Recursive Gaussian Processes and Fingerprinting for Indoor Navigation

Tales Imbiriba, Peng Wu, Gerald LaMountain, Deniz Erdoğmuş, and Pau Closas
Department of Electrical and Computer Engineering
Northeastern University
Boston, MA

***Abstract*—This paper investigates the use of Gaussian Processes (GP) for RSS fingerprinting-based navigation. In particular, a recursive GP scheme to adapt the fingerprints as they evolve over time is discussed which accounts for the uncertainty of position labels associated to new RSS measurements. The marginalization over the uncertainty of position labels is here implemented numerically through cubature rules, which is seen from computer simulations to enhance field estimation performance and, ultimately, positioning results.**

***Index Terms*—Indoor navigation, fingerprinting, Gaussian Processes, geospatial sensing, uncertainty propagation.**

I. Introduction

The advance of telecommunication systems leads to new paradigms where technology becomes increasingly interconnected. This phenomenon associated with the necessity for seamless solutions in location-based services in indoor environments resulted in a steady increase in academic and commercial interest in indoor navigation systems capable of localize and tracking devices in real-time. Indoor navigation offers a number of difficult challenges which include the heterogeneity of technologies, resulting in the lack of a legacy solution, and the impossibility of using the widely diffused Global Navigation Satellite System (GNSS) technology, one of the most accurate sources of position information when it is available, which is often infeasible in indoor or obstructed environments [1]. Instead, alternative systems have to be adopted. One approach to positioning and tracking in such environments is fingerprinting, also referred to as mapping or scene analysis.

Fingerprinting consists in building a database containing a collection of measured features at designated reference locations and applying regression techniques the learn the underlying map between locations and measurements. The position in feature space associated through the database with a particular physical reference location is referred to as the "fingerprint" of the environment at that location, and the assumption that these feature vectors are relatively unique forms the basis for the fingerprinting technique. The fingerprinting procedure typically operates in two stages [1]: an offline stage in which the environment is surveyed at known locations and the results are recorded into a database, and an online stage in which navigation is performed by matching new measurements with the content of the database. Once a match is made, position may be inferred based on the reference positions associated with those database measurements.

Fingerprinting offers several important advantages as an indoor positioning technology. One important advantage is that it does not require a specific measurement model: since indoor environments are often segmented and highly non homogeneous in composition, useful observables such as sound and electromagnetic radiation often exhibit highly nonuniform or nonlinear diffuse behaviors which can not be easily or accurately modeled even without advanced and thorough knowledge of the contents of the environment. Fingerprinting replaces the need for an accurate measurement model with a need for an offline training stage, a requirement which may be more realistic in certain contexts. Other advantages are that they are strictly passive and can often make use of existing features of the environment such as installed WiFi networking. For this reason among others, WiFi received signal strength (RSS) fingerprinting is now widely used in indoor positioning and navigation problems [2]. Another remarkable feature of fingerprinting methods is that they do not require any knowledge regarding the location of the transmitting nodes, which is particularly relevant in other schemes such as those which are geometric-based [1]. On the other hand, a big limiting factors in generating fingerprinting databases are related to cost and time constraints which makes this strategy prohibitive for complex and dynamical environments.

One possible strategy to circumvent this limitation consists in collecting RSS measurements from interconnected devices transiting in the region of interest, a strategy adopted in *crowdsourcing* applications [3]. For instance, tracking algorithms are used to provide position estimations from time-of-arrival (TOA) [4] or RSS observations [5]. Thus, by exploiting tracking mechanisms embedded in Navigation systems, which often provide a full distribution characterization for position estimates [6], pairs of RSS values and their corresponding locations can be easily obtained. Although simple and cheap, adopting such approach comes at the expense of knowing the exact locations where measurements actually took place, thus, introducing input uncertainty. In this work, we consider an hybrid scenario where an initial Fingerprinting data is available while new measurements taken at unknown locations are constantly streamed accompanied by the distribution of the corresponding location estimate.

This work has been partially supported by the NSF under Awards CNS-1815349 and ECCS-1845833.

978-1-7281-0244-3/20/$31.00 ©2020 IEEE

One of the core building blocks of fingerprinting-based navigation systems is the mathematical model that interconnects locations and RSS measurements. The literature presents a variety of models which include the physic-based path-loss model [1], parametric models such as Neural Networks [6]–[8], and non-parametric (or semi-parametric) models such as kernel-based methods [9], [10]. One advantage of the later methods is the lack of assumption about the environment or the underlying function modeling RSS values as a function of spatial localization. In this work, we focus in kernel-based strategies. More specifically, we adopt Gaussian Process (GP) models since they cope with the Bayesian framework often adopted in Navigation systems.

GP regression methods consist of defining stochastic models for functions and performing inference in functional spaces [11]. These methods have been shown to be useful in a wide variety of fields and tasks including regression and classification [11], detection [12], unmixing [13], and Bayesian optimization [14], to name but a few. The basic idea behind GP regression consists of modeling a function for which very little information is available other than some smoothness assumption. The GP inference is often performed considering a training set comprised of paired known noiseless inputs and measured noisy outputs. In the context of indoor navigation, GP-based approaches have been employed under both traditional fingerprinting and noisy locations settings, see, [5], [10], [15], and references there in. When uncertain locations are considered, inference is performed by marginalizing the uncertain locations [10], [15], where, due to the nonlinear nature of GP models, Gaussian approximations and numerical integration methods are used.

A major drawback of GP is the high complexity associated with the inverse and determinant calculations of the kernel matrix. This limitation makes the employment of GP on its direct form impracticable as the number of available data grows. The literature presents different strategies to address this issue which can be crudely divided into local or global inference approximations [16]. Global approximations aim at reducing the complexity by distillation of the kernel matrix which can be achieved by selecting a subset of the training data (inducing points) [17], removing low-correlated entries (truncating), or considering Nyström approximation inspired strategies [11], [18]. Such strategies result in smaller, sparse and low-rank kernel matrices respectively. Local strategies consists in providing inference over local samples subsets of training data which can be combined using a global fusion strategy. The most widely studied approach involves mixture or product of local experts [16], [19], [20] providing smooth predictions and valid uncertainty measures [16]. In most strategies discussed above, accessibility to the whole training data is assumed and learning can be performed offline or in batch mode. When data arrives sequentially, online low-cost strategies should be considered. In [21], an online sparse GP regression strategy was presented. It promotes recursive updates of the posterior GP moments for a finite set of basis vectors, the so called *active set*. The methodology is based on Bayesian state estimation techniques and achieved by marginalizing the contribution provided by the newly incoming data.

In this contribution, we address the recursive active set GP strategy in the context of RSS field learning for indoor navigation accounting for location uncertainty. Specifically, we modify standard recursive GP based on active set to account for location errors. We consider a Bayesian approach where marginalization with respect to the uncertain locations is incorporated into the recursive GP algorithm. Due to the intrinsic nonlinearity of the problem and of the Gaussian process regression approach adopted, we consider Gaussian approximations of marginalized distributions implemented by numerical integration strategies such as cubature rules [22]. Finally, the marginalized distributions are incorporated in the recursive GP algorithm which is then tested in different related scenarios where the fingerprints, used for positioning purposes, change over time.

This paper is organized as follows. Section II presents standard GP regression while Section III presents the derivation of recursive GP (RGP). Section IV discusses the issue of position uncertainty and proposes a strategy to marginalize the uncertainty. Experiments to assess the performance for RSS field estimation and tracking performance with RGP are presented in Section V. Finally, concluding remarks are presented in Section VI.

