Gaussian Process Regression under Location Uncertainty using Monte Carlo Approximation

Martin Ptáček
Department of Radio Electronics
Brno University of Technology
Brno, Czech Republic
xptace18@vut.cz
Brno Ph.D. Talent Scholarship Holder
Funded by the Brno City Municipality

***Abstract*—Gaussian Process Regression (GPR) is a common statistical framework for spatial function estimation. While its flexibility and availability of closed-form estimation solution after training are its advantages, it suffers on applicability constraints in scenarios with uncertain training positions. This paper presents the derivation of the exact GPR operating on uncertain training positions along with approximation of the resulting terms using Monte Carlo (MC) sampling. This method is then implemented in a simulation environment and shown to improve the estimation quality over the standard GPR approach with uncertain training positions.**

***Index Terms*—Spatial function estimation, GPR, Uncertain training positions, Probabilistic inference, Monte Carlo approximation**

I. Introduction

Spatial function estimation is a problem common to numerous scientific and practical fields including geostatistics, astronomy, meteorology, mining but also telecommunications and other machine learning applications. In the field of telecommunications the application can be found in estimating Channel state information maps to aid the development of mobile networks and optimization resource allocation [1] [2].

GPR is a method for spatial function estimation that recently gained popularity because of its success as a general machine learning method and its interpretation as a neural network with infinite basis functions [3]. The advantages of GPR are its flexibility and the availability of a closed form expression for the estimates after the training phase. Further significant advantages are the ability of GPR to indicate the level of belief (uncertainty) for the provided estimates and its performance on limited data sets.

GPR relies on precise knowledge of the training positions. In many practical scenarios, this may not be available and the performance of the GPR framework is prone to degradation. This motivates the search for approaches capable of overcoming this issue. This paper is focused on presenting the derivation of MC-GPR method, presenting the simulation setup, demonstrating the performance gains offered by this approach and the increased computation complexity costs.

The author would like to thank Ao.Univ.Prof. Franz Hlawatsch (TU Wien) for many helpful discussions on Gaussian Process Regression and early research guidance as well as to doc. RNDr. Jitka Poměnková, Ph.D. (BUT) for her comments.

Research described in this paper was supported by the Internal Grant Agency of the Brno University of Technology under project no. FEKT-S-20-6361.

II. Spatial Function Estimation

A spatial function in the chosen scenario may be imagined as a distribution of a scalar parameter over a continuous space. The space considered here shall be $D = 2$ dimensional, representing for example the Earth's surface. The task is to estimate the value of the spatial function at a given *test position* given the observations done by sensors placed at some given *training positions*. An example of such situation is depicted in Figure 1.

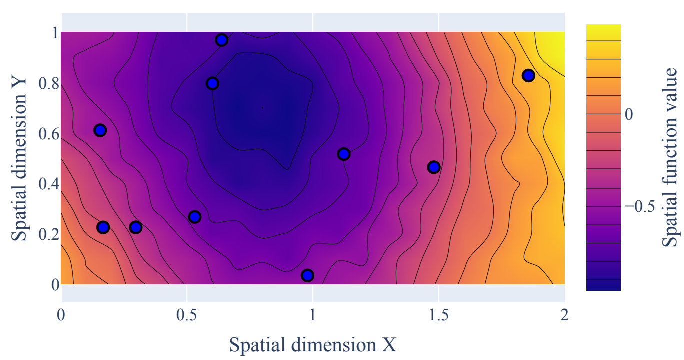

Fig. 1. Realization of a spatial function (represented by contour levels) and training positions (indicated by blue dots).

A popular framework for solving the Spatial function estimation problem is the Gaussian Process Regression.

III. Gaussian Process Regression Basics

A GP is a collection of random variables (RVs), any finite number of which have a joint Gaussian distribution.

