. Section II provides a general overview of GP models and the IEKF. Section III formulates the state to be estimated. Section IV describes the GP implementation used to produce a heading prediction and Section V derives the IEKF filter that couples gyroscope data and the GP model. Finally, Section VI presents the performance of the proposed method on real data and Section VII provides concluding remarks.

II. PRELIMINARIES

A. Gaussian Process

A GP models a distribution over functions, which can be viewed as an infinite collection of random variables (RVs) [14]. Specifically, GPs attempt to capture the distribution over possible functions $f(\mathbf{x})$, provided with inputs $\mathbf{x} \in \mathbb{R}^d$ and outputs $y \in \mathbb{R}$. In reality, the measured outputs y are noisy and assumed to take the form

$$y = f(\mathbf{x}) + \epsilon, \tag{1}$$

where the noise $\epsilon \sim \mathcal{N}(0, \sigma^2)$ is normally distributed with variance σ^2. A GP is completely defined by a mean $m(\mathbf{x})$ and covariance function or *kernel* $k(\mathbf{x}, \mathbf{x}')$, which are explicitly stated as

$$m(\mathbf{x}) = \mathrm{E}[f(\mathbf{x})], \tag{2}$$

$$k(\mathbf{x}, \mathbf{x}') = \mathrm{E}[(f(\mathbf{x}) - m(\mathbf{x}))(f(\mathbf{x}') - m(\mathbf{x}'))]. \tag{3}$$

Using these expressions, a GP can be written as

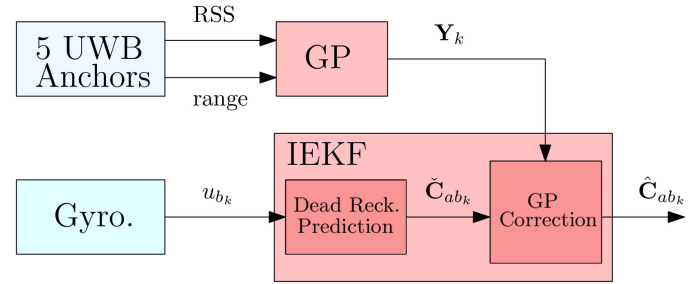$$f(\mathbf{x}) \sim \mathcal{GP}(m(\mathbf{x}), k(\mathbf{x}, \mathbf{x}')), \tag{4}$$

where the distribution of the output of the function $f(\mathbf{x})$ is estimated for a given input $\mathbf{x}$. Typically, this value is a scalar and multiple GPs are built to model a vector of output values. The construction of multiple-output GPs greatly increases the complexity and is an active area of research [15].

A GP model produces predictions by conditioning the query RV $y^\star$ corresponding to the test input $\mathbf{x}^\star \in \mathbb{R}^d$, on all of the available n training RVs, which are the set of measurements $\mathbf{y} = \begin{bmatrix} y_1 & y_2 & \cdots & y_n \end{bmatrix}^\mathsf{T} \in \mathbb{R}^n$ corresponding to inputs $\mathbf{X} = \begin{bmatrix} \mathbf{x}_1 & \mathbf{x}_2 & \cdots & \mathbf{x}_n \end{bmatrix}^\mathsf{T} \in \mathbb{R}^{n \times d}$. The mean and covariance functions of the conditional distribution of $y^\star | \mathbf{y}, \mathbf{X}, \mathbf{x}^\star, \boldsymbol{\theta}$ are given by

$$m(f(\mathbf{x}^\star)) = \mathbf{K}(\mathbf{x}^\star, \mathbf{X})(\mathbf{K}(\mathbf{X}, \mathbf{X}))^{-1}\mathbf{y}, \tag{5}$$

$$\mathrm{cov}(f(\mathbf{x}^\star)) = \mathbf{K}(\mathbf{x}^\star, \mathbf{x}^\star) - \mathbf{K}(\mathbf{x}^\star, \mathbf{X})(\mathbf{K}(\mathbf{X}, \mathbf{X}))^{-1}\mathbf{K}(\mathbf{X}, \mathbf{x}^\star), \tag{6}$$

where $\mathbf{K}(\cdot, \cdot)$ is the covariance matrix, and $\boldsymbol{\theta}$ is the collection of hyperparameters within the covariance function. The covariance matrix has the form $\mathbf{K}_{ij}(\mathbf{X}, \mathbf{X}') = k(\mathbf{X}_i, \mathbf{X}'_j)$ and is sized according to the number of inputs that $\mathbf{X}$ and $\mathbf{X}'$ have. For example, $\mathbf{K}(\mathbf{x}^\star, \mathbf{X}) \in \mathbb{R}^{1 \times n}$. Training a GP is the act of optimizing

$$\boldsymbol{\theta}^\star = \underset{\boldsymbol{\theta}}{\mathrm{argmax}} \, \log(p(\mathbf{y}|\mathbf{X}, \boldsymbol{\theta})) \tag{7}$$

in order to find the hyperparameters $\boldsymbol{\theta}^\star$ that maximize the log maximum likelihood $\log(p(\mathbf{y}|\mathbf{X}, \boldsymbol{\theta}))$, which is the probability that the training inputs $\mathbf{X}$ produce the measured training outputs $\mathbf{y}$ through the modelled functional distribution.

