

Gaussian Processes for RSS Fingerprints Construction in Indoor Localization

Yuxin Zhao¹, Chao Liu², Lyudmila S Mihaylova², Fredrik Gunnarsson¹

1. Ericsson Research, Linköping, SE-58330, Sweden

2. Department of Automatic Control and Systems Engineering, University of Sheffield, Sheffield, S1 3JD, UK

(E-mail: {Yuxin.Zhao, Fredrik.Gunnarsson}@ericsson.com, {cliu47, l.s.mihaylova}@sheffield.ac.uk)

Abstract—Location-based applications attract more and more attention in recent years. Examples of such applications include commercial advertisements, social networking software and patient monitoring. The received signal strength (RSS) based location fingerprinting is one of the most popular solutions for indoor localization. However, there is a big challenge in collecting and maintaining a relatively large RSS fingerprint database. In this work, we propose and compare two algorithms namely, the Gaussian process (GP) and Gaussian process with variogram, to estimate and construct the RSS fingerprints with incomplete data. The fingerprint of unknown reference points is estimated based on measurements at a limited number of surrounding locations. To validate the effectiveness of both algorithms, experiments using Bluetooth-low-energy (BLE) infrastructure have been conducted. The constructed RSS fingerprints are compared to the true measurements, and the result is analyzed. Finally, using the constructed fingerprints, the localization performance of a probabilistic fingerprinting method is evaluated.

Index Terms—Indoor localization, received signal strength, Gaussian process, variogram, location fingerprinting.

I. INTRODUCTION

Wireless positioning has received considerable attention due to the ever-increasing demands on location-based services (LBS) in various areas. Wireless network infrastructures include, but not limited to, cellular radio networks (LTE and 5G), Wi-Fi networks, Bluetooth-low-energy (BLE) networks. A cheap but competent solution is to use the received signal strength (RSS) as position related measurements since they are easy to obtain for most of the existing networks.

Ranging-base positioning algorithms, such as trilateration, depend highly on the range-related signal propagation model. A plethora of explicit RSS models already exist in the literature, for instance the conventional linear log-distance model and ray-tracing model [1]. Alternatively, the scene analysis methods estimate target locations based on maps of features. For example, the RSS-based location fingerprinting method calculates a target location depending on a set of RSS maps.

Furthermore, for each base station involved, a map consisting of RSS measured at all reference points (RPs) is necessary. For higher positioning accuracy, a map of refined RPs (more RPs in the same area) is essential, but the first-time construction and later maintenance could be labor-intensive and time-consuming.

Given measurements at some known points, the interpolation/extrapolation algorithms can be used to estimate measurements at new locations. The paper [2] summarizes and compares three common-used interpolation/extrapolation algorithms for constructing RSS fingerprints. However, those algorithms are fairly simple for indoor RSS measurements, considering the complex indoor propagation conditions. Advanced method, such as data-driven method, can be more accurate and flexible in modeling and interpolating/extrapolating RSS measurement in terms of 2D geometry.

Nowadays, there is an increasing interest of using the non-parametric Gaussian process (GP) for RSS modeling. The reasons are twofold. First, it is a powerful tool for exploring the relationship in a set of variables given the training dataset. Second, GP perfectly fits in the Bayesian framework, which allows for explicit probabilistic interpretation of the model outputs [3]. Prior work includes [4]–[8]. The key difference lies in the use of different kernel functions in the non-parametric model. For instance, Matérn kernel was used in [4] and squared-exponential (SE) kernel used in [8]. Recently, [9] compared different kernels thoroughly using simulations and proved that the SE kernel works quite well in most cases. Most recent work on kernel design include [10]–[12]. GP based RSS models have been used in different Bayesian filters, which include extended Kalman filter (EKF), unscented Kalman filter (UKF) and particle filter [5].

In the meanwhile, Kriging methods, or sometimes called Gaussian process regression with a variogram, which are considered belonging to the family of Gaussian process, been adopted jointly with a RSS fingerprint database in several recent works. The simple Kriging method is used in [13], while the ordinary Kriging in [14] and the universal Kriging in [15]. The paper [16] proposes a new distance measure for wall attenuation in spatial correlation modeling for the Kriging method. A modified ordinary Kriging is proposed in [17] and a modified universal Kriging is proposed in [18].

The contributions of this paper are as follows. First, theoretical descriptions of the two methods are introduced in the context of RSS fingerprints constructions. Second, to answer the frequent question about the differences between the GP and GP with variogram, theoretic details are given and results from

algorithms performance validation are presented. To the best of authors' knowledge, this is the first time that the two methods are compared in the context of localization. At last, the two methods are applied to real BLE data. Results of the RSS fingerprint construction and performance of a probabilistic fingerprinting method based on the constructed database are evaluated and analyzed.

The remainder of this paper is organized as follows: Section II and Section III present the GP and GP with variogram methods in context of RSS fingerprint construction. In Section IV, the adopted probabilistic fingerprinting algorithm is demonstrated. Section V concludes the similarities and differences between the two methods theoretically. The collection and processing of practical BLE data will be described at the beginning of Section VI-A, followed by experimental results and analysis. Section VII concludes the paper. An Appendix presents the algorithm for hyperparameter estimation.

Throughout this paper, matrices are presented with uppercase letters and vectors with boldface lowercase letters. The operator $[\cdot]^T$ stands for vector/matrix transpose and $[\cdot]^{-1}$ stands for the inverse of a non-singular square matrix. The operator $|\cdot|$ stands for determinant of a matrix. The operator $\|\cdot\|_2$ stands for ℓ_2 norm of a vector. The operator $\text{diag}(\Sigma)$ returns the diagonal elements of matrix Σ , and $[\Sigma]_{i,j}$ gives the entry at the i th row and j th column of Σ . Further, $\mathcal{N}(\mu, \sigma^2)$ denotes a Gaussian distribution with mean μ and variance σ^2 . Notation $\mathbf{0}_N$ means an $N \times N$ matrix of all zeros, \mathbf{I}_N denotes an identity matrix and $\mathbf{1}$ denotes a vector of all ones. The operator \ln and \lg denote logarithm to base e and 10, respectively. The operator $\mathbb{E}[\cdot]$ is the mathematical expectation.