The GP will be denoted as $f(\boldsymbol{x})$, where $\boldsymbol{x} \in \mathbb{R}^D$ is an argument specifying the position within the GP, in this case considered as a general position. The complete GP is then defined as

$$p(f(\boldsymbol{x})) = \mathcal{GP}(m(\boldsymbol{x}), k(\boldsymbol{x}, \boldsymbol{x}')) \,, \tag{1}$$

DOI 10.13164/eeict.2023.222

where $\boldsymbol{x}, \boldsymbol{x}' \in \mathbb{R}^D$ are generic position variables. Further, $m(\boldsymbol{x}) \triangleq \mathrm{E}\{f(\boldsymbol{x})\}$ is the *mean function* and $k(\boldsymbol{x}, \boldsymbol{x}') \triangleq \mathrm{cov}\{f(\boldsymbol{x}), f(\boldsymbol{x}')\}$ is the *covariance function* of the GP. $p(\cdot)$ here denotes *probability density function* but is further used also as *probability mass function* depending on the context.

A. Observation Model

We shall consider that the GP is observed at I known *training positions* $\boldsymbol{x}^{(i)}$ with $i = 1, 2, \ldots, I$ stacked to a vector according to

$$\boldsymbol{x}_\mathrm{t} \triangleq \mathrm{col}(\boldsymbol{x}^{(1)}, \boldsymbol{x}^{(2)}, \ldots, \boldsymbol{x}^{(I)}) \in \mathbb{R}^{DI} \ . \quad (2)$$

The GP RVs $f(\boldsymbol{x}^{(i)})$ at the individual training positions $\boldsymbol{x}^{(i)}$ shall also be arranged into a vector according to

$$\boldsymbol{f} \triangleq (f(\boldsymbol{x}^{(1)}) \quad f(\boldsymbol{x}^{(2)}) \quad \ldots \quad f(\boldsymbol{x}^{(I)}))^\mathrm{T} \in \mathbb{R}^I \ . \quad (3)$$

It follows from (1) that $\boldsymbol{f}$ is a random vector with multivariate Gaussian distribution

$$p(\boldsymbol{f}) = \mathcal{N}(\boldsymbol{f};\, \boldsymbol{m},\, \boldsymbol{K}) \ , \quad (4)$$

where $\boldsymbol{m} = \mathrm{E}\{\boldsymbol{f}\}$ denotes the vector of the mean values and $\boldsymbol{K} = \mathrm{cov}\{\boldsymbol{f}\}$ the covariance matrix.

In the considered problem setup we assume not having direct access to the GP RVs $\boldsymbol{f}$ at the training positions. Instead, we are provided with vector $\boldsymbol{y}$ of observations of $\boldsymbol{f}$, i.e.,

$$\boldsymbol{y} \triangleq (y^{(1)} \quad y^{(2)} \quad \ldots \quad y^{(I)})^\mathrm{T} \in \mathbb{R}^I \ , \quad (5)$$

which is a noisy version of $\boldsymbol{f}$ according to

$$\boldsymbol{y} = \boldsymbol{f} + \boldsymbol{\epsilon} \ , \quad (6)$$

where $\boldsymbol{\epsilon} \triangleq (\epsilon^{(1)} \quad \epsilon^{(2)} \quad \ldots \quad \epsilon^{(I)})^\mathrm{T} \in \mathbb{R}^I$ is the vector of observation errors (measurement noise). $\boldsymbol{\epsilon}$ is assumed to be zero-mean isotropic Gaussian according to

$$p(\boldsymbol{\epsilon}) = \mathcal{N}(\boldsymbol{\epsilon};\, \boldsymbol{0},\, \sigma_\epsilon^2 \mathbf{I}_I) \ , \quad (7)$$

where σ_ϵ^2 is the single observation error variance and $\mathbf{I}_I$ is an identity matrix of dimension I.

It is assumed that $\boldsymbol{\epsilon}$ is independent of $\boldsymbol{f}$. From (6) and our statistical assumptions it follows that $\boldsymbol{y}$ is a Gaussian random vector distributed according to

$$p(\boldsymbol{y}) = \mathcal{N}(\boldsymbol{y};\, \boldsymbol{m},\, \boldsymbol{Q}) \quad (8)$$

with a covariance matrix $\boldsymbol{Q}$ of the random vector $\boldsymbol{y}$ given as

$$\boldsymbol{Q} = \mathrm{cov}\{\boldsymbol{f}\} + \mathrm{cov}\{\boldsymbol{\epsilon}\} = \boldsymbol{K} + \sigma_\epsilon^2 \mathbf{I}_I \ , \quad (9)$$

where we used the independence of random vectors $\boldsymbol{f}$ and $\boldsymbol{\epsilon}$.