B. Matrix Lie Groups

A matrix Lie group G is a set of invertible square matrices that form a group under the operation of matrix multiplication [16]. The matrix Lie algebra associated with a matrix Lie group G is denoted as $\mathfrak{g}$ and is the tangent space at the identity element on the matrix Lie group manifold. The exponential map $\exp(\cdot) : \mathfrak{g} \to G$ and the logarithm map $\log(\cdot) : G \to \mathfrak{g}$ provide a way to transition between these spaces. The "vee" operator $(\cdot)^\vee : \mathfrak{g} \to \mathbb{R}^q$ and the "wedge" operator $(\cdot)^\wedge : \mathbb{R}^q \to \mathfrak{g}$ map the matrix Lie algebra to and from the vector space $\mathbb{R}^q$, for the q degrees of freedom of the matrix Lie group elements. Using these operators, it is possible to express an error on a matrix Lie group element $\mathbf{X} \in G$ as either a left or right multiplication. A

left-invariant error, which is used throughout this letter, is of the form $\exp(\delta\boldsymbol{\xi}^\wedge) = \mathbf{X}^{-1}\bar{\mathbf{X}}$, where $\bar{\mathbf{X}} \in G$ is a nominal matrix Lie group element and $\delta\boldsymbol{\xi} \in \mathbb{R}^k$ is associated with the error $\exp(\delta\boldsymbol{\xi}^\wedge) \in G$. An error of this form is said to be left-invariant, since $\mathbf{X}$ and $\bar{\mathbf{X}}$ can be left multiplied by another element $\mathbf{X}' \in G$ without any change in the overall error expression.

C. Invariant Extended Kalman Filter

The IEKF is a variation of the extended Kalman filter (EKF) that can be used to directly estimate elements of matrix Lie groups [13]. The IEKF uses a particular error definition, which often leads to Jacobians that are state-estimate independent. Additionally, the IEKF frequently has enhanced transient performance. In the case of attitude estimation, the IEKF enables estimation of the $SO(2)$ or $SO(3)$ matrix Lie group elements directly.

The IEKF is constructed by linearizing the process and measurement models. The linearized process model is of the form

$$\delta\boldsymbol{\xi}_k = \mathbf{A}_{k-1}\delta\boldsymbol{\xi}_{k-1} + \delta\mathbf{w}_{k-1}, \tag{8}$$

where $\exp(\delta\boldsymbol{\xi}_k^\wedge) = \mathbf{X}^{-1}\bar{\mathbf{X}}$, and $\delta\mathbf{w}_{k-1} \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}_{k-1})$ is white Gaussian process noise with covariance matrix $\mathbf{Q}_{k-1}$. The linearized measurement model is of the form

$$\delta\boldsymbol{\xi}_k^y = \mathbf{C}_k\delta\boldsymbol{\xi}_k + \mathbf{M}_k\delta\boldsymbol{\epsilon}_k, \tag{9}$$

where $\delta\boldsymbol{\xi}_k^y$ is a small change in the measurement $\mathbf{y}_k$, and $\delta\boldsymbol{\epsilon}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{R}_k)$ is white Gaussian measurement noise with covariance matrix $\mathbf{R}_k$.

The prediction step of the IEKF is $\check{\mathbf{X}}_k = \mathbf{F}(\hat{\mathbf{X}}_{k-1}, \mathbf{u}_{k-1})$, where $\mathbf{u}_{k-1}$ is the interoceptive measurement, $\hat{\mathbf{X}}_{k-1}$ is the corrected state estimate at t_{k-1}, $\mathbf{F}(\hat{\mathbf{X}}_{k-1}, \mathbf{u}_{k-1})$ is the nonlinear process model, and $\check{\mathbf{X}}_k$ is the predicted state at t_k. The predicted covariance is

$$\check{\mathbf{P}}_k = \mathbf{A}_{k-1}\hat{\mathbf{P}}_{k-1}\mathbf{A}_{k-1}^\mathsf{T} + \mathbf{Q}_{k-1}. \tag{10}$$

The correction step requires the Kalman gain, that being

$$\mathbf{K}_k = \check{\mathbf{P}}_k\mathbf{C}_k^\mathsf{T}(\mathbf{C}_k\check{\mathbf{P}}_k\mathbf{C}_k^\mathsf{T} + \mathbf{M}_k\mathbf{R}_k\mathbf{M}_k^\mathsf{T})^{-1}. \tag{11}$$