II. STANDARD GAUSSIAN PROCESS FOR FINGERPRINT CONSTRUCTION

Fingerprinting methods estimate the location of a target by consulting a pre-defined fingerprint database. In deterministic framework [19], [20], a fingerprint consists of a measurement vector, and the corresponding location (i.e., this can be considered as an augmented vector $[\mathbf{r}, \mathbf{p}]$, where \mathbf{r} is the measurement and \mathbf{p} is the corresponding location where the measurement has been collected). Once the target measurements are obtained, metrics such as the ℓ_1 -norm or ℓ_2 -norm are used to justify the similarity between the target and fingerprints. Afterwards, the target location is given by either the most matching fingerprint, or a weighted linear combination of several most matching fingerprints where the weights are computed as the inverse of aforementioned metrics.

In this work, a probabilistic fingerprinting method is considered. More precisely, in the database construction phase, the RSS from m base stations are first collected at the pre-defined RPs, whose locations are denoted by $\mathbf{Q} = \{\mathbf{q}_1, \dots, \mathbf{q}_S\}$. The s -th fingerprint consists of the location \mathbf{q}_s and a set of Gaussian distributed measurements in which the m -th distribution is of posterior mean $\bar{r}_{m,s}$ and variance $\bar{\sigma}_{m,s}^2$.

In order to construct an RSS fingerprinting map, we first aim to train a model for each reference network node to specify

the RSS distribution. The model¹ we use here is given by:

$$r(\mathbf{p}) = \mu(\mathbf{p}) + e(\mathbf{p}) + w, \quad (1)$$

where $r(\mathbf{p})$ denotes the RSS observed at any position \mathbf{p} (in x , y , and z dimensions in general) from the m th reference network node, $\mu(\mathbf{p})$ is the noise-free RSS which is also considered as a propagation function, $e(\mathbf{p})$ is a position dependent noise term which represents the shadowing effects, and w is a position independent noise term which account for the joint influence of the interference from other devices, signal absorption from human bodies, (unsuccessfully removed) small-scale fading, as well as the background noise. We assume w to be independent and identically Gaussian distributed with zero mean and variance σ_w^2 .

We follow an empirical propagation function given below:

$$\mu(\mathbf{p}) = A + 10B \cdot \log_{10} d(\mathbf{p}), \quad (2)$$

where A is the RSS measured at 1 meter away, B is the path loss exponent, and $d(\mathbf{p})$ denotes the Euclidean distance between \mathbf{p} and the corresponding network node. More sophisticated function, i.e., $\mu(\mathbf{p})$ that take into account the wall effect and other factors, can also be considered.

Given a training dataset $\mathcal{D}_m \triangleq \{\mathbf{P}_m, \mathbf{r}_m\}$ for the m -th network node and rewrite (1) in matrix form, we have

$$\mathbf{r}_m = \boldsymbol{\mu}_m + \mathbf{e}_m + \mathbf{w}_m, \quad (3)$$

where \mathbf{P}_m is the set of training positions, \mathbf{r}_m , $\boldsymbol{\mu}_m$, \mathbf{e}_m and \mathbf{w}_m are composite vectors of corresponding variables, that are given by:

$$\mathbf{P}_m \triangleq [\mathbf{p}_{m,1}^T, \dots, \mathbf{p}_{m,N}^T]^T, \quad (4a)$$

$$\mathbf{r}_m \triangleq [r(\mathbf{p}_{m,1}), \dots, r(\mathbf{p}_{m,N})]^T, \quad (4b)$$

$$\boldsymbol{\mu}_m \triangleq [\mu(\mathbf{p}_{m,1}), \dots, \mu(\mathbf{p}_{m,N})]^T, \quad (4c)$$

$$\mathbf{e}_m \triangleq [e(\mathbf{p}_{m,1}), \dots, e(\mathbf{p}_{m,N})]^T, \quad (4d)$$

$$\mathbf{w}_m \triangleq [w_{m,1}, \dots, w_{m,N}]^T, \quad (4e)$$

and N is the number of data points in the training set.

A. Characterizing Spatial Correlation

With standard Gaussian process (SGP), the position-dependent noise term, \mathbf{e}_m , is modeled by

$$\mathbf{e}_m \sim \mathcal{GP}(\mathbf{0}, \mathbf{K}_m(\mathbf{P}_m, \mathbf{P}_m)), \quad (5)$$

where (we follow the notation $\mathcal{GP}(\cdot, \cdot)$ used in [3]) the covariance (kernel) function $\mathbf{K}_m(\mathbf{P}_m, \mathbf{P}_m)$ characterize the spatial correlation of the shadowing effect, of which the i, j th element is given by

$$[\mathbf{K}_m(\mathbf{P}_m, \mathbf{P}_m)]_{i,j} = k_m(\mathbf{p}_{m,i}, \mathbf{p}_{m,j}) = \mathbb{E}[e(\mathbf{p}_{m,i})e(\mathbf{p}_{m,j})]. \quad (6)$$

¹The observed RSS values are often preprocessed. Take LTE system as an example, the RSS values are first averaged at the physical layer and low-pass filtered at the network layer, in order to remove small-scale fading effects.

The kernel function $k_m(\mathbf{p}_{m,i}, \mathbf{p}_{m,j})$ can take any appropriate form, such as the Matérn kernel used in [4], or the well-established squared exponential (SE) model [3]. Here we take the SE model, which is formulated as

$$k_m(\mathbf{p}_{m,i}, \mathbf{p}_{m,j}) = \sigma_{m,e}^2 \cdot \exp \left[\frac{-\|\mathbf{p}_{m,i} - \mathbf{p}_{m,j}\|^2}{2l_{m,c}^2} \right], \quad (7)$$

where $\sigma_{m,e}^2$ accounts for the uncertainty introduced by the shadow fading into the GP model and $l_{m,c}$ denotes the correlation distance.