B. Regression Formulation

The aim of the GPR is to estimate the GP RV at a single known *test position* $\boldsymbol{x}^{(*)} \in \mathbb{R}^D$, i.e., $f(\boldsymbol{x}^{(*)})$, which will be denoted shortly as f_*. To utilize the observations of the GP $\boldsymbol{y}$ at the training positions $\boldsymbol{x}_\mathrm{t}$ in order to estimate $f(\boldsymbol{x}^{(*)})$, the GPR framework expresses the posterior distribution (predictive distribution) of the RV to be estimated given the observed RVs according to the Bayes' Theorem as

$$p(f_*|\boldsymbol{y}) = \frac{p(\boldsymbol{y}|f_*)\, p(f_*)}{p(\boldsymbol{y})} = \frac{p(f_*, \boldsymbol{y})}{p(\boldsymbol{y})} \ . \quad (10)$$

The likelihood term $p(\boldsymbol{y})$ is recognized as a normalizing constant. The posterior pdf can therefore be determined using the joint pdf $p(f_*, \boldsymbol{y})$ only, i.e., $p(f_*|\boldsymbol{y}) \propto p(f_*, \boldsymbol{y})$.

According to the definition of the GP, the joint distribution is Gaussian and given by

$$p(f_*, \boldsymbol{y}) = \mathcal{N}\left(\begin{pmatrix} f_* \\ \boldsymbol{y} \end{pmatrix};\, \begin{pmatrix} m_* \\ \boldsymbol{m} \end{pmatrix},\, \begin{pmatrix} k_* & \boldsymbol{c}^\mathrm{T} \\ \boldsymbol{c} & \boldsymbol{Q} \end{pmatrix}\right) \ , \quad (11)$$

where m_* and k_* are the prior mean and variance of f_*, i.e. the GP value at the test position, respectively. Vector $\boldsymbol{c} \triangleq \mathrm{cov}\{\boldsymbol{y},\, f_*\}$ is the cross-covariance of the vector of spatial function observations $\boldsymbol{y} = \boldsymbol{f} + \boldsymbol{\epsilon}$ and the GP random variable f_* at the test position $\boldsymbol{x}^{(*)}$. We obtain

$$\boldsymbol{c} = \mathrm{cov}\{\boldsymbol{f} + \boldsymbol{\epsilon},\, f_*\} = \mathrm{cov}\{\boldsymbol{f},\, f_*\} + \mathrm{cov}\{\boldsymbol{\epsilon},\, f_*\} \in \mathbb{R}^I \ . \quad (12)$$

Since $\boldsymbol{\epsilon}$ and f_* are independent, $\mathrm{cov}\{\boldsymbol{\epsilon},\, f_*\} = \boldsymbol{0}$ and thus we get $\boldsymbol{c} = \mathrm{cov}\{\boldsymbol{f},\, f_*\}$, which can be evaluated using the covariance function k as

$$\boldsymbol{c} = (k(\boldsymbol{x}^{(1)}, \boldsymbol{x}^{(*)}) \quad k(\boldsymbol{x}^{(2)}, \boldsymbol{x}^{(*)}) \quad \ldots \quad k(\boldsymbol{x}^{(I)}, \boldsymbol{x}^{(*)}))^\mathrm{T} \ . \quad (13)$$

Since the posterior distribution $p(f_*|\boldsymbol{y})$ is Gaussian, it is fully specified using the posterior mean $\mu_{f_*|y}$ and variance $\sigma^2_{f_*|y}$. These parameters can be expressed using the *completing the square* approach that expresses the posterior distribution according to the joint distribution (11) in a single exponential form, i.e., as a new Gaussian distribution

$$p(f_*|\boldsymbol{y}) = \mathcal{N}(f_*;\, \mu_{f_*|y},\, \sigma^2_{f_*|y}) \ , \quad (14)$$

where the posterior mean value is expressed as

$$\mu_{f_*|y} = m_* + \boldsymbol{c}^\mathrm{T} \boldsymbol{Q}^{-1} (\boldsymbol{y} - \boldsymbol{m}) \ . \quad (15)$$

This is the *Minimum Mean-squared Error* (MMSE) estimate of f_* given $\boldsymbol{y}$. The posterior variance $\sigma^2_{f_*|y}$ can be expressed in a similar way. The expressions for variance are omitted also in the following sections since their derivation is similar to the ones of the posterior mean [3, ch. 2] [4, sec. 2.3].