Using a left-invariant error definition, the correction step is

$$\hat{\mathbf{X}}_k = \check{\mathbf{X}}_k\exp\left[-(\mathbf{K}_k(\check{\mathbf{x}}_k^{-1}\mathbf{z}_k))^\wedge\right], \tag{12}$$

where $\mathbf{z}_k$ is the *innovation* or the difference between the measurement and the expected measurement based on the measurement model and $\check{\mathbf{X}}_k$. The corrected covariance is

$$\hat{\mathbf{P}}_k = (\mathbf{1} - \mathbf{K}_k\mathbf{C}_k)\check{\mathbf{P}}_k(\mathbf{1} - \mathbf{K}_k\mathbf{C}_k)^\mathsf{T} + \mathbf{K}_k\mathbf{M}_k\mathbf{R}_k\mathbf{M}_k^\mathsf{T}\mathbf{K}_k^\mathsf{T}. \tag{13}$$

The IEKF formulation is presented in a general matrix form, indicated by the bolded letters. However, when applied to an $SO(2)$ problem, as is done in Section III of this letter, many elements are scalar, and are not bolded.

III. PROBLEM DEFINITION

The goal is to estimate the discrete-time matrix Lie group state

$$\mathbf{X}_k = \mathbf{C}_{ab_k} \in SO(2), \tag{14}$$

where $\mathbf{C}_{ab_k}$ is the 2D attitude of the body frame $\mathcal{F}_b$ relative to the local frame $\mathcal{F}_a$. The two frames are visualized in Figure 1. The inverse of the state is simply the transpose of the element, and is written as $\mathbf{X}_k^{-1} = \mathbf{C}_{ab_k}^\mathsf{T}$.

The state $\mathbf{X}_k = \mathbf{C}_{ab_k} = \exp(\xi_k^\wedge) \in SO(2)$ has one degree of freedom, the "yaw" angle $\xi_k^\theta \in (-\pi, \pi]$. The matrix Lie algebra associated with $SO(2)$ is

$$\xi^{\theta\wedge} = \begin{bmatrix} 0 & -\xi^\theta \\ \xi^\theta & 0 \end{bmatrix} \in \mathfrak{so}(2). \tag{15}$$

The exponential map for $SO(2)$ is given by

$$\exp(\xi^{\theta\wedge}) = \begin{bmatrix} \cos(\xi^\theta) & -\sin(\xi^\theta) \\ \sin(\xi^\theta) & \cos(\xi^\theta) \end{bmatrix} \in SO(2). \tag{16}$$

IV. GAUSSIAN PROCESS IMPLEMENTATION

In order to produce an estimate for the full $SO(2)$ element, two separate GPs are trained, one to predict $\sin(\xi^\theta)$ and another to predict $\cos(\xi^\theta)$. Estimating the $SO(2)$ element directly avoids issues arising from ambiguities and angle wrap-around. However, one cannot guarantee that the GP predictions will be restricted to $[-1, 1]$, nor that their outputs together satisfy $\sin(\xi^\theta)^2 + \cos(\xi^\theta)^2 = 1$. As such, these "pseudo" sine and cosine predictions of the true $\sin(\xi^\theta)$ and $\cos(\xi^\theta)$ are denoted simply as s and c, respectively. The range and RSS measurements from each of the 5 anchors in the room are used as inputs for both GPs. A motion capture system provides ground truth heading that is used for the training outputs and for test validation.

A. Kernel Selection and Training

Each of the GPs use the standard squared exponential (SE) kernel for the covariance function, which can be written for two arbitrary inputs $\mathbf{x}$ and $\mathbf{x}'$ as [17]

$$k(\mathbf{x}, \mathbf{x}') = \sigma_f^2\exp\left(-\frac{1}{2\sigma_\ell^2}\|\mathbf{x} - \mathbf{x}'\|^2\right), \tag{17}$$

where σ_ℓ and σ_f are the hyperparameters to be optimized and are known as the *characteristic lengthscale* and *signal variance*, respectively [14, Chapter 2.3]. The hyperparameters are optimized using the MATLAB `fitrgp` library, which selects hyperparameters that minimize the five-fold cross-validation loss using the training data [14, Chapter 5.3]. The SE kernel is chosen after experimentation due to its simplicity and comparable-or-better performance compared to other standard kernels, such as the Matern or ARD kernels.

B. Gaussian Process Output Normalization

The full $SO(2)$ element can be assembled from the individual s and c components through the relationship (16). However, since there is no guarantee that the produced element would satisfy the constraint $s^2 + c^2 = 1$, the outputs of the two GPs are first normalized and then assembled into a proper $SO(2)$ element using

$$\mathbf{Y}_k = \frac{1}{\sqrt{s_k^2 + c_k^2}} \begin{bmatrix} c_k & -s_k \\ s_k & c_k \end{bmatrix} \in SO(2), \tag{18}$$

where s_k and c_k are GP predictions at timestep t_k.