B. Estimate Model Parameters

The unknown parameters in GP model need to be estimated first. Herein, we assume all positions in the training set \mathcal{D}_m are precisely known. The likelihood function $\mathcal{L}(\mathbf{r}_m | \mathbf{P}_m, \boldsymbol{\theta}_m)$ of the training dataset is given as follows:

$$\mathcal{L}(\mathbf{r}_m | \mathbf{P}_m, \boldsymbol{\theta}_m) \sim \mathcal{N}(\boldsymbol{\mu}_m, \mathbf{C}_m), \quad (8)$$

with the following notations:

$$\boldsymbol{\theta}_m \triangleq [A_m, B_m, \sigma_{m,e}, l_{m,c}, \sigma_{m,w}]^T, \quad (9a)$$

$$\mathbf{C}_m \triangleq \mathbf{K}_m(\mathbf{P}_m, \mathbf{P}_m) + \sigma_{m,w}^2 \cdot \mathbf{I}_N. \quad (9b)$$

Then, the maximum-likelihood estimate (MLE) can be adopted here to obtain a good approximation of the underlying parameters $\boldsymbol{\theta}_m$. Further description is given in the Appendix.

C. Construct New Fingerprints

Having the trained model for the m -th network node, according to [3] we are able to jointly compute the posterior means and variances for RSS measurement distribution for all RPs, that is

$$\bar{\mathbf{r}}_m = \mathbf{K}_m^T(\mathbf{Q}, \mathbf{P}_m) \mathbf{C}_m^{-1} (\mathbf{r}_m - \boldsymbol{\mu}_m) + \boldsymbol{\mu}_m, \quad (10a)$$

$$\bar{\mathbf{C}}_m = \mathbf{K}_m(\mathbf{Q}, \mathbf{Q}) - \mathbf{K}_m^T(\mathbf{Q}, \mathbf{P}_m) \mathbf{C}_m^{-1} \mathbf{K}_m(\mathbf{Q}, \mathbf{P}_m) + \sigma_{m,w}^2 \mathbf{I}_S, \quad (10b)$$

where $\bar{\mathbf{r}}_m$ is composed of the posterior means of difference RPs, i.e., $\bar{\mathbf{r}}_m = [\bar{r}_{m,1}, \dots, \bar{r}_{m,S}]^T$, and $\bar{\mathbf{C}}_m$ is a matrix with the diagonal entries being the posterior variances of difference RPs, i.e., $\text{diag}(\bar{\mathbf{C}}_m) = [\bar{\sigma}_{m,1}^2, \dots, \bar{\sigma}_{m,S}^2]^T$. Therefore, the probabilistic fingerprint of the s -th RP can be composed of the corresponding values in all $\bar{\boldsymbol{\mu}}$ and $\bar{\mathbf{C}}$.

III. GAUSSIAN PROCESS WITH VARIOGRAM METHOD FOR FINGERPRINT CONSTRUCTION

The Gaussian process with variogram (GPV) method [21], which is also called Kriging, named after Danie G. Krige, refers to a group of least-squared based interpolation methods, which are widely used in geostatistics discipline. GPV is also considered as a kind of Gaussian process based method, but with a different implementation. The GPV method used in this work is the *universal Kriging* (UK), which is given below.

Given the training dataset $\mathcal{D}_m \triangleq \{\mathbf{P}_m, \mathbf{r}_m\}$ for the m -th network node, the GPV requires an RSS model that is similar to (1) and (2), i.e.,

$$r(\mathbf{p}_{m,i}) = \mu(\mathbf{p}_{m,i}) + R(\mathbf{p}_{m,i}), \quad (11)$$

where $R(\mathbf{p}_{m,i})$ represents the location-dependent noise. Comparing (3) and (11), the location-independent noise $w_{m,i}$ is not considered here.

The aim is to build new RSS fingerprints. Similar to the SGP method, the GPV estimates an unknown RSS value based on the known data points and a spatial correlation model of the location-dependent residual $R(\mathbf{p}_{m,i})$. It needs to be emphasized that the residual $R(\mathbf{p}_{m,i})$ needs to be decomposed from the $r(\mathbf{p}_{m,i})$.

A. Characterizing Spatial Correlation

To characterize the spatial correlation of $R(\mathbf{p}_{m,i})$, the assumption of stationarity, i.e., *intrinsic stationarity*, is required: by constructing a new variable, which is the difference between the residual of two neighbor points $\delta_{i,j} = R(\mathbf{p}_{m,i}) - R(\mathbf{p}_{m,j})$, the intrinsic stationarity implies that the mean of $\delta_{i,j}$ is zero in the local neighborhood and the variance of $\delta_{i,j}$ depends only on the separation distance $\|\mathbf{p}_{m,i} - \mathbf{p}_{m,j}\|$, i.e., $\mathbb{E}[\delta_{i,j}] = 0$, and $\text{Var}(\delta_{i,j}) = 2\gamma(\|\mathbf{p}_{m,i} - \mathbf{p}_{m,j}\|) = 2\gamma(h)$, where $\gamma(\cdot)$ is the *variogram* function, and $h = \|\mathbf{p}_{m,i} - \mathbf{p}_{m,j}\|$, is called *lag* which represents the separation distance between $\mathbf{p}_{m,i}$ and $\mathbf{p}_{m,j}$.

To obtain a variogram, the classic regression analysis method is adopted. First, all possible pairs of locations are categorized according to the lag h . Then, the *empirical variogram*, $\hat{\gamma}(h)$, is calculated as

$$\hat{\gamma}(h) = \frac{1}{2} \cdot \frac{1}{U(h)} \sum_{i=1}^{U(h)} \left(R(\mathbf{p}_{m,i}) - R(\mathbf{p}_{m,i+h}) \right)^2, \quad (12)$$

where $U(h)$ is the number of location pairs whose lag is h , and $\mathbf{p}_{m,i+h}$ represents the location with lag h to $\mathbf{p}_{m,i}$. A regression model is then selected to fit the empirical variogram using the least squares method [22]. This model is frequently chosen from *spherical* model, *exponential* model, *Gaussian* model, *power* model and *linear* model [23]. The obtained regression model is used later to retrieve variogram value for any lag h .

B. Estimate Parameters and Build New Fingerprints

The GPV method estimates the value at a new location \mathbf{q}_s ($s = 1, \dots, S$) as a weighted sum of known neighbor data points, that is,

$$\bar{r}_{m,s} = \sum_{i=1}^H \lambda_i \cdot r(\mathbf{p}_{m,i}) \quad (13a)$$

$$= \sum_{i=1}^H \lambda_i \left(\beta_{m,1} + \beta_{m,2} \log_{10} d_m(\mathbf{p}_{m,i}) + R(\mathbf{p}_{m,i}) \right), \quad (13b)$$

where $\lambda_1, \dots, \lambda_H$ are GPV weights, H represents the number of data points within the correlation distance $l_{m,c}$, $\beta_{m,1}$ and $\beta_{m,2}$ are unknown parameters and are not necessary to estimate.