IV. GPR Under Uncertain Training Positions

In the previous section the training positions $\boldsymbol{x}^{(i)}$ were considered known. In practical applications this information may not be available. We shall therefore consider that the training positions are known only up to their probability density functions (pdfs) $p(\boldsymbol{x}^{(i)})$, which can be expressed for all the training positions jointly as

$$p(\boldsymbol{x}_\mathrm{t}) = p(\boldsymbol{x}^{(1)}, \boldsymbol{x}^{(2)}, \ldots, \boldsymbol{x}^{(I)}) \ . \tag{16}$$

Since now the training positions are random and the GP RVs $\boldsymbol{f}$ need to have their position explicitly specified by conditioning on the random training positions to attain the original meaning, i.e.,

$$\boldsymbol{f}|\boldsymbol{x}_\mathrm{t} = (f(\boldsymbol{x}^{(1)}) \quad f(\boldsymbol{x}^{(2)}) \quad \ldots \quad f(\boldsymbol{x}^{(I)}))^\mathrm{T} \in \mathbb{R}^I \ . \tag{17}$$

The posterior distribution as in (10) is then obtained by conditioning all the pdfs on $\boldsymbol{x}_\mathrm{t}$, which yields

$$p(f_*|\boldsymbol{y}, \boldsymbol{x}_\mathrm{t}) = \frac{p(f_*, \boldsymbol{y}|\boldsymbol{x}_\mathrm{t})}{p(\boldsymbol{y}|\boldsymbol{x}_\mathrm{t})} \ . \tag{18}$$

The term in (18) corresponds to the standard GPR setup with known training positions $\boldsymbol{x}_\mathrm{t}$ and would operate equally as (10) if we knew them. Since we do not know $\boldsymbol{x}_\mathrm{t}$ and only the pdf (16) of this vector is available, we need to express the *unconditional* pdf the GP RV at the test position f_* given the observations vector $\boldsymbol{y}$, i.e., $p(f_*|\boldsymbol{y})$, which can be obtained from $p(f_*, \boldsymbol{x}_\mathrm{t}|\boldsymbol{y})$ using the *sum rule* as

$$p(f_*|\boldsymbol{y}) = \int_{\mathbb{R}^{DI}} p(f_*, \boldsymbol{x}_\mathrm{t}|\boldsymbol{y}) \, d\boldsymbol{x}_\mathrm{t} \ , \tag{19}$$

which can be further developed using the *product rule* as

$$p(f_*|\boldsymbol{y}) = \int_{\mathbb{R}^{DI}} p(f_*|\boldsymbol{y}, \boldsymbol{x}_\mathrm{t}) p(\boldsymbol{x}_\mathrm{t}) \, d\boldsymbol{x}_\mathrm{t} \ . \tag{20}$$

The posterior pdf (20) is not a Gaussian pdf in general. Nevertheless, we will approximate it as a Gaussian pdf using its 1st two moments to achieve a similar term as in (14), i.e.,

$$p(f_*|\boldsymbol{y}) \sim \mathcal{N}(f_*; \, \mathrm{E}\{f_*|\boldsymbol{y}\}, \, \mathrm{var}\{f_*|\boldsymbol{y}\}) \ . \tag{21}$$

A. Posterior Mean

The mean value of the posterior distribution in (21) can be expressed as

$$\mathrm{E}\{f_*|\boldsymbol{y}\} = \int_{\mathbb{R}} f_* p(f_*|\boldsymbol{y}) \, df_* \ , \tag{22}$$

which after plugging (20) becomes

$$\begin{aligned} \mathrm{E}\{f_*|\boldsymbol{y}\} &= \int_{\mathbb{R}} f_* \int_{\mathbb{R}^{DI}} p(f_*|\boldsymbol{y}, \boldsymbol{x}_\mathrm{t}) p(\boldsymbol{x}_\mathrm{t}) \, d\boldsymbol{x}_\mathrm{t} \, df_* && (23) \\ &= \int_{\mathbb{R}^{DI}} \int_{\mathbb{R}} f_* p(f_*|\boldsymbol{y}, \boldsymbol{x}_\mathrm{t}) \, df_* p(\boldsymbol{x}_\mathrm{t}) \, d\boldsymbol{x}_\mathrm{t} && (24) \\ &= \int_{\mathbb{R}^{DI}} \mu_*(\boldsymbol{x}_\mathrm{t}) p(\boldsymbol{x}_\mathrm{t}) \, d\boldsymbol{x}_\mathrm{t} \ , && (25) \end{aligned}$$