The normalized prediction $\mathbf{Y}_k$ is a nonlinear function of the independently predicted sine and cosine terms, each with their respective uncertainties. As such, the uncertainty of the resulting normalized element is approximated through a standard linearization procedure. The GP outputs s_k and c_k are random variables with mean $\bar{s}_k$ with variance R^s, and mean $\bar{c}_k$ with variance R^c, respectively. For brevity, the subscript k is omitted. Equation (18) can be written as

$$\mathbf{Y} = \frac{1}{\sqrt{s^2 + c^2}} \left(c \underbrace{\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}}_{\mathbf{1}} + s \underbrace{\begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}}_{\boldsymbol{\Omega}} \right). \quad (19)$$

The perturbation of the first term in (19) can be approximated with a first-order expansion as

$$\frac{1}{\sqrt{s^2 + c^2}} \approx \underbrace{\frac{1}{\sqrt{\bar{s}^2 + \bar{c}^2}}}_{\alpha_1} + \underbrace{\frac{-\bar{c}}{(\bar{s}^2 + \bar{c}^2)^{\frac{3}{2}}}}_{\alpha_2} \delta c + \underbrace{\frac{-\bar{s}}{(\bar{s}^2 + \bar{c}^2)^{\frac{3}{2}}}}_{\alpha_3} \delta s. \quad (20)$$

This expansion can be included in the general perturbation of both sides of (19) as

$$\bar{\mathbf{Y}}(\mathbf{1} - \delta\xi^{y\wedge}) = (\alpha_1 + \alpha_2 \delta c + \alpha_3 \delta s)\left((\bar{c} + \delta c)\mathbf{1} + (\bar{s} + \delta s)\boldsymbol{\Omega} \right),$$

$$-\bar{\mathbf{Y}}\delta\xi^{y\wedge} = \underbrace{\left((\alpha_1 + \alpha_2 \bar{c})\mathbf{1} + \alpha_2 \bar{s}\boldsymbol{\Omega} \right)}_{\mathbf{D}} \delta c$$

$$+ \underbrace{\left((\alpha_1 + \alpha_3 \bar{s})\boldsymbol{\Omega} + \alpha_3 \bar{c}\mathbf{1} \right)}_{\mathbf{E}} \delta s,$$

$$\delta\xi^y = \left(-\bar{\mathbf{Y}}^{-1}\mathbf{D} \right)^{\vee} \delta c + \left(-\bar{\mathbf{Y}}^{-1}\mathbf{E} \right)^{\vee} \delta s. \quad (21)$$

The final variance R^θ on the normalized element $\mathbf{y}$ is then

$$R^\theta = \left(-\bar{\mathbf{Y}}^{\mathsf{T}}\mathbf{D} \right)^{\vee} R^c \left(-\bar{\mathbf{Y}}^{\mathsf{T}}\mathbf{D} \right)^{\vee\mathsf{T}} + \left(-\bar{\mathbf{Y}}^{\mathsf{T}}\mathbf{E} \right)^{\vee} R^s \left(-\bar{\mathbf{Y}}^{\mathsf{T}}\mathbf{E} \right)^{\vee\mathsf{T}}, \quad (22)$$

where $\bar{\mathbf{Y}}^{\mathsf{T}} = \bar{\mathbf{Y}}^{-1}$. This uncertainty is used in Section V-B.

V. FILTER FORMULATION

A. Process Model

The continuous-time attitude kinematics are given by Poisson's equation

$$\dot{\mathbf{C}}_{ab} = \mathbf{C}_{ab}\omega_b^{ba\wedge}, \quad (23)$$

where $\omega_b^{ba} \in \mathbb{R}$ is the angular velocity of the body frame relative to the local frame, resolved in the body frame, and time dependence is dropped for notational convenience.

The interoceptive sensor in this problem, the gyroscope, measures angular velocity subject to noise as

$$u_b = \omega_b^{ba} - w_b, \quad (24)$$

where $w_b \sim \mathcal{N}(0, Q_c(t)\delta(t - \tau))$ is the zero-mean additive noise, $Q_c(\cdot)$ is the power spectral density, and $\delta(\cdot)$ is the Dirac delta function. Equation (23) can thus be rewritten as

$$\dot{\mathbf{C}}_{ab} = \mathbf{C}_{ab}(u_b + w_b)^{\wedge}. \quad (25)$$

In order to achieve the linearized model (8), the left-invariant error definition,

$$\delta\mathbf{C} = \mathbf{C}_{ab}^{-1}\check{\mathbf{C}}_{ab} = \mathbf{C}_{ab}^{\mathsf{T}}\check{\mathbf{C}}_{ab}, \quad (26)$$

is used. A left-invariant error is chosen in order to be consistent with the left-invariant measurement model. The rate of change of the error (26) is

$$\delta\dot{\mathbf{C}} = \mathbf{C}_{ab}^{\mathsf{T}}\dot{\check{\mathbf{C}}}_{ab} + \dot{\mathbf{C}}_{ab}^{\mathsf{T}}\check{\mathbf{C}}_{ab}$$

$$= \mathbf{C}_{ab}^{\mathsf{T}}\check{\mathbf{C}}_{ab}(u_b + \check{w}_b)^{\wedge} - (u_b + w_b)^{\wedge}\mathbf{C}_{ab}^{\mathsf{T}}\check{\mathbf{C}}_{ab}$$

$$= \delta\mathbf{C}u_b^{\wedge} - (u_b + w_b)^{\wedge}\delta\mathbf{C}. \quad (27)$$