To get an optimized estimate, the GPV weights are derived through minimizing the estimator error variance, that is,

$$\min_{\lambda_i \in R} \text{Var}(\bar{r}_{m,s} - \mu_m(\mathbf{q}_s)), \quad (14)$$

under the unbiasedness constraint:

$$\mathbb{E}[\bar{r}_{m,s} - \mu_m(\mathbf{q}_s)] = 0. \quad (15)$$

The optimized GPV weights and parameters are derived as

$$\begin{bmatrix} \lambda_1 \\ \vdots \\ \lambda_H \\ L_{m,1} \\ L_{m,2} \end{bmatrix} = \begin{bmatrix} \gamma^{(1,1)} & \dots & \gamma^{(1,H)} & 1 & f_m^{(1)} \\ \vdots & \ddots & \vdots & \vdots & \vdots \\ \gamma^{(H,1)} & \dots & \gamma^{(H,H)} & 1 & f_m^{(H)} \\ 1 & \dots & 1 & & \\ f_m^{(1)} & \dots & f_m^{(H)} & & \end{bmatrix}^{-1} \begin{bmatrix} \gamma^{(1,s)} \\ \vdots \\ \gamma^{(H,s)} \\ 1 \\ f_m^{(s)} \end{bmatrix}, \quad (16)$$

where $L_{m,1}$ and $L_{m,2}$ are multipliers that result from utilizing the *Lagrange multiplier* optimization method, and some variables are written in shorthand as

$$\gamma^{(i,j)} = \gamma(\|\mathbf{p}_{m,i} - \mathbf{p}_{m,j}\|), \quad (17a)$$

$$\gamma^{(i,s)} = \gamma(\|\mathbf{p}_{m,i} - \mathbf{q}_s\|), \quad (17b)$$

$$f_m^{(i)} = \log_{10} d_m(\mathbf{p}_{m,i}), \quad (17c)$$

$$f_m^{(s)} = \log_{10} d_m(\mathbf{q}_s). \quad (17d)$$

Therefore, the posterior mean and variance of the s -th RP can be given by

$$\bar{r}_{m,s} = \sum_{i=1}^H \lambda_i \cdot r(\mathbf{p}_{m,i}), \quad (18)$$

$$\bar{\sigma}_{m,s}^2 = \sum_{i=1}^H \lambda_i \gamma^{(i,s)} + L_{m,1} + L_{m,2} \cdot \log_{10} d_m(\mathbf{q}_s). \quad (19)$$

Algorithm 1 GP Based Probabilistic Fingerprinting

Compute RPs Weights: For $\mathbf{q}_s (s = 1, 2, \dots, S)$, the weight is computed by

$$\alpha_s = \prod_{m=1}^M \frac{1}{\sqrt{(2\pi)\bar{\sigma}_{m,s}^2}} \exp\left(-\frac{1}{2} \frac{(r_m - \bar{r}_{m,s})^2}{\bar{\sigma}_{m,s}^2}\right). \quad (20)$$

Find K -Nearest Neighbors: Sort the weights, $\alpha_s (s = 1, 2, \dots, S)$. The largest $K (K \leq S)$ weights are then normalized to sum to one, i.e.,

$$\bar{\alpha}_s = \frac{\alpha_s}{\sum_{s=1}^K \alpha_s}. \quad (21)$$

Position Estimate: An estimate of the unknown position, \mathbf{p} , is computed by

$$\hat{\mathbf{p}} = \sum_{s=1}^K \bar{\alpha}_s \mathbf{q}_s. \quad (22)$$

IV. LOCATION FINGERPRINTING METHOD

In what above, we have proposed two advanced methods to construct RSS fingerprints. Then, in the position retrieval phase (testing phase), given a target measurements $\mathbf{r} = [r_1, \dots, r_M]^T$, the probabilities $\alpha_s (s = 1, \dots, S)$ which justify the similarity are then computed for each fingerprint. After

normalizing K largest probabilities into weights $\bar{\alpha}_s$, the target location estimate $\hat{\mathbf{p}}$ is given as a weighted linear combination of several fingerprints. More details of the employed probabilistic fingerprinting method are given in Algorithm 1.

V. COMPARISON BETWEEN STANDARD GAUSSIAN PROCESS AND GAUSSIAN PROCESS WITH VARIOGRAM

In this section, the comparison between SGP (equation (5) to (10b)) and GPV (equation (11) to (19)) will be discussed. From previous sections, both similarities and difference between the two methods can be observed, which can be summarized as follow.

A. Stationarity Assumption

Both SGP and GPV methods use a function/model to characterize the covariance of random field. The Gaussian process adopts the kernel function $k_m(\mathbf{p}_i, \mathbf{p}_j)$ and the GPV method adopts the variogram $\gamma(\mathbf{p}_i, \mathbf{p}_j)$. Therefore, the stationarity assumptions of both methods shall be investigated.

A strictly stationary random field indicates that the joint probability of two data points, $p(\mathbf{p}_i, \mathbf{p}_j)$, is specified by the distance between \mathbf{p}_i and \mathbf{p}_j , and does not change with location shift of the data points. However, the strict stationarity is usually too strong for signal processing. The second-order stationarity is usually adopted in signal processing, which indicates that only the first order moment (mean) and the second order moment (covariance) is stationary. In the SGP method, the noise component is assumed to be zero mean, and the covariance can be characterized by the kernel function.

For GPV method, the intrinsic stationarity assumption is adopted, which is weaker compared to the second-order stationarity. The intrinsic stationarity indicates that within the neighbor, the expected difference are zeros, and the variogram, which is half the variance of the differences, depends only on the separation distance.

Both the SGP kernel function and the variogram are assumed isotropic, which means the relationship between two data points is independent of the link direction.

B. Modeling and Regression Analysis

In SGP, the component of the model and the kernel function are a priori specified. The parameters of the model components and kernel function can be estimated at the same time using different approaches, e.g., maximum likelihood estimation.