II. Gaussian Process Regression

This section briefly presents the standard Gaussian process regression [11]. Given a set of N input-output pairs $\{\boldsymbol{x}_k, y_k\}_{k=1}^N$, $\boldsymbol{x} \in \mathcal{X} \subset \mathbb{R}^d$, $y \in \mathbb{R}$ related according to an arbitrary model such as

$$y_k = \psi(\boldsymbol{x}_k) + \eta_k \tag{1}$$

with $\eta \sim \mathcal{N}(0, \sigma_\eta^2)$, and $\psi \in \mathcal{H}$ considered to be a function of a reproducing kernel Hilbert space $\mathcal{H}$ defined over a compact set $\mathcal{X}$, GPs assume a Gaussian functional distribution as prior for the function $\psi|\boldsymbol{x}_k \sim \mathcal{N}(0, \kappa(\boldsymbol{x}_k, \boldsymbol{x}_k))$, where κ is a kernel function such that $\kappa(\cdot, \boldsymbol{x}) \in \mathcal{H}$. For a set of input points $\boldsymbol{X} = [\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N]$ the prior distribution for ψ becomes $\boldsymbol{\psi}|\boldsymbol{X} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{K})$, where $\boldsymbol{K} \in \mathbb{R}^{N \times N}$ is the Gram matrix with entries $[\boldsymbol{K}]_{ij} = \kappa(\boldsymbol{x}_i, \boldsymbol{x}_j)$. For a given set of measurements $\boldsymbol{y} = [y_1, \ldots, y_N]^\top$ associated with the positions $\boldsymbol{X}$, the prior distribution becomes

$$\boldsymbol{y} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{K} + \sigma_\eta^2 \boldsymbol{I}). \tag{2}$$

The predictive distribution allows one to "predict" the value of the function $\psi_\star$ for a new input value $\boldsymbol{x}_\star$. Thus, we have $\psi_\star|\boldsymbol{x}_\star \sim \mathcal{N}(0, \kappa_{\star\star})$, where $\kappa_{\star\star} \triangleq \kappa(\boldsymbol{x}_\star, \boldsymbol{x}_\star)$. Since $\boldsymbol{y}$ and $\psi_\star$ are jointly Gaussian their joint PDF is given by

$$\begin{bmatrix} \boldsymbol{y} \\ \psi_\star \end{bmatrix} \sim \mathcal{N}\left(\boldsymbol{0}, \begin{bmatrix} \boldsymbol{K} + \sigma_\eta^2 \boldsymbol{I} & \boldsymbol{\kappa}_\star \\ \boldsymbol{\kappa}_\star^\top & \kappa_{\star\star} \end{bmatrix}\right), \tag{3}$$

where $\boldsymbol{\kappa}_\star \triangleq [\kappa(\boldsymbol{x}_1, \boldsymbol{x}_\star), \ldots, \kappa(\boldsymbol{x}_N, \boldsymbol{x}_\star)]^\top$. Finally the predictive distribution can be obtained by conditioning $\psi_\star$ over the observation and the its respective positions as

$$\psi_\star | \boldsymbol{y}, \boldsymbol{X}, \boldsymbol{x}_\star \sim \mathcal{N}(\mu_{\psi|\boldsymbol{y},\boldsymbol{X},\boldsymbol{x}_\star}, s^2_{\psi|\boldsymbol{y},\boldsymbol{X},\boldsymbol{x}_\star}) \tag{4}$$

with $\mu_{\psi|\boldsymbol{y},\boldsymbol{X},\boldsymbol{x}_\star} = \boldsymbol{\kappa}_\star^\top \left(\boldsymbol{K} + \sigma_\eta^2 \boldsymbol{I}\right)^{-1} \boldsymbol{y}$, and $s^2_{\psi|\boldsymbol{y},\boldsymbol{X},\boldsymbol{x}_\star} = \kappa_{\star\star} - \boldsymbol{\kappa}_\star^\top \left(\boldsymbol{K} + \sigma_\eta^2 \boldsymbol{I}\right)^{-1} \boldsymbol{\kappa}_\star$. Using the GP distribution above, the distribution of $p(y_\star)$ can be obtained by replacing (4) in (1)

$$\hat{p}(y_\star) = \mathcal{N}(\mu_{\psi|\boldsymbol{y},\boldsymbol{X},\boldsymbol{x}_\star}, s^2_{\psi|\boldsymbol{y},\boldsymbol{X},\boldsymbol{x}_\star} + \sigma_\eta^2). \tag{5}$$

The Bayesian framework also provides strategies to estimate free parameters, such as the kernel parameters $\boldsymbol{\theta}$ and the noise power σ_η^2. The classical approach [11] aims at maximizing the marginal likelihood $p(\boldsymbol{y}|\boldsymbol{X}, \sigma_\eta^2, \boldsymbol{\theta})$ with respect to $(\sigma_\eta^2, \boldsymbol{\theta})$.

If we consider a prior with mean $m(x)$ then the GP posterior mean and variance becomes

$$\mu_{\psi_\star} = m(\boldsymbol{x}_\star) + \boldsymbol{\kappa}_\star^\top \left(\boldsymbol{K} + \sigma_\eta^2 \boldsymbol{I}\right)^{-1} \boldsymbol{y} \tag{6}$$

$$s^2_{\psi_\star} = \kappa_{\star\star} - \boldsymbol{\kappa}_\star^\top \left(\boldsymbol{K} + \sigma_\eta^2 \boldsymbol{I}\right)^{-1} \boldsymbol{\kappa}_\star. \tag{7}$$

III. Recursive Gaussian Process

Recursive Gaussian processes [21] aims at recursively learn the underlying function ψ by updating the GP model, anchored at a set of basis points (active set) $\boldsymbol{X}$, $\boldsymbol{\psi} \triangleq \psi(\boldsymbol{X})$, by exploiting the information provided by new observations $\boldsymbol{y}_t$ taken at inputs $\boldsymbol{X}_t$, with $t = 0, 1, \ldots$ being a time index, but without necessarily increasing the number of basis points $\boldsymbol{X}$, and, thus, keeping the GP's computational complexity limited. This objective is achieved by marginalizaing the predictions $\boldsymbol{\psi}_t \triangleq \psi(\boldsymbol{X}_t)$ obtained for the new set $\boldsymbol{X}_t$.

Thus, assuming an initial prior for the GP, such that $p(\boldsymbol{\psi}|\boldsymbol{X}) = \mathcal{N}_{\boldsymbol{\psi}}(\boldsymbol{\mu}_0^\psi, \boldsymbol{\Sigma}_0^\psi)$, with $\boldsymbol{\mu}_0^\psi \triangleq m(\boldsymbol{X})$ and $\boldsymbol{\Sigma}_0^\psi \triangleq \boldsymbol{K}$, the goal is to calculate the posterior distribution $p(\boldsymbol{\psi}|\boldsymbol{y}_{1:t})$ by updating the distribution of $\boldsymbol{\psi}$ from the previous step $t-1$

$$p(\boldsymbol{\psi}|\boldsymbol{y}_{1:t-1}) = \mathcal{N}_{\boldsymbol{\psi}}\left(\boldsymbol{\mu}_{t-1}^\psi, \boldsymbol{\Sigma}_{t-1}^\psi\right) \tag{8}$$

by exploiting the new observation $\boldsymbol{y}_t$. $p(\boldsymbol{\psi}|\boldsymbol{y}_{1:t})$ can be obtained after marginalization over $\boldsymbol{\psi}_t$:

$$\begin{aligned} p(\boldsymbol{\psi}|\boldsymbol{y}_{1:t}) &= \int p(\boldsymbol{\psi}, \boldsymbol{\psi}_t | \boldsymbol{y}_{1:t}) d\boldsymbol{\psi}_t \\ &= \int \frac{p(\boldsymbol{y}_t | \cancel{\boldsymbol{\psi}}, \boldsymbol{\psi}_t, \cancel{\boldsymbol{y}_{1:t-1}}) p(\boldsymbol{\psi}, \boldsymbol{\psi}_t | \boldsymbol{y}_{1:t-1})}{p(\boldsymbol{y}_{1:t-1})} d\boldsymbol{\psi}_t \\ &= \int \frac{p(\boldsymbol{y}_t | \boldsymbol{\psi}_t) p(\boldsymbol{\psi}_t | \boldsymbol{\psi}, \cancel{\boldsymbol{y}_{1:t-1}}) p(\boldsymbol{\psi} | \boldsymbol{y}_{1:t-1})}{p(\boldsymbol{y}_{1:t-1})} d\boldsymbol{\psi}_t \\ &= \int c_t p(\boldsymbol{y}_t | \boldsymbol{\psi}_t) p(\boldsymbol{\psi}_t | \boldsymbol{\psi}) p(\boldsymbol{\psi} | \boldsymbol{y}_{1:t-1}) d\boldsymbol{\psi}_t \end{aligned} \tag{9}$$

which requires computation of the distributions in the integral.