(27) can be linearized by approximating $\delta\mathbf{C} \approx \mathbf{1} + \delta\xi^{\theta\wedge}$, $w_b \approx \delta w_b$ and simplifying as

$$\delta\dot{\xi}^{\theta\wedge} = (\mathbf{1} + \delta\xi^{\theta\wedge})u_b^{\wedge} - (u_b + \delta w_b)^{\wedge}(\mathbf{1} + \delta\xi^{\theta\wedge})$$

$$= u_b^{\wedge} + \delta\xi^{\theta\wedge}u_b^{\wedge} - u_b^{\wedge} - \delta w_b^{\wedge} - u_b^{\wedge}\delta\xi^{\theta\wedge}$$

$$\approx -\delta w_b^{\wedge} + \delta\xi^{\theta\wedge}u_b^{\wedge} - u_b^{\wedge}\delta\xi^{\theta\wedge} \quad (28)$$

$$= -\delta w_b^{\wedge},$$

$$\delta\dot{\xi}^{\theta} = -\delta w_b.$$

The final continuous-time linearized model is

$$\delta\dot{\xi}^{\theta} = \underbrace{0}_{A_c}\delta\xi^{\theta} + \underbrace{-1}_{L_c}\delta w_b, \quad (29)$$

where the subscript c implies continuous time. The discrete time A_{k-1} and Q_{k-1} matrices needed for the discrete time form (8) can be computed from A_c, L_c, and $Q_c(t)$ based on [2, Chapter 3.5.5]. The discrete-time linearized process model, of the form (8), is

$$\delta\xi_k^{\theta} = \underbrace{1}_{A_{k-1}}\delta\xi_{k-1}^{\theta} + \delta w_{b_{k-1}}, \quad (30)$$

where $\delta w_{b_{k-1}} \sim \mathcal{N}(0, Q_{k-1})$.

B. Measurement Model

The GP runs separately from the filter and produces individual independent components of an $SO(2)$ element. These outputs are then normalized as per (18) to form a full element of the group, which is then used as a heading measurement in the correction step of the filter. The GP also produces a variance R_k^{θ} on this element as per (22). The discrete-time measurement is then defined as

$$\mathbf{Y}_k = \mathbf{C}_{ab_k}\exp(\epsilon_k^{\theta\wedge}), \quad (31)$$

where $\epsilon_k^{\theta} \sim \mathcal{N}(0, R_k^{\theta})$ and which is left-invariant according to the definition in Section II-B. This is similar to the error definition used on quaternions in [18] and on $SE(3)$ matrix Lie group elements in [19]. As such, linearization can be done directly on the measurement model by rewriting both $\mathbf{Y}_k$ and $\mathbf{C}_{ab_k}$ in terms of the left-invariant error as

$$\check{\mathbf{Y}}_k\delta\mathbf{Y}_k^{-1} = \check{\mathbf{C}}_{ab_k}\delta\mathbf{C}_k^{-1}\exp(\epsilon_k^{\theta\wedge}), \quad (32)$$

where $\check{\mathbf{Y}}_k = \check{\mathbf{C}}_{ab_k}$, allowing the simplification

$$\delta\mathbf{Y}_k^{-1} = \delta\mathbf{C}_k^{-1}\exp(\epsilon_k^{\theta\wedge}). \quad (33)$$
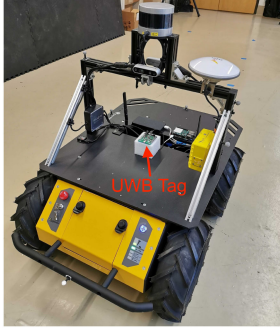
Fig. 4. The experimental setup of a Pozyx UWB Developer Tag mounted on a Clearpath Husky mobile robot.
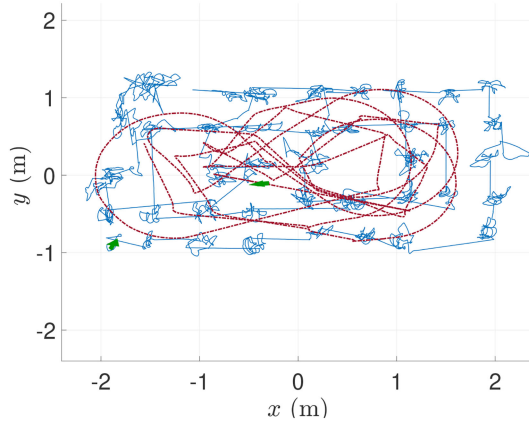

Fig. 5. The training path used for datasets Husky #1 and #2 (blue) and the test path of dataset Husky #2 (red). Green arrows show the starting poses of the robot for both paths.