On the contrary, in GPV method, the components of the model can be specified but the parameters need to be estimated first to compute the residuals. The variogram is built upon the residuals through least square regression. The function can be chosen among several basic functions based on fitness.

C. Estimation

To estimate value at an unknown location, standard Gaussian process adopts all training data points and therefore the dimension of the kernel matrix can be extremely large.

In GPV method, based on the decorrelation distance obtained by first analyzing the variogram, data points out of

the decorrelation distance (neighborhood) can be ignored. The known data points within the decorrelation distance have sufficient amount of relationship, and therefore are used to estimate the new points. Following this manner, the variogram matrix can be much smaller than the kernel matrix in SGP.

According to [24], the weights in (13b) should sum to 1. Therefore, (13b) can be further given as

$$\bar{r}_{m,s} = \sum_{i=1}^H \lambda_i \cdot R(\mathbf{p}_{m,i}) + \beta_{m,1} + \beta_{m,2} \log_{10} d_m(\mathbf{p}_{m,i}). \quad (23)$$

Considering $\mathbf{K}_m^T(\mathbf{Q}, \mathbf{P}_m) \mathbf{C}_m^{-1}$ in (10a) as a weights computation system and comparing it with the above equation, it is not hard to find the similarity: both SGP and GPV estimator include a weighted sum of residuals and noise-free term.

D. Computational Complexity

For SGP, the computational complexity mainly depends on the inversion of the kernel matrix which has dimension N_m , where N_m is the number of training data. It is easy to see that for each RP, the complexity to compute the RSS mean and variance from m th network node scales as $\mathcal{O}(N_m^3)$. When the number of training data grows, the computational complexity for SGP increase dramatically.

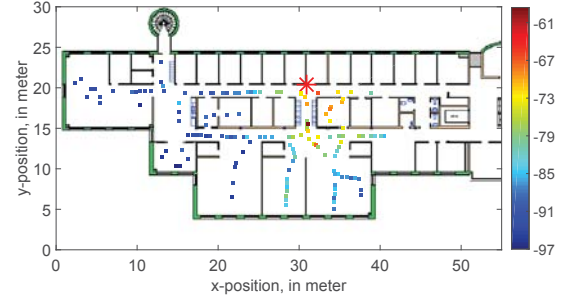
However, as discussed previously, the GPV method only considers the training data points which are within the correlation distance. In such a way, the computational complexity lies in inverting the matrix as shown in (16), which scales as $\mathcal{O}(H^3)$ for each RP. In the case when $H \ll N_m$, GPV yields a much lighter way of constructing the RSS fingerprints as compared to SGP.

VI. FIELD CAMPAIGN

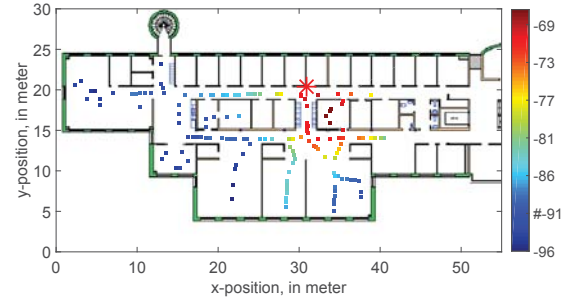
A. Data Collection

Field trials were conducted in a typical office environment at Ericsson Research, Linköping, Sweden. In total $N = 12$ Bluetooth-low-energy (BLE) beacons are placed uniformly in the area, and they serve as the reference network nodes. The floor plan as well as the beacon positions are illustrated in [25, Figure 2]. The BLE beacons serve as transmitters and broadcast data packages regularly. The transmit power is set to $P_T = -58$ dBm identically for all BLE beacons. Moderate scale measurement campaign was conducted during normal work hours. Throughout the field trial, RSS measurements are conducted along predefined tracks and a total number of 28214 RSS measurements are collected from all BLE beacons. The track positions are obtained from an app-based positioning algorithm developed by Senion.

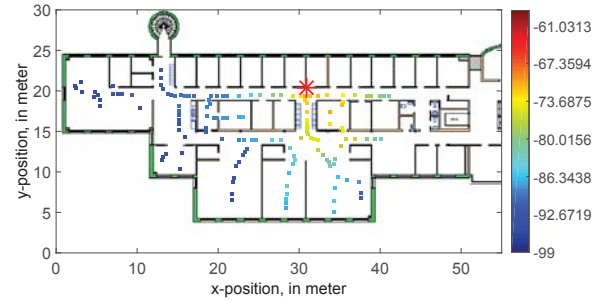
After obtaining the training data, pre-processing is performed. In this step, the collected RSS measurements for each beacon are extracted. For SGP regression, the initial values of A_m , B_m and $\sigma_{n,m}$ used to initialize the parameter optimization are obtained by the Linear Least Square (LLS) regression as given in [8]. For GPV, the model parameters A_m and B_m are assumed to be known and set to the values obtained from the LLS fit.



(a)



(b)



(c)

Fig. 1: 1a: Real measured RSS values for BLE beacon 4. 1b: Estimated RSS values for BLE beacon 4 using standard Gaussian process. 1c: Estimated RSS values for BLE beacon 4 using Gaussian process with variogram.

B. RSS Map Reconstruction Results

In previous sections, two RSS fingerprint construction methods, namely SGP and GPV, have been discussed thoroughly. In order to validate the two methods, the RSS fingerprint construction performance for beacon 4 is given in this section.

To be more precise, we extracted the RSS measurements for BLE beacon 4 from all measurements. Then, 50% of the measurements are randomly selected as training data to train both SGP and GPV models. The rest 50% of the

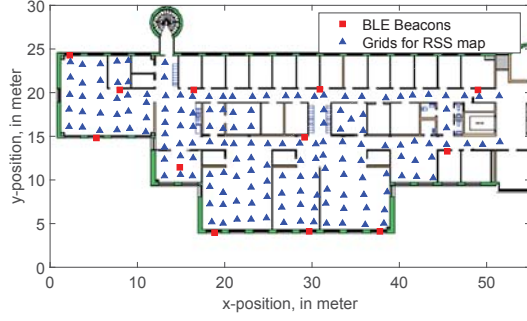


Fig. 2: Predefined reference points used to build up the fingerprinting database.

measurements are used for validation. The corresponding results illustrating the construction performance are shown in Figure 1.