The distribution $p(\boldsymbol{\psi}_t|\boldsymbol{\psi})$ can be obtained from the following joint distribution

$$\begin{bmatrix} \boldsymbol{\psi} \\ \boldsymbol{\psi}_t \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \boldsymbol{m} \\ \boldsymbol{m}_t \end{bmatrix}, \begin{bmatrix} \boldsymbol{K} & \boldsymbol{K}_t \\ \boldsymbol{K}_t^\top & \boldsymbol{K}_{t,t} \end{bmatrix} \right), \tag{10}$$

with $\boldsymbol{m} \triangleq m(\boldsymbol{X})$, $\boldsymbol{m}_t \triangleq m(\boldsymbol{X})_t$, $\boldsymbol{K}_t \triangleq \kappa(\boldsymbol{X}, \boldsymbol{X}_t)$ and $\boldsymbol{K}_{tt} \triangleq \kappa(\boldsymbol{X}_t, \boldsymbol{X}_t)$. Thus,

$$p(\boldsymbol{\psi}_t|\boldsymbol{\psi}) = \mathcal{N}\left(\boldsymbol{\mu}^{\boldsymbol{\psi}_t|\boldsymbol{\psi}}, \boldsymbol{\Sigma}^{\boldsymbol{\psi}_t|\boldsymbol{\psi}}\right) \tag{11}$$

with

$$\begin{aligned} \boldsymbol{\mu}^{\boldsymbol{\psi}_t|\boldsymbol{\psi}} &= \boldsymbol{m}_t + \boldsymbol{K}_t^\top \boldsymbol{K}^{-1} (\boldsymbol{\psi} - \boldsymbol{m}) \\ &= \boldsymbol{m}_t + \boldsymbol{J}_t (\boldsymbol{\psi} - \boldsymbol{m}) \end{aligned} \tag{12}$$

and

$$\boldsymbol{\Sigma}^{\boldsymbol{\psi}_t|\boldsymbol{\psi}} = \boldsymbol{K}_{tt} - \boldsymbol{J}_t \boldsymbol{K}_t \tag{13}$$

where

$$\boldsymbol{J}_t = \boldsymbol{K}_t^\top \boldsymbol{K}^{-1}. \tag{14}$$

The joint distribution $p(\boldsymbol{\psi}, \boldsymbol{\psi}_t|\boldsymbol{y}_{1:t-1}) = p(\boldsymbol{\psi}_t|\boldsymbol{\psi}) p(\boldsymbol{\psi}|\boldsymbol{y}_{1:t-1})$ can be obtained from (8) and (11) by completing the square[1] in (15) where the quadratic part on $\boldsymbol{\psi}_t$ and $\boldsymbol{\psi}$ is given in (16).

Using the identity in [23] we have

$$\boldsymbol{\Sigma}_{\boldsymbol{\psi},\boldsymbol{\psi}_t} = \left[\begin{array}{c|c} \boldsymbol{\Sigma}_{t-1}^{\boldsymbol{\psi}} & \boldsymbol{\Sigma}_{t-1}^{\boldsymbol{\psi}} \boldsymbol{J}_t^\top \\ \hline \boldsymbol{J}_t \boldsymbol{\Sigma}_{t-1}^{\boldsymbol{\psi}} & \boldsymbol{\Sigma}^{\boldsymbol{\psi}_t|\boldsymbol{\psi}} + \boldsymbol{J}_t \boldsymbol{\Sigma}_{t-1}^{\boldsymbol{\psi}} \boldsymbol{J}_t^\top \end{array}\right]. \tag{17}$$

The crossed term is given by

$$\begin{aligned} C &= \boldsymbol{z} \boldsymbol{\Sigma}^{-1}_{\boldsymbol{\psi},\boldsymbol{\psi}_t} \boldsymbol{\mu}_z \\ &= \begin{bmatrix} \boldsymbol{\psi} \\ \hline \boldsymbol{\psi}_t \end{bmatrix}^\top \begin{bmatrix} \left(\boldsymbol{\Sigma}_{t-1}^{\boldsymbol{\psi}}\right)^{-1} \boldsymbol{\mu}_{t-1}^{\boldsymbol{\psi}} - \boldsymbol{J}_t^\top \left(\boldsymbol{\Sigma}^{\boldsymbol{\psi}_t|\boldsymbol{\psi}}\right)^{-1} (\boldsymbol{m}_t - \boldsymbol{J}_t \boldsymbol{m}) \\ \hline \left(\boldsymbol{\Sigma}^{\boldsymbol{\psi}_t|\boldsymbol{\psi}}\right)^{-1} (\boldsymbol{m}_t - \boldsymbol{J}_t \boldsymbol{m}) \end{bmatrix} \\ &= \begin{bmatrix} \boldsymbol{\psi} \\ \hline \boldsymbol{\psi}_t \end{bmatrix}^\top \left[\begin{array}{c|c} \left(\boldsymbol{\Sigma}_{t-1}^{\boldsymbol{\psi}}\right)^{-1} + \boldsymbol{J}_t^\top \left(\boldsymbol{\Sigma}^{\boldsymbol{\psi}_t|\boldsymbol{\psi}}\right)^{-1} \boldsymbol{J}_t & -\boldsymbol{J}_t^\top \left(\boldsymbol{\Sigma}^{\boldsymbol{\psi}_t|\boldsymbol{\psi}}\right)^{-1} \\ \hline -\left(\boldsymbol{\Sigma}^{\boldsymbol{\psi}_t|\boldsymbol{\psi}}\right)^{-1} \boldsymbol{J}_t & \left(\boldsymbol{\Sigma}^{\boldsymbol{\psi}_t|\boldsymbol{\psi}}\right)^{-1} \end{array}\right] \\ &\quad \times \begin{bmatrix} \boldsymbol{\mu}^{\boldsymbol{\psi}} \\ \hline \boldsymbol{\mu}^{\boldsymbol{\psi}_t} \end{bmatrix} \end{aligned} \tag{18}$$

from where we conclude that

$$\boldsymbol{\mu}_z = \begin{bmatrix} \boldsymbol{\mu}_{t-1}^{\boldsymbol{\psi}} \\ \boldsymbol{m}_t + \boldsymbol{J}_t \left(\boldsymbol{\mu}_{t-1}^{\boldsymbol{\psi}} - \boldsymbol{m}\right) \end{bmatrix} \tag{19}$$

Thus, the vector $\boldsymbol{z} = [\boldsymbol{\psi}, \boldsymbol{\psi}_t]^\top$ is Gaussian distributed according to

$$\begin{aligned} \begin{bmatrix} \boldsymbol{\psi} \\ \hline \boldsymbol{\psi}_t \end{bmatrix} \sim \mathcal{N}\Bigg(&\begin{bmatrix} \boldsymbol{\mu}_{t-1}^{\boldsymbol{\psi}} \\ \hline \boldsymbol{m}_t + \boldsymbol{J}_t \left(\boldsymbol{\mu}_{t-1}^{\boldsymbol{\psi}} - \boldsymbol{m}\right) \end{bmatrix}, \\ &\left[\begin{array}{c|c} \boldsymbol{\Sigma}_{t-1}^{\boldsymbol{\psi}} & \boldsymbol{\Sigma}_{t-1}^{\boldsymbol{\psi}} \boldsymbol{J}_t^\top \\ \hline \boldsymbol{J}_t \boldsymbol{\Sigma}_{t-1}^{\boldsymbol{\psi}} & \boldsymbol{\Sigma}^{\boldsymbol{\psi}_t|\boldsymbol{\psi}} + \boldsymbol{J}_t \boldsymbol{\Sigma}_{t-1}^{\boldsymbol{\psi}} \boldsymbol{J}_t^\top \end{array}\right] \Bigg) \end{aligned} \tag{20}$$

and the marginal of $\boldsymbol{\psi}_t$

$$p(\boldsymbol{\psi}_t|\boldsymbol{y}_{1:t-1}) = \mathcal{N}\left(\boldsymbol{\mu}^{\boldsymbol{\psi}_t}, \boldsymbol{\Sigma}^{\boldsymbol{\psi}_t}\right) \tag{21}$$

with moments

$$\boldsymbol{\mu}^{\boldsymbol{\psi}_t} = \boldsymbol{m}_t + \boldsymbol{J}_t \left(\boldsymbol{\mu}_{t-1}^{\boldsymbol{\psi}} - \boldsymbol{m}\right) \tag{22}$$

$$\boldsymbol{\Sigma}^{\boldsymbol{\psi}_t} = \boldsymbol{\Sigma}^{\boldsymbol{\psi}_t|\boldsymbol{\psi}} + \boldsymbol{J}_t \boldsymbol{\Sigma}_{t-1}^{\boldsymbol{\psi}} \boldsymbol{J}_t^\top. \tag{23}$$

[1] Noting that $-\frac{1}{2}(\boldsymbol{z}-\boldsymbol{\mu}_z)^\top \boldsymbol{\Sigma}_z^{-1}(\boldsymbol{z}-\boldsymbol{\mu}_z) = -\frac{1}{2}\boldsymbol{z}^\top \boldsymbol{\Sigma}_z^{-1}\boldsymbol{z} + \boldsymbol{z}^\top \boldsymbol{\Sigma}_z^{-1}\boldsymbol{\mu}_z + constant$, one needs only to identify the squared and crossed terms to obtain the covariance and mean of the resulting distribution.