where

$$\mu_*(\boldsymbol{x}_\mathrm{t}) \triangleq \mathrm{E}\{f_*|\boldsymbol{y}, \boldsymbol{x}_\mathrm{t}\} = \int_{\mathbb{R}} f_* p(f_*|\boldsymbol{y}, \boldsymbol{x}_\mathrm{t}) \, df_* \tag{26}$$

is the standard GPR posterior mean value according to (15) considering known training positions $\boldsymbol{x}_\mathrm{t}$.

V. Monte Carlo Approximation

The integrals used for evaluating the posterior mean in (25) cannot be computed in closed form in general. As suggested in [1], this problem can be solved by using the *Monte Carlo* (MC) approximation, which provides a solution in a computationally feasible way.

To express the MC approximation of the posterior mean in (25) we firstly reformulate it as an expectation with respect to the pdf $p(\boldsymbol{x}_\mathrm{t})$, i.e.,

$$\mathrm{E}\{f_*|\boldsymbol{y}\} = \mathrm{E}^{p(\boldsymbol{x}_\mathrm{t})}\{\mu_*(\boldsymbol{x}_\mathrm{t})\} \ . \tag{27}$$

The pdf of training positions $p(\boldsymbol{x}_\mathrm{t})$ can now be approximated using a set of samples $\boldsymbol{x}_{\mathrm{t},i}, i = 1, \ldots, s$ drawn according to $p(\boldsymbol{x}_\mathrm{t})$ with a point-mass function

$$p_\mathrm{MC}(\boldsymbol{x}_\mathrm{t}) = \frac{1}{s} \sum_{i=1}^{s} \delta(\boldsymbol{x}_\mathrm{t} - \boldsymbol{x}_{\mathrm{t},i}) \ , \tag{28}$$

where $\delta(\cdot)$ is the Dirac delta function. The posterior mean in (27) can then be approximated as

$$\begin{aligned} \mathrm{E}^\mathrm{MC}\{f_*|\boldsymbol{y}\} &= \mathrm{E}^{p_\mathrm{MC}(\boldsymbol{x}_\mathrm{t})}\{\mu_*(\boldsymbol{x}_\mathrm{t})\} \\ &= \frac{1}{s} \sum_{i=1}^{s} \mu_*(\boldsymbol{x}_{\mathrm{t},i}) \ . && (29) \end{aligned}$$

It shall be noted that the computation complexity of (29) grows linearly with the number of MC samples s compared to the standard GPR as in (15).

VI. Simulation Setup

To verify the improved performance of the MC-GPR we developed a modular simulation setup in Python available at [5]. A block diagram representing simplified functionality of a single simulation run is depicted in Figure 2.

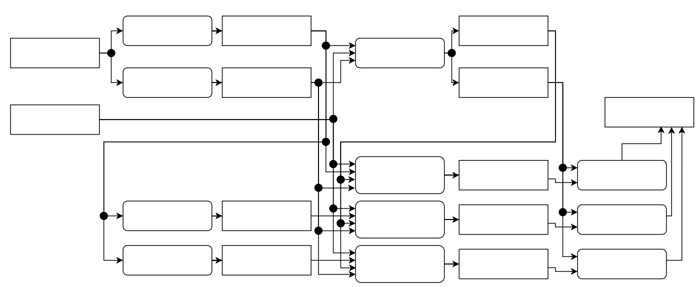

Fig. 2. Simulation setup scheme.

In order to be able to evaluate the performance of the investigated method in a fully described case, several simulation parameters need to be assumed. We shall consider a 2-dimensional space spanned by a rectangle of size 2×1 in arbitrary spatial units. Within this rectangle, 10 random, uniformly distributed training positions are assumed and sampled.