By comparing the real and estimated measurements, it is observed that both SGP and GPV can successfully interpolate a smooth RSS map. Both methods give similar performance from the illustrations. To compare the accuracy of the estimated RSS values, we also calculate the root mean square error (RMSE) for both cases. For SGP model, the RMSE of the estimated RSS measurements is 3 dB and for the GPV, the RMSE is around 2.9 dB.

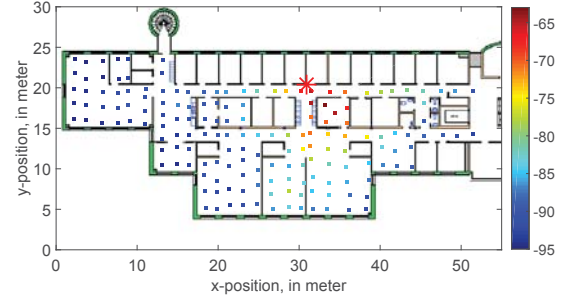
C. Localization Results

1) *RSS Map*: Before performing RSS fingerprinting to obtain the estimated position, we select a set of RPs which build up the fingerprinting database as shown in Figure 2. In order to reduce the storage and wireless communications but in the meanwhile maintain necessary position information, merely 166 grid points are selected to represent the most frequently visited spots. Then, RSS maps are built for all BLE beacons by computing the posterior mean and variance using SGP and GPV, given the pre-collected training data.

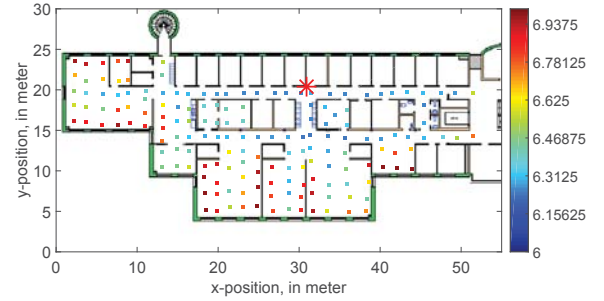
We show the RSS map of the fourth BLE beacon constructed by SGP and GPV in Figure 3 and 4. The posterior mean ranges from -68 dBm to -95 dBm, while the posterior standard deviation ranges from 6 dB to 7 dB for the SGP. The posterior standard deviation is higher at points where there is no training data in the vicinity. By comparing SGP and GPV methods, we can see that the predicted mean at those grids are similar from both methods. However, with SGP, the posterior variance is much larger than the case using GPV.

2) *Experiment Results*: In the previous subsection, the complete RSS maps are constructed with the RPs covering all locations of interest. In what follows, the localization performance of methods given in Section IV will be shown.

We use the 2-D RSS maps obtained in the previous subsection to localize a mobile terminal moving from the west end of the floor to the east end, as indicated by the blue solid line in Figure 5. While walking, the mobile terminal measures



(a)



(b)

Fig. 3: Illustration of the calibrated RSS map for the 4th BLE beacon using SGP: 3a depicts the posterior mean and 3b depicts the posterior standard deviation.

RSS from the BLE beacons at 807 sample positions on this track. The size of the noisy RSS measurements \mathbf{r}_k varies with k at different locations of the floor plan.

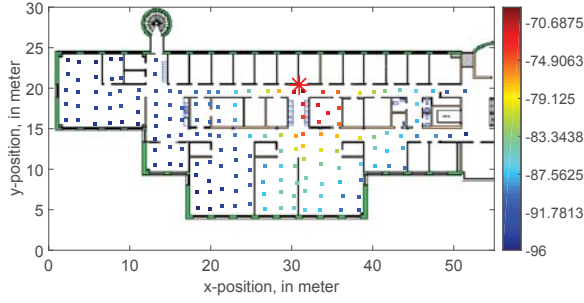
First, we test the GP based probabilistic fingerprinting (FP). We run Algorithm 1 in a snapshot manner for each time instance k . The position estimates are depicted in Figure 5 along with the ground truth trajectory. The corresponding positioning RMSE is computed to be around 4.1 meters with the number of nearest neighbors set to $K = 3$. Secondly, for GPV based fingerprinting, we obtain the estimated positions in a similar manner as described in previous paragraphs. The position estimates are plotted in Figure 5. The RMSE of position estimate is around 4.2 meters.

We tested various settings of K and found that for this specific example $K = 3$ achieves the best performance. Note that in our evaluations, physical constraints, such as the mobile terminal cannot go through a wall, are not taken into account.

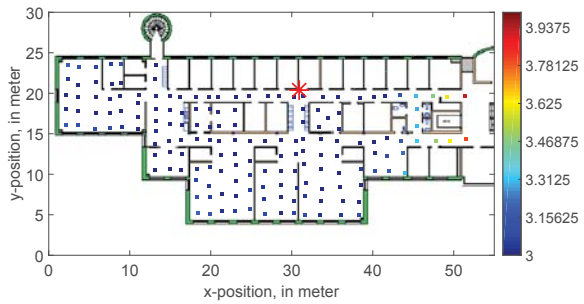
In addition to the positioning RMSE, we further evaluate the cumulative distribution function (CDF) of the position estimation error. The error is computed at each position as follows:

$$E_k = \sqrt{(\hat{p}_{x_k} - p_{x_k}^*)^2 + (\hat{p}_{y_k} - p_{y_k}^*)^2}, \quad (24)$$

where \hat{p}_{x_k} , \hat{p}_{y_k} denote the estimated position and $p_{x_k}^*$, $p_{y_k}^*$



(a)



(b)

Fig. 4: Illustration of the calibrated RSS map for the 4th BLE beacon using GPV: 4a depicts the posterior mean and 4b depicts the posterior standard deviation.

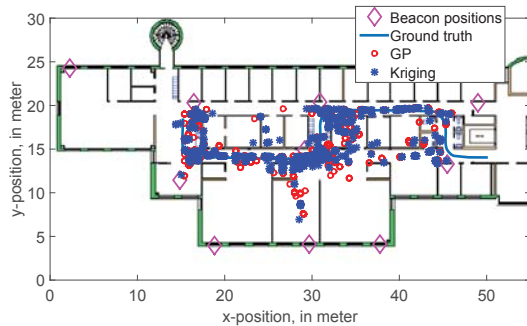


Fig. 5: Position estimates obtained from SGP and GPV based probabilistic fingerprinting with the number of nearest neighbors $K = 3$.