$$S = -\frac{1}{2}(\boldsymbol{\psi}_t - \boldsymbol{\mu}^{\boldsymbol{\psi}_t|\boldsymbol{\psi}})^\top [\boldsymbol{\Sigma}^{\boldsymbol{\psi}_t|\boldsymbol{\psi}}]^{-1}(\boldsymbol{\psi}_t - \boldsymbol{\mu}^{\boldsymbol{\psi}_t|\boldsymbol{\psi}}) - \frac{1}{2}(\boldsymbol{\psi} - \boldsymbol{\mu}^{\boldsymbol{\psi}}_{t-1})^\top [\boldsymbol{\Sigma}^{\boldsymbol{\psi}}_{t-1}]^{-1}(\boldsymbol{\psi} - \boldsymbol{\mu}^{\boldsymbol{\psi}}_{t-1})$$
$$= -\frac{1}{2}(\boldsymbol{\psi}_t - \boldsymbol{m}_t - \boldsymbol{J}_t(\boldsymbol{\psi} - \boldsymbol{m}))^\top [\boldsymbol{\Sigma}^{\boldsymbol{\psi}_t|\boldsymbol{\psi}}]^{-1}(\boldsymbol{\psi}_t - \boldsymbol{m}_t - \boldsymbol{J}_t(\boldsymbol{\psi} - \boldsymbol{m})) - \frac{1}{2}(\boldsymbol{\psi} - \boldsymbol{\mu}^{\boldsymbol{\psi}}_{t-1})^\top [\boldsymbol{\Sigma}^{\boldsymbol{\psi}}_{t-1}]^{-1}(\boldsymbol{\psi} - \boldsymbol{\mu}^{\boldsymbol{\psi}}_{t-1}) \quad (15)$$

$$Q = -\frac{1}{2}\boldsymbol{z}^\top \boldsymbol{\Sigma}^{-1}_{\boldsymbol{\psi},\boldsymbol{\psi}_t}\boldsymbol{z} = -\frac{1}{2}\begin{bmatrix}\boldsymbol{\psi}\\ \boldsymbol{\psi}_t\end{bmatrix}^\top \left[\begin{array}{c|c} (\boldsymbol{\Sigma}^{\boldsymbol{\psi}}_{t-1})^{-1} + \boldsymbol{J}_t^\top (\boldsymbol{\Sigma}^{\boldsymbol{\psi}_t|\boldsymbol{\psi}})^{-1}\boldsymbol{J}_t & -\boldsymbol{J}_t^\top (\boldsymbol{\Sigma}^{\boldsymbol{\psi}_t|\boldsymbol{\psi}})^{-1} \\ \hline -(\boldsymbol{\Sigma}^{\boldsymbol{\psi}_t|\boldsymbol{\psi}})^{-1}\boldsymbol{J}_t & (\boldsymbol{\Sigma}^{\boldsymbol{\psi}_t|\boldsymbol{\psi}})^{-1}\end{array}\right]\begin{bmatrix}\boldsymbol{\psi}\\ \boldsymbol{\psi}_t\end{bmatrix} \quad (16)$$

The update part is obtained by performing the marginalization in (9). Note that (9) can be rewritten as

$$\begin{aligned} p(\boldsymbol{\psi}|\boldsymbol{y}_{1:t}) &= \int c_t p(\boldsymbol{y}_t|,\boldsymbol{\psi}_t)p(\boldsymbol{\psi}_t,\boldsymbol{\psi}|\boldsymbol{y}_{1:t-1})d\boldsymbol{\psi}_t \\ &= \int c_t p(\boldsymbol{y}_t|,\boldsymbol{\psi}_t)p(\boldsymbol{\psi}|\boldsymbol{\psi}_t,\boldsymbol{y}_{1:t-1})p(\boldsymbol{\psi}_t|\boldsymbol{y}_{1:t-1})d\boldsymbol{\psi}_t \\ &= \int \underbrace{c_t p(\boldsymbol{y}_t|,\boldsymbol{\psi}_t)p(\boldsymbol{\psi}_t|\boldsymbol{y}_{1:t-1})}_{\text{Kalman filter update}} p(\boldsymbol{\psi}|\boldsymbol{\psi}_t,\boldsymbol{y}_{1:t-1})d\boldsymbol{\psi}_t \\ &= \int p(\boldsymbol{\psi}_t|\boldsymbol{y}_{1:t})p(\boldsymbol{\psi}|\boldsymbol{\psi}_t,\boldsymbol{y}_{1:t-1})d\boldsymbol{\psi}_t \end{aligned} \quad (24)$$

where, since $p(\boldsymbol{y}_t|,\boldsymbol{\psi}_t) = \mathcal{N}(\boldsymbol{\psi}_t, \sigma_\eta^2\boldsymbol{I})$ and $p(\boldsymbol{\psi}_t|\boldsymbol{y}_{1:t-1}) = \mathcal{N}(\boldsymbol{\mu}^{\boldsymbol{\psi}_t}, \boldsymbol{\Sigma}^{\boldsymbol{\psi}_t})$ are both Gaussian distributions, the term $c_t p(\boldsymbol{y}_t|,\boldsymbol{\psi}_t)p(\boldsymbol{\psi}_t|\boldsymbol{y}_{1:t-1}) = p(\boldsymbol{\psi}_t|\boldsymbol{y}_{1:t})$ can be seen as a standard Kalman filter update (under the model $\boldsymbol{y}_t = \boldsymbol{\psi}_t + \boldsymbol{n}_t$) whose posterior $p(\boldsymbol{\psi}_t|\boldsymbol{y}_{1:t})$ is also Gaussian with moments given by

$$\boldsymbol{\mu}^{\boldsymbol{\psi}_t}_e = \boldsymbol{\mu}^{\boldsymbol{\psi}_t} + \boldsymbol{G}_t(\boldsymbol{y}_t - \boldsymbol{\mu}^{\boldsymbol{\psi}_t}) \quad (25)$$
$$\boldsymbol{\Sigma}^{\boldsymbol{\psi}_t}_e = \boldsymbol{\Sigma}^{\boldsymbol{\psi}_t} - \boldsymbol{G}_t\boldsymbol{\Sigma}^{\boldsymbol{\psi}_t} \quad (26)$$

where $\boldsymbol{G}_t \triangleq \boldsymbol{\Sigma}^{\boldsymbol{\psi}_t}(\boldsymbol{\Sigma}^{\boldsymbol{\psi}_t} + \sigma_\eta^2\boldsymbol{I})^{-1}$ is the Kalman gain.

Finally, $p(\boldsymbol{\psi}|\boldsymbol{\psi}_t, \boldsymbol{y}_{1:t-1})$ can be obtained from the joint distribution in (20) by applying the identity in [23, pg. 87]. Thus, $p(\boldsymbol{\psi}|\boldsymbol{\psi}_t, \boldsymbol{y}_{1:t-1}) = \mathcal{N}(\boldsymbol{\mu}^{\boldsymbol{\psi}|\boldsymbol{\psi}_t}, \boldsymbol{\Sigma}^{\boldsymbol{\psi}|\boldsymbol{\psi}_t})$ with moments

$$\boldsymbol{\mu}^{\boldsymbol{\psi}|\boldsymbol{\psi}_t} = \boldsymbol{\mu}^{\boldsymbol{\psi}}_{t-1} + \boldsymbol{L}_t(\boldsymbol{\psi}_t - \boldsymbol{\mu}^{\boldsymbol{\psi}_t}) \quad (27)$$
$$\boldsymbol{\Sigma}^{\boldsymbol{\psi}|\boldsymbol{\psi}_t} = \boldsymbol{\Sigma}^{\boldsymbol{\psi}}_{t-1} - \boldsymbol{L}_t\boldsymbol{J}_t\boldsymbol{\Sigma}^{\boldsymbol{\psi}}_{t-1} \quad (28)$$

and $\boldsymbol{L}_t \triangleq \boldsymbol{\Sigma}^{\boldsymbol{\psi}}_{t-1}\boldsymbol{J}_t^\top \left(\boldsymbol{\Sigma}^{\boldsymbol{\psi}_t}\right)^{-1}$.