The unknown spatial function (GP) is sampled at the test positions jointly with training positions. The test positions are placed uniformly on a grid within the considered rectangle with density of 10 test positions per spatial unit, i.e., resulting in $21 \times 11 = 231$ test positions. The estimates of spatial function and root-mean-square error (RMSE) evaluation are further performed only on these test positions.

The mean function of the GP used for simulations was considered to be constant, zero. The covariance function was chosen to be in the squared exponential form according to

$$k(\boldsymbol{x}, \boldsymbol{x}') = \sigma^2 \exp\left(-\frac{\|\boldsymbol{x} - \boldsymbol{x}'\|^2}{2\sigma_{\mathrm{x}}^2}\right) , \tag{30}$$

where the variance and spatial scale parameters were chosen as $\sigma^2 = 1$ and $\sigma_{\mathrm{x}}^2 = 1$ respectively. The result of sampling a GP is depicted in Figure 1, which represents the ground-truth values of a spatial function to be estimated later. The observations of the spatial function at the training positions were evaluated according to (6), where the measurement noise variance was chosen as $\sigma_\epsilon^2 = 10^{-4}$.

Evaluating the standard GPR with known training positions according to (15) results in Figure 3 for the posterior mean and in Figure 4 for the posterior variance respectively.

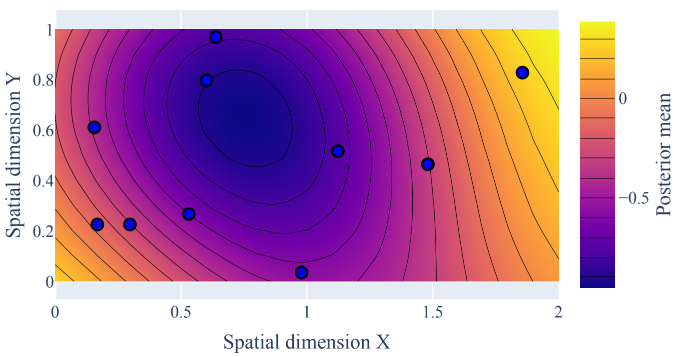

Fig. 3. GPR posterior mean values (represented by contour levels) using true training positions (indicated by blue dots).

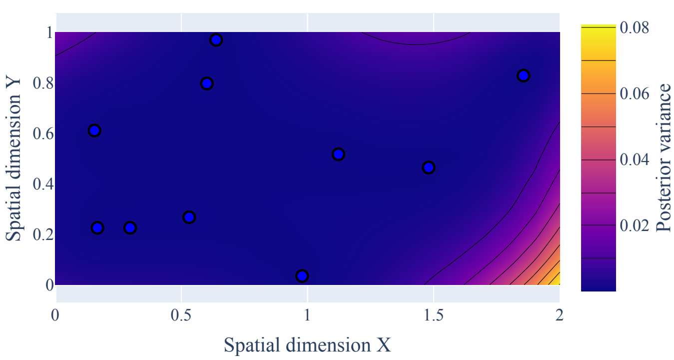

Fig. 4. GPR posterior variance (represented by contour levels) using true training positions (indicated by blue dots).

Since the training positions shall be further assumed uncertain, we will consider that only observations $\tilde{\boldsymbol{x}}^{(i)}$ containing additive Gaussian noise according to

$$p(\tilde{\boldsymbol{x}}^{(i)}) = \mathcal{N}(\tilde{\boldsymbol{x}}^{(i)};\, \boldsymbol{x}^{(i)},\, \sigma_{\mathrm{v}}^2 \mathbf{I}_I) , \qquad i = 1, \ldots, I . \tag{31}$$

with $\sigma_{\mathrm{v}}^2 = 10^{-2}$ are available. Using these observed training positions instead of the original ones in the standard GPR method results in degraded estimation quality, which is apparent in Figure 5 showing the posterior mean and in Figure 6 showing the posterior variance.