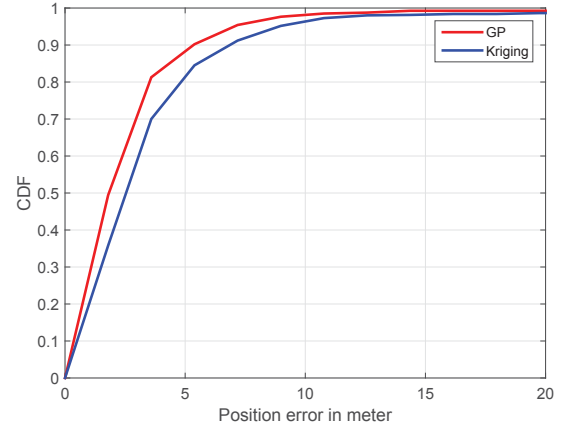


Fig. 6: CDF curves of estimation error for both SGP and GPV based RSS maps.

denote the ground truth. The CDF curves for various positioning algorithms are illustrated in Figure 6. It is observed that SGP outperforms GPV method. From the results, it can be concluded that SGP is more accurate in position estimation, but with relatively higher computational complexity. On the other hand, GPV yields less computational complexity and is more suitable for applications not requiring high positioning accuracy.

VII. CONCLUSIONS

This work focuses on indoor localization with Gaussian processes. Two approaches are implemented and their performance are compared. These are the standard Gaussian process and a Gaussian process with variograms for RSS fingerprints construction. By combining either SGP or GPV method with probabilistic fingerprinting algorithms, positioning performance has been evaluated within an office environment with experimental data. It is concluded that both methods provide satisfactory performance to construct RSS fingerprints and thus provide more powerful and more flexible tools in fingerprinting based positioning methods. Both methods result in good positioning accuracy, while each has its own advantages. Using both methods, the average positioning accuracy in an ordinary office layout is around 2 – 3 meters.

VIII. ACKNOWLEDGEMENT

This work is funded by the European Union FP7 Marie Curie training programme on Tracking in Complex Sensor Systems (TRAX) with grant number 607400. Furthermore, we acknowledge the support from Senion, who provided the BLE beacons as well as associated positioned RSS measurements.

IX. APPENDIX

The maximum-likelihood estimate of the GP model parameters, $\hat{\theta}_m$, can be obtained by maximizing the Gaussian

prior likelihood function, cf.(8), with respect to θ_m , which is equivalent to

$$\arg \min_{\theta_m} g(\theta_m) \triangleq (\mathbf{r}_m - \mu_m)^T \mathbf{C}_m^{-1} (\mathbf{r}_m - \mu_m) + \ln |\mathbf{C}_m|. \quad (25)$$

Various existing numerical methods can be adopted to solve this minimization problem, such as the batch processing based limited-memory BFGS (LBFGS) quasi-Newton method [3] or stochastic gradient descent for reduced complexity when the number of training samples is large. Herein, we adopt the former method (implemented in [26]) which requires the first-order derivatives of the cost function, $g(\theta_m)$. Due to the form of mean function $\mu_m(\mathbf{p})$ in (1) and the SE kernel function in (7), the first-order derivatives are computed as follows:

$$\frac{\partial g(\theta_m)}{\partial A_m} = \mathbf{a}_m (\mathbf{C}_m^{-1} + (\mathbf{C}_m^{-1})^T) (\mathbf{r}_m - \mu_m) \quad (26a)$$

$$\frac{\partial g(\theta_m)}{\partial B_m} = \mathbf{b}_m (\mathbf{C}_m^{-1} + (\mathbf{C}_m^{-1})^T) (\mathbf{r}_m - \mathbf{m}_m) \quad (26b)$$

$$\frac{\partial g(\theta_m)}{\partial \sigma_{e,m}^2} = \text{tr} \left\{ \left[\mathbf{C}_m^{-1} - (\mathbf{C}_m^{-1} (\mathbf{r}_m - \mu_m)) (\cdot)^T \right] \frac{\partial \mathbf{C}_m}{\partial \sigma_{e,m}^2} \right\} \quad (26c)$$

$$\frac{\partial g(\theta_m)}{\partial l_{c,m}} = \text{tr} \left\{ \left[\mathbf{C}_m^{-1} - (\mathbf{C}_m^{-1} (\mathbf{r}_m - \mu_m)) (\cdot)^T \right] \frac{\partial \mathbf{C}_m}{\partial l_{c,m}} \right\} \quad (26d)$$

$$\frac{\partial g(\theta_m)}{\partial \sigma_{n,m}^2} = \text{tr} \left\{ \left[\mathbf{C}_m^{-1} - (\mathbf{C}_m^{-1} (\mathbf{r}_m - \mu_m)) (\cdot)^T \right] \frac{\partial \mathbf{C}_m}{\partial \sigma_{n,m}^2} \right\}, \quad (26e)$$

where

$$\mathbf{a}_m \triangleq \frac{\partial (\mathbf{r}_m - \mu_m)^T}{\partial A_m} = -\mathbf{1}^T, \quad (27a)$$

$$\mathbf{b}_m \triangleq \frac{\partial (\mathbf{r}_m - \mu_m)^T}{\partial B_m} = -10[\tilde{d}_{m,1}, \tilde{d}_{m,2}, \dots, \tilde{d}_{m,N_m}], \quad (27b)$$

$$\tilde{d}_{m,j} \triangleq \lg d_{m,j}, \quad (27c)$$

$$\left[\frac{\partial \mathbf{C}_m}{\partial \sigma_{e,m}^2} \right]_{j,k} = \begin{cases} 1, & j = k \\ \exp \left[\frac{-\|\mathbf{p}_{m,j} - \mathbf{p}_{m,k}\|_2^2}{2l_{c,m}^2} \right], & j \neq k \end{cases} \quad (27d)$$

$$\left[\frac{\partial \mathbf{C}_m}{\partial l_{c,m}} \right]_{j,k} = \begin{cases} 0, & j = k \\ \sigma_{e,m}^2 \exp \left[\frac{-\|\mathbf{p}_{m,j} - \mathbf{p}_{m,k}\|_2^2}{2l_{c,i}^2} \right] \frac{\|\mathbf{p}_{m,j} - \mathbf{p}_{m,k}\|_2^2}{l_{c,m}^3}, & j \neq k \end{cases} \quad (27e)$$

$$\frac{\partial \mathbf{C}_m}{\partial \sigma_{n,m}^2} = \mathbf{I}_{N_m}. \quad (27f)$$

Here we use $(A)(\cdot)^T$ to denote $(A)(A)^T$ for brevity.