This can be linearized by substituting $\delta \mathbf{Y}_k^{-1} \approx \mathbf{1} - \delta \xi_k^{y^\wedge}$, $\delta \mathbf{C}_k^{-1} \approx \mathbf{1} - \delta \xi_k^{\theta^\wedge}$, $\epsilon_k^\theta \approx \delta \epsilon_k^\theta$, $\exp(\epsilon_k^{\theta^\wedge}) \approx \mathbf{1} + \delta \epsilon_k^{\theta^\wedge}$, and ignoring higher order terms to yield

$$\mathbf{1} - \delta \xi_k^{y^\wedge} = (\mathbf{1} - \delta \xi_k^{\theta^\wedge})(\mathbf{1} + \delta \epsilon_k^{\theta^\wedge}),$$

$$\delta \xi_k^y = \underbrace{1}_{\mathbf{C}_k} \delta \xi_k^\theta + \underbrace{-1}_{\mathbf{M}_k} \delta \epsilon_k^\theta. \tag{34}$$

The innovation in vector space can then be defined following the left error definition as

$$z_k = \log(\mathbf{Y}_k^{-1} \check{\mathbf{Y}}_k)^\vee, \tag{35}$$

where $\check{\mathbf{Y}}_k = \check{\mathbf{C}}_{ab_k}$.

VI. RESULTS

The GP and IEKF are tested on real data collected using 5 Pozyx UWB Creator Anchor modules with a Pozyx UWB Developer Tag mounted on a Clearpath Husky mobile robot, as shown in Fig. 4. The Pozyx modules use the DW1000 transceiver and a linearly polarized UWB chip antenna. The radiation pattern of the antenna is not provided by the supplier, further motivating the use of a GP. The training data is collected by moving and rotating the robot in an area of approximately $4 \text{ m} \times 2 \text{ m}$ for roughly 86 000 data points. Two test sets are collected in the same area, with care taken to follow a path

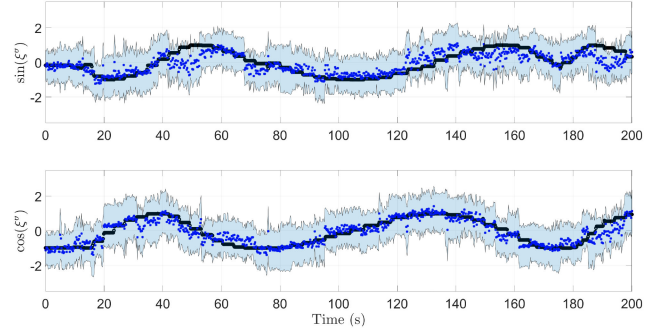|	Dataset	Husky #1	Husky #2	Manual
$\sin(\xi^\theta)$	RMSE	0.52	0.55	0.49
	Mean $\pm 3\sigma$	1.35	1.35	0.79
$\cos(\xi^\theta)$	RMSE	0.55	0.56	0.45
	Mean $\pm 3\sigma$	1.28	1.28	0.89
	No. Train Points	17,274	17,274	25,747
	No. Test Points	15,737	3,937	71,333

Fig. 6. The ground truth $\sin(\xi^\theta)$ and $\cos(\xi^\theta)$ (black), overlaid with the GP estimates (dark blue) and their $\pm 3\sigma$ bounds (light blue). Results are produced from the manually collected dataset, with the test set down-sampled and shortened for visual clarity.

different from the training data and each other. Ground truth is recorded using a motion capture system to evaluate performance. The two test sets are referred to as the "Husky #1" and "Husky #2" datasets. The training path and one of the test paths is shown in Fig. 5.

The results for a "Manual" dataset are also provided for completeness. This dataset is collected by manually moving a Pozyx UWB Developer Tag in an area of approximately $2 \text{ m} \times 2 \text{ m}$ for roughly 200 000 data points. The setup for this dataset is otherwise identical to the Husky ones.

A. Gaussian Process

The numerical results of the raw GP predictions of the $SO(2)$ components $\sin(\xi^\theta)$ and $\cos(\xi^\theta)$ are shown in Table I. A visualization is shown in Fig. 6. Although it is clear that a relationship is learned, the prediction quality is poor. A possible reason is that the Pozyx Developer Tags only provide RSS measurements to the nearest integer value in dBi. In these experiments, the RSS measurements vary by around 10 dBi, meaning that only a few RSS values are fed into the GP. Thus, it is likely that the accuracy could be improved with higher-resolution RSS measurements.

B. Filter

In order to evaluate the proposed framework, three different estimators are considered.

1) Dead-Reckoning: In order to validate the benefit of adding the GP model as a measurement step in an IEKF with a gyroscope-based prediction, a purely gyroscope-based dead reckoning estimation approach is considered.