With these results the product in the right side of (24) direct leads to the joint Gaussian distribution of $\boldsymbol{\psi}, \boldsymbol{\psi}_t|\boldsymbol{y}_{1:t}$, whose quadratic form is

$$Q_{\boldsymbol{\psi},\boldsymbol{\psi}_t} = -\frac{1}{2}\boldsymbol{z}^\top \boldsymbol{\Sigma}^{-1}_{\boldsymbol{\psi},\boldsymbol{\psi}_t}\boldsymbol{z} \quad (29)$$

with $\boldsymbol{z} = [\boldsymbol{\psi}, \boldsymbol{\psi}_t]^\top$ and

$$\boldsymbol{\Sigma}^{-1}_{\boldsymbol{\psi},\boldsymbol{\psi}_t} = \left[\begin{array}{c|c} (\boldsymbol{\Sigma}^{\boldsymbol{\psi}|\boldsymbol{\psi}_t})^{-1} & -(\boldsymbol{\Sigma}^{\boldsymbol{\psi}_t|\boldsymbol{\psi}})^{-1}\boldsymbol{L}_t \\ \hline -\boldsymbol{L}_t^\top(\boldsymbol{\Sigma}^{\boldsymbol{\psi}_t|\boldsymbol{\psi}})^{-1} & (\boldsymbol{\Sigma}^{\boldsymbol{\psi}_t}_e)^{-1} + \boldsymbol{L}_t^\top(\boldsymbol{\Sigma}^{\boldsymbol{\psi}_t|\boldsymbol{\psi}})^{-1}\boldsymbol{L}_t\end{array}\right] \quad (30)$$

from where we conclude that

$$\boldsymbol{\Sigma}_{\boldsymbol{\psi},\boldsymbol{\psi}_t} = \left[\begin{array}{c|c} \boldsymbol{\Sigma}^{\boldsymbol{\psi}|\boldsymbol{\psi}_t} + \boldsymbol{L}_t\boldsymbol{\Sigma}^{\boldsymbol{\psi}_t}_e\boldsymbol{L}_t^\top & \boldsymbol{L}_t\boldsymbol{\Sigma}^{\boldsymbol{\psi}_t}_e \\ \hline \boldsymbol{\Sigma}^{\boldsymbol{\psi}_t}_e\boldsymbol{L}_t^\top & \boldsymbol{\Sigma}^{\boldsymbol{\psi}_t}_e\end{array}\right] \quad (31)$$

and

$$\begin{aligned} \boldsymbol{\Sigma}^{\boldsymbol{\psi}}_t &= \boldsymbol{\Sigma}^{\boldsymbol{\psi}|\boldsymbol{\psi}_t} + \boldsymbol{L}_t\boldsymbol{\Sigma}^{\boldsymbol{\psi}_t}_e\boldsymbol{L}_t^\top \\ &= \boldsymbol{\Sigma}^{\boldsymbol{\psi}}_{t-1} - \boldsymbol{L}_t\boldsymbol{J}_t\boldsymbol{\Sigma}^{\boldsymbol{\psi}}_{t-1} + \boldsymbol{L}_t\boldsymbol{\Sigma}^{\boldsymbol{\psi}_t}_e\boldsymbol{L}_t^\top \\ &= \boldsymbol{\Sigma}^{\boldsymbol{\psi}}_{t-1} - \boldsymbol{L}_t\boldsymbol{J}_t\boldsymbol{\Sigma}^{\boldsymbol{\psi}}_{t-1} \\ &\quad + \boldsymbol{L}_t\left(\boldsymbol{\Sigma}^{\boldsymbol{\psi}_t} - \boldsymbol{G}_t\boldsymbol{\Sigma}^{\boldsymbol{\psi}_t}\right)\left(\boldsymbol{\Sigma}^{\boldsymbol{\psi}}_{t-1}\boldsymbol{J}_t^\top\left(\boldsymbol{\Sigma}^{\boldsymbol{\psi}_t}\right)^{-1}\right)^\top \\ &= \boldsymbol{\Sigma}^{\boldsymbol{\psi}}_{t-1} - \boldsymbol{L}_t\boldsymbol{G}_t\boldsymbol{J}_t\boldsymbol{\Sigma}^{\boldsymbol{\psi}}_{t-1} \\ &= \boldsymbol{\Sigma}^{\boldsymbol{\psi}}_{t-1} - \widetilde{\boldsymbol{G}}_t\boldsymbol{J}_t\boldsymbol{\Sigma}^{\boldsymbol{\psi}}_{t-1} \end{aligned} \quad (32)$$

Equivalently the mean becomes

$$\boldsymbol{\mu}^{\boldsymbol{\psi}}_t = \boldsymbol{\mu}^{\boldsymbol{\psi}}_{t-1} + \widetilde{\boldsymbol{G}}_t(\boldsymbol{y}_t - \boldsymbol{\mu}^{\boldsymbol{\psi}_t}). \quad (33)$$

with

$$\widetilde{\boldsymbol{G}}_t = \boldsymbol{L}_t\boldsymbol{G}_t. \quad (34)$$

Algorithm 1: Recursive GP Update

Input : $\boldsymbol{X}$, $\boldsymbol{X}_t$, $\mathcal{GP}_{t-1} \triangleq (\boldsymbol{\mu}^{\boldsymbol{\psi}}_{t-1}, \boldsymbol{\Sigma}^{\boldsymbol{\psi}}_{t-1})$
Output: $\boldsymbol{\mu}^{\boldsymbol{\psi}}_t$, $\boldsymbol{\Sigma}^{\boldsymbol{\psi}}_t$
% *Inference* ;
Compute $\boldsymbol{J}_t$ using (14);
Compute $\boldsymbol{\mu}^{\boldsymbol{\psi}_t}$ using (22);
Compute $\boldsymbol{\Sigma}^{\boldsymbol{\psi}_t}$ using (23);
% *Update* ;
Compute $\widetilde{\boldsymbol{G}}_t$ using (34);
Compute $\boldsymbol{\mu}^{\boldsymbol{\psi}}_t$ using (33);
Compute $\boldsymbol{\Sigma}^{\boldsymbol{\psi}}_t$ (32);
return $\boldsymbol{\mu}^{\boldsymbol{\psi}}_t$, $\boldsymbol{\Sigma}^{\boldsymbol{\psi}}_t$

IV. Accounting for position uncertainty

In most spatial-temporal applications the target device localization is unknown and, thus, must be estimated using some kind of positioning or tracking algorithm. However, positioning estimates provided by such algorithms contain errors which could be assessed by exploiting uncertainty measures often provided by such algorithms. Particularly, Bayesian filtering tracking algorithms provide a distribution characterization of the states over time. Here, we focus on the spatial RSS field estimation when new measures are obtained at uncertain locations, that is, new RSS vectors $\boldsymbol{y}_t$ are measured at some uncertain locations, which can be statistically characterized as $\boldsymbol{x}_t \sim p(\boldsymbol{x}_t) = \mathcal{N}(\bar{\boldsymbol{x}}_t, \boldsymbol{\Sigma}_{\boldsymbol{x}_t})$.

Following a Bayesian framework, to deal with the intrinsic uncertainty provided by the distribution $p(\boldsymbol{x}_t)$ one must resort to marginalization of the random variable $\boldsymbol{x}_t$. This will have a direct impact int the computation of the kernel matrices $\boldsymbol{K}_t$ and $\boldsymbol{K}_{tt}$ that depend on new locations $\boldsymbol{x}_t$.

Our objective is to compute $\boldsymbol{\mu}_t^{\psi}$ and $\boldsymbol{\Sigma}_t^{\psi}$ considering that $\boldsymbol{X}_t$ is a Gaussian random matrix. We propose then to consider the expected values with respect to $\boldsymbol{X}_t$ such as

$$\begin{aligned}\boldsymbol{\mu}_t^{\psi} &= \mathbb{E}_{\boldsymbol{X}_t}\left\{\mathbb{E}\{\boldsymbol{\psi}|\boldsymbol{y}_{1:t}, \boldsymbol{X}_t\}\right\} = \mathbb{E}_{\boldsymbol{X}_t}\left\{\boldsymbol{\mu}_t^{\psi}|\boldsymbol{X}_t\right\} \\ &= \mathbb{E}_{\boldsymbol{X}_t}\left\{\boldsymbol{\mu}_{t-1}^{\psi} + \widetilde{\boldsymbol{G}}_t(\boldsymbol{y}_t - \boldsymbol{\mu}^{\psi_t})\right\} \\ &= \boldsymbol{\mu}_{t-1}^{\psi} + \mathbb{E}_{\boldsymbol{X}_t}\left\{\boldsymbol{f}(\boldsymbol{X}_t)\right\} \end{aligned} \tag{35}$$