REFERENCES

- [1] A. Goldsmith, *Wireless Communications*. New York, NY, USA: Cambridge University Press, 2006.
- [2] J. Talvitie, M. Renfors, and E. S. Lohan, "Distance-based interpolation and extrapolation methods for RSS-based localization with indoor wireless signals," *IEEE Transactions on Vehicular Technology*, vol. 64, no. 4, pp. 1340–1353, April 2015.
- [3] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning*. Cambridge, MA, USA: MIT Press, 2006.
- [4] A. Schwaighofer, M. Grigoras, V. Tresp, and C. Hoffmann, "GPPS: A Gaussian process positioning system for cellular networks," in *Advances in Neural Information Processing Systems (NIPS)*. MIT Press, 2004.
- [5] B. Ferris, D. Hhnel, and D. Fox, "Gaussian processes for signal strength-based location estimation," in *In Proc. of Robotics: Science and Systems (RSS)*, Philadelphia, USA, August 2006, pp. 1–8.
- [6] R. Di Taranto, S. Muppirisetty, R. Raulefs, D. Sloock, T. Svensson, and H. Wymeersch, "Location-aware communications for 5G networks: How location information can improve scalability, latency, and robustness of 5G," *IEEE Signal Process. Mag.*, vol. 31, no. 6, pp. 102–112, Nov. 2014.
- [7] 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, Feb 2016.
- [8] F. Yin, Y. Zhao, and F. Gunnarsson, "Proximity report triggering threshold optimization for network-based indoor positioning," in *Proc. Int. Conf. on Information Fusion*, Washington D.C., USA, July 2015, pp. 1061–1069.
- [9] P. Richter and M. Toledano-Ayala, "Revisiting Gaussian process regression modeling for localization in wireless sensor networks," *Sensors (Basel)*, vol. 15, no. 9, pp. 22587–22615, Sep. 2015.
- [10] T. Chen, "Continuous-time DC kernel - a stable generalized first-order spline kernel," *provisionally accepted by IEEE Trans. on Automatic Control*, 2018.
- [11] —, "On kernel design for regularized LTI system identification," *Automatica*, vol. 90, pp. 109–122, 2018.
- [12] F. Yin, X. He, L. Pan, Z. Luo, and T. Chen, "Sparse structure enabled grid spectral mixture kernel for temporal Gaussian process regression," in *21st International Conference on Information Fusion*, submitted, 2018.
- [13] S. S. Jan, S. J. Yeh, and Y. W. Liu, "Received signal strength database interpolation by Kriging for a Wi-Fi indoor positioning system," *Sensors (Basel, Switzerland)*, vol. 15, no. 9, pp. 21377–21393, 2015.
- [14] C. Liu, A. Kiring, N. Salman, and L. Mihaylova, "A Kriging Algorithm for Location Fingerprinting based on Received Signal Strength," in *Proc. 10th Work. Sens. Data Fusion*, Bonn, Germany, 2015.
- [15] B. Li, Y. Wang, H. K. Lee, A. Dempster, and C. Rizos, "Method for Yielding a Database of Location Fingerprints in WLAN," *IEE Proc. Commun.*, vol. 152, no. 5, pp. 580–586, 2005.
- [16] A. Konak, "Estimating Path Loss in Wireless Local Area Networks Using Ordinary Kriging," in *Proc. 2010 Winter Simul. Conf.*, Baltimore, MD, 2010, pp. 2888–2896.
- [17] A. Manirabona and L. C. Fourati, "A Kriged Fingerprinting for Wireless Body Area Network Indoor Localization," *Wirel. Pers. Commun.*, vol. 80, no. 4, pp. 1501–1515, 2015.
- [18] D. Wu and L. Xia, "Reference Device-Assisted Adaptive Location Fingerprinting," *Sensors*, vol. 16, no. 6, p. 802, 2016. [Online]. Available: <http://www.mdpi.com/1424-8220/16/6/802>
- [19] P. Bahl and V. N. Padmanabhan, "Radar: An in-building rf-based user location and tracking system," in *EEE INFOCOM 2000. The 19th annual conference on Computer Communications*, 2000.
- [20] M. Bshara, U. Orguner, F. Gustafsson, and L. Van Biesen, "Fingerprinting localization in wireless networks based on received-signal-strength measurements: A case study on WiMAX networks," *IEEE Trans. Vehicular Technology*, vol. 59, no. 1, pp. 283–294, Jan. 2010.
- [21] D. G. Krige, "A Statistical Approach to Some Basic Mine Valuation Problems on the Witwatersrand," Ph.D. dissertation, the University of Witwatersrand, Johannesburg, South Africa, 1951.
- [22] S. Weisberg, *Applied Linear Regression*, 3rd ed. John Wiley & Sons, 2013.
- [23] G. C. Bohling, "Introduction To Geostatistics and Variogram Analysis," Lecture Notes, 2005. [Online]. Available: <http://people.ku.edu/~gbohling/cpe940/Variograms.pdf>
- [24] N. A. C. Cressie, *Statistics for Spatial Data*. John Wiley, 1993.
- [25] Y. Zhao, C. Fritsche, F. Yin, F. Gunnarsson, and F. Gustafsson, "Sequential Monte Carlo methods and theoretical bounds for state inference based on proximity reports," *To appear in IEEE Transactions on Vehicular Technology*, 2018.
- [26] C. E. Rasmussen and H. Nickisch, "Gaussian processes for machine learning (GPML) toolbox," *J. Mach. Learn. Res.*, vol. 11, pp. 3011–3015, Dec. 2010.