2) Magnetometer Correction: To provide a comparison to more classic indoor heading estimation methods, a magnetometer-based IEKF (mag IEKF) is also run. This filter
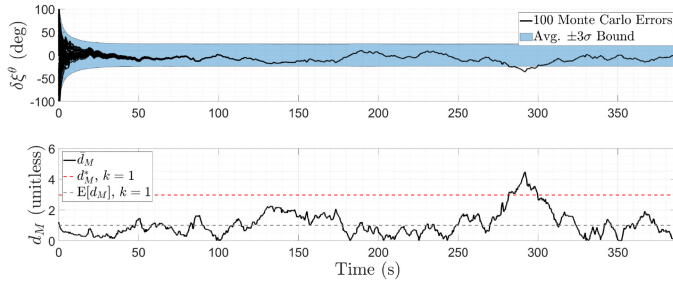
Fig. 7. Top: The errors $\delta\xi^\theta$ and average $\pm3\sigma$ bounds of the GP IEKF estimate from 100 Monte Carlo runs. Bottom: The average Mahalanobis distance of the GP IEKF error from 100 Monte Carlo runs plotted with the 99.7% 1DOF bound (red), and the expected mean distance (grey).
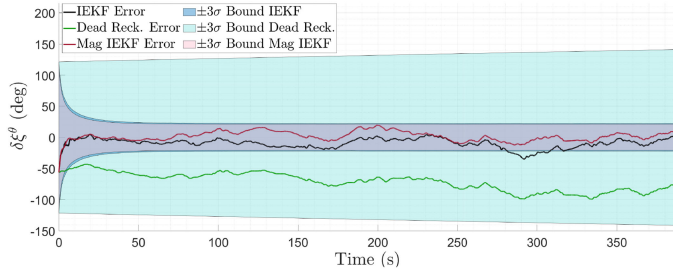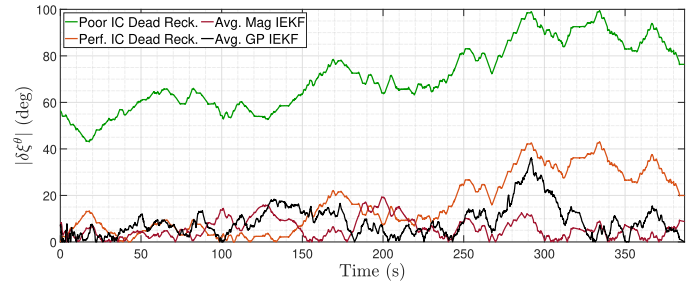

Fig. 9. The absolute value of the error $\delta\xi^\theta$ for perfectly (orange) and imperfectly (green) initialized dead reckoning, and for 100 Monte Carlo averaged mag IEKF (red) and GP IEKF (black).

Fig. 8. The error $\delta\xi^\theta$ and the $\pm3\sigma$ bounds produced by dead reckoning (green), the mag IEKF filter (red), and the GP IEKF filter (black) initialized with 1 rad of error.

TABLE II
INVARIANT EXTENDED KALMAN FILTER RESULTS

	Dataset	Husky #1	Husky #2	Manual
GP	ξ^θ RMSE (deg)	9.74	9.16	11.45
	Steady $\pm3\sigma$ (deg)	23.49	23.49	22.92
Mag	ξ^θ RMSE (deg)	7.90	9.14	27.61
	Steady $\pm3\sigma$ (deg)	20.93	20.91	38.37
	Test Time (sec)	386.87	96.42	786.68

is identical to the filter outlined in Section V, except that the $SO(2)$ measurement in the correction step is produced from magnetometer data, as opposed to from a GP estimate. The magnetometer data is calibrated for bias and distortion from each respective training set. The local magnetic field vector is also computed based on each training set independently. The mag IEKF is tested using 100 Monte Carlo runs, where the filter is run on experimental data with a random initial prediction sampled from $\mathcal{N}(\xi^\theta_{0\,\text{true}}, 1.0\ \text{rad}^2)$. The numerical results are presented in Table II. The poor performance on the Manual dataset is attributed to the significantly closer proximity of the UWB tag, which includes the magnetometer, to the floor and the pipes and other metallic objects below it. This likely impacted the magnetometer measurements by both causing more outliers during the test run and also decreasing the quality of the calibration procedure that was run on the training data.

3) Gaussian Process Correction: An IEKF with a GP correction step, as outlined in Section V, is tested on experimental data using 100 Monte Carlo runs with a random initial prediction sampled from $\mathcal{N}(\xi^\theta_{0\,\text{true}}, 1.0\ \text{rad}^2)$. The numerical results for all datasets are presented in Table II.

For the Husky #1 dataset, the filter produces an average RMSE of 9.74 degrees, with an average $\pm3\sigma$ bound of 23.49 degrees. The full Monte Carlo results are visualized in Fig. 7. The filter is consistent and recovers from any initial prediction error. The only exception to the filter consistency occurs at approximately 292 s, where the error and Mahalanobis distance both go outside of their respective bounds. At this time, the robot drastically changes its heading and the predictions generated using the gyroscope and the GP are both poor. Similar spikes in the error of the dead reckoning predictions can be observed in the same time period. Although the filter temporarily deviates, consistency is quickly recovered.