and

$$\begin{aligned}\boldsymbol{\Sigma}_t^{\psi} &= \mathrm{var}_{\boldsymbol{X}_t}\left\{\mathbb{E}\left\{\boldsymbol{\psi}|\boldsymbol{y}_{1:t}, \boldsymbol{X}_t\right\}\right\} + \mathbb{E}_{\boldsymbol{X}_t}\left\{\mathrm{var}\left\{\boldsymbol{\psi}|\boldsymbol{y}_{1:t}, \boldsymbol{X}_t\right\}\right\} \\ &= \mathrm{var}_{\boldsymbol{X}_t}\left\{\boldsymbol{\mu}_t^{\psi}|\boldsymbol{X}_t\right\} + \mathbb{E}_{\boldsymbol{X}_t}\left\{\boldsymbol{\Sigma}_t^{\psi}|\boldsymbol{X}_t\right\} \\ &= \mathbb{E}_{\boldsymbol{X}_t}\left\{\left(\boldsymbol{\mu}_t^{\psi}|\boldsymbol{X}_t\right)\left(\boldsymbol{\mu}_t^{\psi}|\boldsymbol{X}_t\right)^{\top}\right\} \\ &\quad - \left(\mathbb{E}_{\boldsymbol{X}_t}\left\{\boldsymbol{\mu}_t^{\psi}|\boldsymbol{X}_t\right\}\right)\left(\mathbb{E}_{\boldsymbol{X}_t}\left\{\boldsymbol{\mu}_t^{\psi}|\boldsymbol{X}_t\right\}\right)^{\top} \\ &\quad + \mathbb{E}_{\boldsymbol{X}_t}\left\{\boldsymbol{\Sigma}_{t-1}^{\psi} - \widetilde{\boldsymbol{G}}_t\boldsymbol{J}_t\boldsymbol{\Sigma}_{t-1}^{\psi}\right\} \\ &= \mathbb{E}_{\boldsymbol{X}_t}\left\{\boldsymbol{f}(\boldsymbol{X}_t)\boldsymbol{f}^{\top}(\boldsymbol{X}_t)\right\} - \mathbb{E}_{\boldsymbol{X}_t}\left\{\boldsymbol{f}(\boldsymbol{X}_t)\right\}\mathbb{E}_{\boldsymbol{X}_t}^{\top}\left\{\boldsymbol{f}(\boldsymbol{X}_t)\right\} \\ &\quad + \boldsymbol{\Sigma}_{t-1}^{\psi} - \mathbb{E}_{\boldsymbol{X}_t}\left\{\boldsymbol{g}(\boldsymbol{X}_t)\right\}\boldsymbol{\Sigma}_{t-1}^{\psi} \end{aligned} \tag{36}$$

with $\boldsymbol{f}(\boldsymbol{X}_t) = \widetilde{\boldsymbol{G}}_t(\boldsymbol{y}_t - \boldsymbol{\mu}^{\psi_t})$ and $\boldsymbol{g}(\boldsymbol{X}_t) = \widetilde{\boldsymbol{G}}_t\boldsymbol{J}_t$.

The expected values in (35) and (36) cannot be solved analytically due to the nonlinear nature of the functions. A Taylor expansion approach for linearization could be considered, which requires computation of the derivatives of those functions, which could be cumbersome and not accurate due to the non-linearity degree of those. In this article, given that the expectation is over the Gaussian random variable $\boldsymbol{X}_t$, we approximate those expectations using deterministic cubature rules [24]–[26] such as

$$\mathbb{E}_{\boldsymbol{X}_t}\left\{\boldsymbol{f}(\boldsymbol{X}_t)\right\} \approx \frac{1}{m}\sum_{j=1}^{m}\boldsymbol{f}(\boldsymbol{X}_t^{(j)}) \tag{37}$$

$$\mathbb{E}_{\boldsymbol{X}_t}\left\{\boldsymbol{g}(\boldsymbol{X}_t)\right\} \approx \frac{1}{m}\sum_{j=1}^{m}\boldsymbol{g}(\boldsymbol{X}_t^{(j)}) \tag{38}$$

$$\mathbb{E}_{\boldsymbol{X}_t}\left\{\boldsymbol{f}(\boldsymbol{X}_t)\boldsymbol{f}^{\top}(\boldsymbol{X}_t)\right\} \approx \frac{1}{m}\sum_{j=1}^{m}\boldsymbol{f}(\boldsymbol{X}_t^{(j)})\boldsymbol{f}^{\top}(\boldsymbol{X}_t^{(j)}) \tag{39}$$

with $\boldsymbol{X}_t^{(j)}$ being sigma-points sampled at optimal locations with respect to the distribution $p(\boldsymbol{X}_t)$.

V. Experiments

This section presents experiments with synthetic data. The data was created using the path-loss model [27] with a path-loss exponent of $\gamma = 3$. White Gaussian noise was added to all RSS values according to the model in (1) and modeling sensor measurement noise. In all simulations we considered

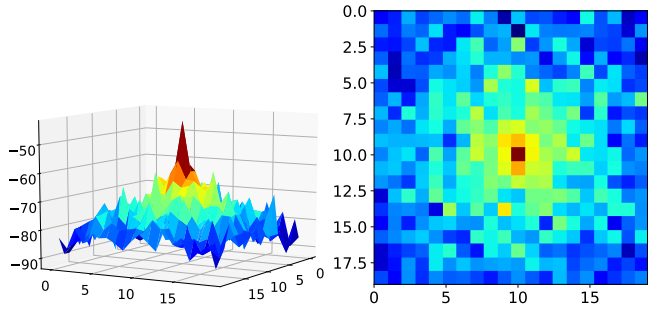

Fig. 1. Fingerprinting data for scenario 1.

$\sigma_\eta = 4$ which is a typical noise standard deviation for RSS modeling [1]. An initial RSS-fingerprinting dataset consisting of 400 location/RSS pairs distributed uniformly across the space was generated, which was used to train an initial GP characterization of the fingerpring.

As error metrics we considered the Root Mean Squared Error (RMSE) for position error and the normalized RMSE (NRMSE) for field error estimation. The RMSE is given by

$$\mathrm{RMSE}(\boldsymbol{x}_t) = \sqrt{\frac{1}{n}\sum_{r=1}^{n}\|\boldsymbol{x}_t^{(r)} - \hat{\boldsymbol{x}}_t^{(r)}\|^2} \tag{40}$$

where n is the number of time samples used to empirically approximate the expectation. The NRMSE for the field estimation is defined as

$$\mathrm{NRMSE}(\boldsymbol{\psi}_t) = \sqrt{\frac{1}{n}\sum_{t=1}^{n}\frac{\|\boldsymbol{\psi}_t - \hat{\boldsymbol{\psi}}_t\|^2}{\|\boldsymbol{\psi}_t\|^2}} \tag{41}$$

where $\boldsymbol{\psi}_t$ and $\hat{\boldsymbol{\psi}}_t$ are respectively the true and estimated field at time t.

A. Field estimation performance

To assess the field estimation performance and convergence of the discussed algorithms, we constructed a simulated scenario consisting in the initial fingerprinting set, a field change and new measurements y_t taking place at uncertain locations. That is, given a mean location $\boldsymbol{\mu}_{\boldsymbol{x}_t}$ and a covariance matrix $\boldsymbol{\Sigma}_{\boldsymbol{x}_t}$, the actual RSS measurement y_t takes place at a location $\boldsymbol{x}_t \sim \mathcal{N}(\boldsymbol{\mu}_{\boldsymbol{x}_t}, \boldsymbol{\Sigma}_{\boldsymbol{x}_t})$. The field change consisted in adding a constant value (a DC level) $d_c = 10$ to all field values.

The simulated data was built considering a $20\times$ 20 m^2 room with one access point placed at the center as depicted in Fig. 1. The room size and location of the transmitter were selected in order to spatially centralize the peak of the RSS field and thus provide a better error assessment. In this scenario we compare the recursive GP (RGP) and the RGP with marginalization of the inputs, namely, *RGPMarg*. For this, we first trained a GP using the fingerprinting data (Fig. 1) and then updated it using RGP or RGPMarg with new measurements taken in uncertain locations. Fig. 2 shows the resulting GP field estimations where the mean is plotted as a 3D surface (first row) and with a top view (middle row), while the covariance

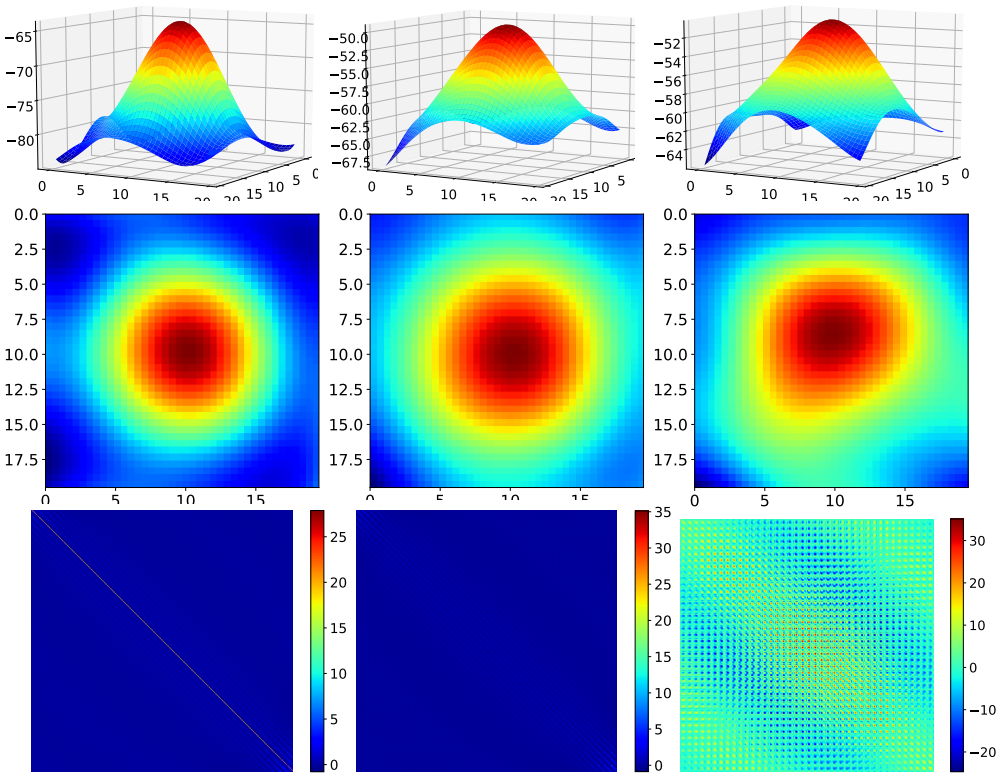

Fig. 2. From left to right: Initial GP for the initial fingerprinting data, RGP, RGPMarg after $d_c = +10$ was added, $\text{STD}_{\text{MAX}} = 5$ and $\sigma_\eta = 4$, and $n = 3000$ samples. From top down: plot of GPs mean (3D), GPs means (top view) and GPs covariance matrices.

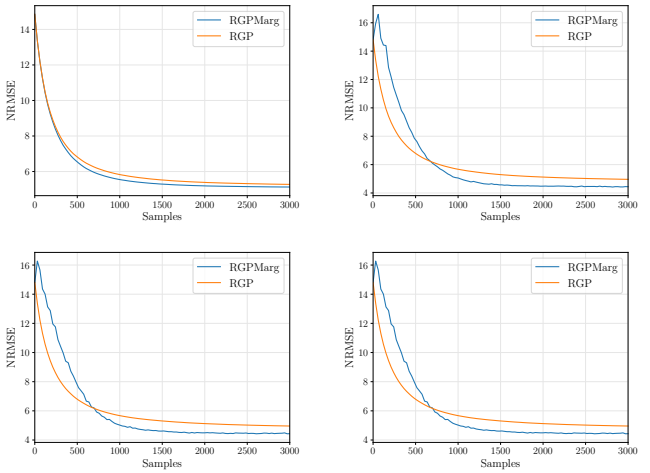

Fig. 3. NRMSE for 100 MC runs, $\sigma_\eta = 4$, $\sigma_x^{\max} = [1, 3, 5, 10]$.

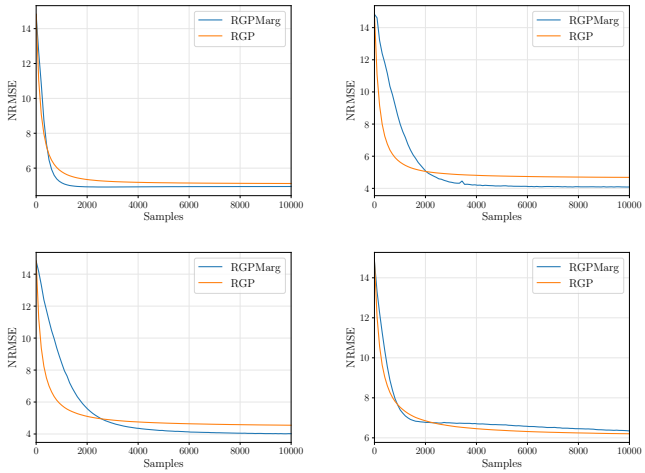

Fig. 4. NRMSE for 10 MC runs, $\sigma_\eta = 4$, fixed $\sigma_x = [1, 3, 5, 10]$.

matrices (bottom row) are depicted as images with values coded by color. For these results we modeled $\boldsymbol{\Sigma}_{\boldsymbol{x}_t} = \xi_t \sigma_x^{\max} \boldsymbol{I}$, with $\xi_t \sim \mathcal{U}_{[0,1]}$, being a constant simulating a time varying uncertainty. The rationale for this time-varying strategy is to mimic results provided by tracking algorithms.

The results in Fig. 2 show marignal performance improvement provided by GPMarg which can be noticed by an overall higher value around the peak. This analysis is coherent with NRMSE computed which both methods $\text{NRMSE}_{\text{RGPMarg}}(\boldsymbol{\psi}_t) = 4.66$ and $\text{NRMSE}_{\text{RGP}}(\boldsymbol{\psi}_t) = 5.08$. Another interesting characteristic of these results is related to the GP covariance matrix. Analysing the covariance matrices in Fig. 2 one can notice that RGP model confidence is erroneously much higher (i.e., providing smaller covariance elements) than the RGPMarg solution. This happens since RGP neglects completely the noisy positioning labels. This

result is very important specially when considering the integration of the field estimation method with tracking filters in charge of computing the position estimate, which may become overconfident in the measurement model. As a result, one of the main features of RGPMarg is the higher, realistic uncertainty characterization on the top of the mean field representation.

Figures 3 and 4 present the NRMSE evolution as new samples are available. In Fig. 3 we considered $\boldsymbol{\Sigma}_{\boldsymbol{x}_t} = \xi_t \sigma_x^{\max} \boldsymbol{I}$ for $\sigma_x^{\max} \in \{1, 3, 5, 10\}$, which allows for eventual large uncertanties but also very precise samples. In contrast, we also performed analogous simulations fixing the covariance matrix. Fig. 4 presents results with fixed $\boldsymbol{\Sigma}_{\boldsymbol{x}_t} = \sigma_x \boldsymbol{I}$ with $\sigma_x \in \{1, 3, 5, 10\}$. The results show smaller RNMSE for RGPMarg after convergence, but a usually faster convergence of RGP. The only exception are the results for $\boldsymbol{\Sigma}_{\boldsymbol{x}_t} = 10 \times \boldsymbol{I}$ (bottom-right panel of Fig. 4) where RGP presented a better performance, and for $\boldsymbol{\Sigma}_{\boldsymbol{x}_t} = \boldsymbol{I}$ (top-left panel of Fig. 4) where RGPMarg was both faster and more accurate. These results show the performance gains obtained by marginalizing uncertainty even in such a simplistic scenario. We highlight that the simulated pathloss fields are very smooth without sharp discontinuities, reflections and attenuation existing even in simple real scenarios. This sharp changes in the field would lead to bigger errors being propagated throughout the recursive estimations.

B. Tracking Performance

In this section we analyse the tracking performance when field changes and uncertainty are present. We implemented a Cubature Kalman Filter (CKF) [28], using the same dynamical model used in [6]. In this simulation the scenario consists of a 40×40 m^2 room with 5 access points (APs) as depicted in Fig. 5 (left). Following the same strategy of our previous simulations, here, we also consider an initial fingerpringing stage followed by a change in the field and the arrival of new measurements $\boldsymbol{y}_t$ taken at uncertain locations $\boldsymbol{x}_t \sim \mathcal{N}(\boldsymbol{\mu}_{\boldsymbol{x}_t}, \boldsymbol{\Sigma}_{\boldsymbol{x}_t})$ with fixed $\boldsymbol{\Sigma}_{\boldsymbol{x}_t} = \boldsymbol{I}$. To assess the tracking performance with the different strategies we generated 100 tracks with 500 points each using a random walk with covariance $\boldsymbol{Q} = q_t \boldsymbol{I}$, with $q_t = 0.1$. To simulate field changes we added DC levels, $\boldsymbol{d}_c = [15, 5, 10, 10, 20]$, to the APs and the resulting RSS field is depicted in Figure 5 (right). Figure 6 presents the RMSE for position estimation. We can see that both recursive strategies performed much better than the static GP even with presence of noise in the locations. Furthermore, a marginal accuracy gain is observable when comparing the RGPMarg and RGP even with this very simplistic scenario.