A comparison between the estimates produced on the Husky #1 dataset by the various estimators is presented in Fig. 8, with Fig. 9 showing the absolute value of the error over time. As expected, the dead reckoning prediction suffers from an inability to correct initial errors and slowly drifts even in perfectly initialized scenarios. The mag IEKF slightly outperforms the GP IEKF on the Husky datasets, although the performance is comparable. On the Manual dataset, magnetic field disturbances caused by proximity to the floor likely result in the poor performance of the mag IEKF.

VII. CONCLUSION

This letter shows the feasibility of estimating the heading of a vehicle using UWB range and RSS measurements to 5 stationary anchors in a GP model. When combined with a gyroscope in an IEKF framework, the GP estimates are shown to realize convergence from arbitrary initial conditions. The filter produces a heading estimate with an RMSE of approximately 10 degrees and a $\pm3\sigma$ bound of approximately 23 degrees across three trials. It is hypothesized that with access to higher resolution RSS measurements, the GP estimate and overall heading prediction could be improved.

REFERENCES

[1] Z. Sahinoglu, S. Gezici, and I. Guvenc, *Ultra-Wideband Positioning Systems: Theoretical Limits, Ranging Algorithms, and Protocols*. Cambridge, England, United Kingdom: Cambridge University Press, 2008.
[2] J. Farrell, *Aided Navigation: GPS With High Rate Sensors*. New York, New York, USA: McGraw-Hill Education, 2008.

[3] M. H. Lee, J. H. Lee, Y. H. Koh, H. G. Park, J. H. Moon, and S. P. Hong, "Observability and estimability analysis of the GPS and INS in the vehicle," *J. Mech. Syst. Transp. Logistics*, vol. 3, no. 3, pp. 537–551, 2010.

[4] A. Ledergerber and R. D'Andrea, "Ultra-wideband range measurement model with gaussian processes," in *Proc. IEEE Conf. Control Technol. Appl.*, 2017, pp. 1929–1934.

[5] L. Zwirello, T. Schipper, M. Harter, and T. Zwick, "UWB localization system for indoor applications: Concept, realization and analysis," *J. Elect. Comput. Eng.*, vol. 2012, pp. 1–11, Mar. 2012.

[6] V. Savic, E. G. Larsson, J. Ferrer-Coll, and P. Stenumgaard, "Kernel methods for accurate UWB-Based ranging with reduced complexity," *IEEE Trans. Wireless Commun.*, vol. 15, no. 3, pp. 1783–1793, Mar. 2016.

[7] A. Ledergerber and R. D'Andrea, "Calibrating away inaccuracies in ultra wideband range measurements: A maximum likelihood approach," *IEEE Access*, vol. 6, pp. 78719–78730, 2018.

[8] W. Zhao, A. Goudar, J. Panerati, and A. P. Schoellig, "Learning-based bias correction for ultra-wideband localization of resource-constrained mobile robots," 2020, *arXiv:2003.09371v1*.

[9] A. Ledergerber, M. Hamer, and R. D'Andrea, "Angle of arrival estimation based on channel impulse response measurements," in *2019 IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, 2019, pp. 6686–6692.

[10] I. Dotlic, A. Cornell, H. Ma, J. Clancy, and M. McLaughlin, "Angle of arrival estimation using decawave DW1000 integrated circuits," in *Proc. 14th Workshop Positioning, Navigation Commun.*, 2017, pp. 1–6.

[11] J. Gross, Y. Gu, and M. Rhudy, "Fixed-wing UAV attitude estimation using single antenna GPS signal strength measurements," *Aerospace*, vol. 3, no. 2, p. 14, 2016.

[12] J. Chen, D. Raye, W. Khawaja, P. Sinha, and I. Guvenc, "Impact of 3D UWB antenna radiation pattern on air-to-ground drone connectivity," in *Proc. IEEE 88th Veh. Technol. Conf.*, 2018, pp. 1–5.

[13] A. Barrau and S. Bonnabel, "The invariant extended kalman filter as a stable observer," *IEEE Trans. Autom. Control*, vol. 62, no. 4, pp. 1797–1812, Apr. 2017.

[14] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning*. Cambridge, Massachusetts, USA: The MIT Press, 2006.

[15] B. Wang and T. Chen, "Gaussian process regression with multiple response variables," *Chemometrics Intell. Lab. Syst.*, vol. 142, pp. 159–165, 2015.

[16] J. Sol, J. Deray, and D. Atchuthan, "A Micro Lie Theory for State Estimation in Robotics," 2020, *arXiv:1812.01537v8*.

[17] D. K. Duvenaud, "Automatic model construction with gaussian processes," Ph.D. dissertation, University of Cambridge, Cambridge, England, United Kingdom, Jun. 2014.

[18] S. M. Persson and I. Sharf, "Invariant momentum-tracking kalman filter for attitude estimation," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2012, pp. 592–598.

[19] K. Lenac, J. Ćesić, I. Marković, and I. Petrović, "Exactly sparse delayed state filter on lie groups for long-term pose graph SLAM," *Int. J. Robot. Res.*, vol. 37, no. 6, pp. 585–610, 2018.