VI. Conclusions

This paper deals with the regeneration of fingerprints for positioning purposes. Particularly, we consider a setup where RSS fingerprints are used to train a Gaussian Process (GP) that characterize the RSS spatial field of a set of anchor nodes, which is then used in a data-fusion filter for tracking and positioning. The paper tackles the issue of those fingerprints

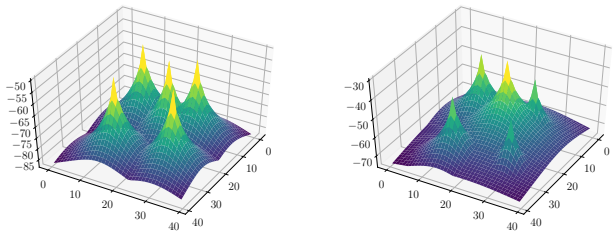

Fig. 5. RSS fields. Initial fields (left) and fields after change (right).

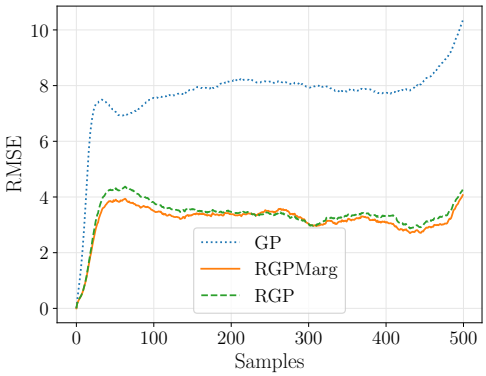

Fig. 6. RMSE evolution for position estimation averaged over 100 tracks.

evolving over time, and thus the need for the database to adapt as well. One sound possibility is the use of a Recursive GP (RGP) approach in order to implement such on-line learning. Additionally, the new RSS values might be recorded at locations whose positions might not be accurately known, but with some associated uncertainty. In this contribution, we proposed a methodology to incorporate such uncertainty by a marginalization process in the RGP. A set of simulation results are provided to validate i) the field estimation update with marginalization, and ii) the tracking performance using the proposed GP strategies. The performance improvements of considering uncertain locations is not only seen in the differet RMSE results, but in the feature of providing more reliable uncertainty measures. We achieve a more realistic covariance estimation of the RGP which impacts the performance of the positioning solution. Future work will explore the effective use of such position-dependent covariance in a practical scheme, as well as the use of real-data.

References

[1] D. Dardari, P. Closas, and P. M. Djuric, "Indoor tracking: Theory, methods, and technologies," *IEEE Transactions on Vehicular Technology*, vol. 64, no. 4, pp. 1263–1278, 2015.

[2] G. M. Mendoza-Silva, P. Richter, J. Torres-Sospedra, E. S. Lohan, and J. Huerta, "Long-Term WiFi Fingerprinting Dataset for Research on Robust Indoor Positioning," *Data*, vol. 3, no. 1, 2018. [Online]. Available: https://doaj.org/article/885221491cae424fb9d9295cd32eb518

[3] E. Arias-de Reyna, P. Closas, D. Dardari, and P. M. Djuric, "Crowd-Based Learning of Spatial Fields for the Internet of Things: From Harvesting of Data to Inference," *IEEE Signal Processing Magazine*, vol. 35, no. 5, pp. 130–139, 2018.

[4] E. Arias-de Reyna, D. Dardari, P. Closas, and P. M. Djurić, "Enhanced indoor localization through crowd sensing," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2017, pp. 2487–2491.
[5] I. Santos, J. J. Murillo-Fuentes, and P. M. Djurić, "Recursive Estimation of Dynamic RSS Fields Based on Crowdsourcing and Gaussian Processes," *IEEE Transactions on Signal Processing*, vol. 67, no. 5, pp. 1152–1162, 2019.
[6] P. Wu, T. Imbiriba, G. LaMountain, J. Vilà-Valls, and P. Closas, "WiFi Fingerprinting and Tracking Using Neural Networks," in *ION GNSS+, The International Technical Meeting of the Satellite Division of The Institute of Navigation*, Miami, Florida, September 2019, pp. 2314–2324.
[7] M. Ibrahim, M. Torki, and M. ElNainay, "CNN based indoor localization using RSS time-series," in *2018 IEEE Symposium on Computers and Communications (ISCC)*. IEEE, 2018, pp. 01 044–01 049.
[8] K. S. Kim, S. Lee, and K. Huang, "A scalable deep neural network architecture for multi-building and multi-floor indoor localization based on Wi-Fi fingerprinting," *Big Data Analytics*, vol. 3, no. 1, p. 4, 2018.
[9] S. Mahfouz, F. Mourad-Chehade, P. Honeine, J. Farah, and H. Snoussi, "Target tracking using machine learning and kalman filter in wireless sensor networks," *IEEE Sensors Journal*, vol. 14, no. 10, pp. 3715–3725, 2014.
[10] L. S. Muppirisetty, T. Svensson, and H. Wymeersch, "Spatial wireless channel prediction under location uncertainty," *IEEE Transactions on Wireless Communications*, vol. 15, no. 2, pp. 1031–1044, 2015.
[11] C. K. I. Williams and C. E. Rasmussen, *Gaussian processes for machine learning*. MIT press Cambridge, MA, 2006, vol. 2, no. 3.
[12] T. Imbiriba, J. C. M. Bermudez, J.-Y. Tourneret, and C. Richard, "Detection of nonlinear mixtures using Gaussian processes: Application to hyperspectral imaging," in *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2014, pp. 7949–7953.
[13] T. Imbiriba, J. C. M. Bermudez, C. Richard, and J.-Y. Tourneret, "Nonparametric detection of nonlinearly mixed pixels and endmember estimation in hyperspectral images," *IEEE Transactions on Image Processing*, vol. 25, no. 3, pp. 1136–1151, 2016.
[14] P. I. Frazier, "A tutorial on Bayesian optimization," *arXiv preprint arXiv:1807.02811*, 2018.
[15] M. Jadaliha, Y. Xu, J. Choi, N. S. Johnson, and W. Li, "Gaussian process regression for sensor networks under localization uncertainty," *IEEE Transactions on Signal Processing*, vol. 61, no. 2, pp. 223–237, 2012.
[16] H. Liu, Y.-S. Ong, X. Shen, and J. Cai, "When Gaussian process meets big data: A review of scalable GPs," *arXiv preprint arXiv:1807.01065*, 2018.
[17] M. Titsias, "Variational learning of inducing variables in sparse gaussian processes," in *Artificial Intelligence and Statistics*, 2009, pp. 567–574.
[18] T. D. Bui, J. Yan, and R. E. Turner, "A unifying framework for Gaussian process pseudo-point approximations using power expectation propagation," *The Journal of Machine Learning Research*, vol. 18, no. 1, pp. 3649–3720, 2017.
[19] M. P. Deisenroth and J. W. Ng, "Distributed Gaussian Processes," *arXiv preprint arXiv:1502.02843*, 2015.
[20] T. N. Hoang, Q. M. Hoang, and B. K. H. Low, "A distributed variational inference framework for unifying parallel sparse gaussian process regression models." in *ICML*, 2016, pp. 382–391.
[21] M. F. Huber, "Recursive Gaussian process: On-line regression and learning," *Pattern Recognition Letters*, vol. 45, pp. 85–91, 2014.
[22] S. Särkkä, *Bayesian filtering and smoothing*. Cambridge University Press, 2013, vol. 3.
[23] C. M. Bishop, *Pattern recognition and machine learning*. springer, 2006.
[24] G. H. Golub and J. H. Welsch, "Calculation of Gauss quadrature rules," *Mathematics of computation*, vol. 23, no. 106, pp. 221–230, 1969.
[25] H. Engels, "Numerical quadrature and cubature," 1980.
[26] P. J. Davis and P. Rabinowitz, *Methods of numerical integration*. Courier Corporation, 2007.
[27] A. Goldsmith, *Wireless communications*. Cambridge university press, 2005.
[28] I. Arasaratnam and S. Haykin, "Cubature Kalman filters," *IEEE Trans. Automatic Control*, vol. 54, no. 6, pp. 1254–1269, June 2009.