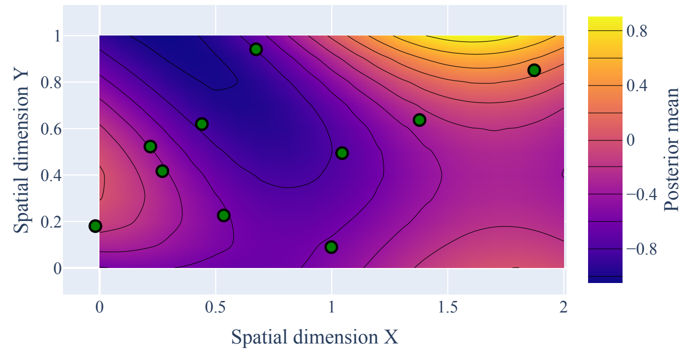

Fig. 5. GPR posterior mean values (represented by contour levels) using observed training positions (indicated by green dots).

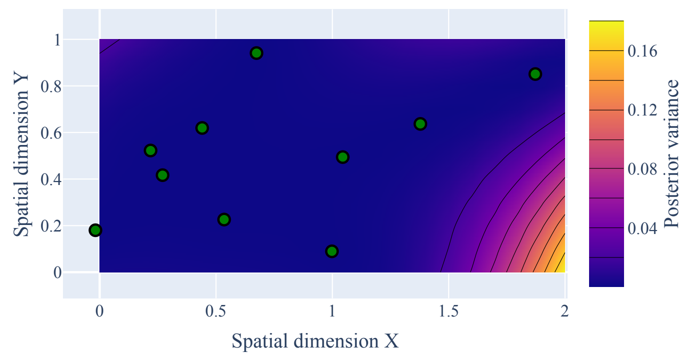

Fig. 6. GPR posterior variance (represented by contour levels) using observed training positions (indicated by green dots).

To account for the training positions uncertainty, we simulated the MC-GPR method according to (29) using $s = 100$ training positions samples generated according the posterior distribution of the training positions with uninformative prior. The resulting posterior mean is depicted in Figure 7 and posterior variance in Figure 8.

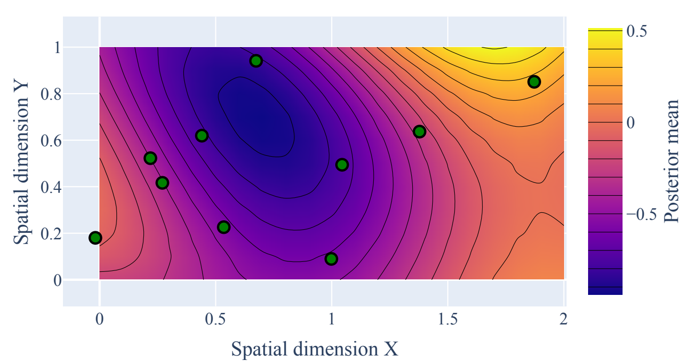

Fig. 7. MC-GPR posterior mean values (represented by contour levels) using observed training positions (indicated by green dots).

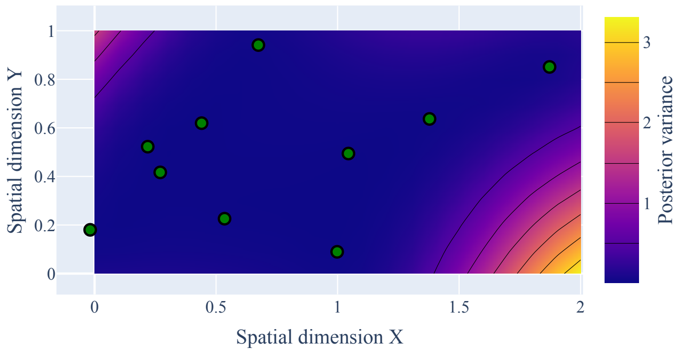

Fig. 8. MC-GPR posterior variance (represented by contour levels) using observed training positions (indicated by green dots).

VII. Results

As can be seen in the comparison of the true spatial function value in Figure 1 and its GPR estimate using the true training positions in Figure 3, this approach results in a high-quality estimation with low estimation uncertainty (variance), which can be seen in Figure 4. On the other hand, using the GPR approach in combination with uncertain (observed) training position without accounting for uncertainty results in a lower-quality estimate, as can be seen in Figure 5 combined with low estimation uncertainty (variance), which may be described as being unreasonably confident about the estimates. The MC-GPR approach accounting for uncertain training positions on the other hand resulted in better-quality estimate as can be seen in Figure 7 combined with higher estimation uncertainty (variance), which is a reasonable outcome of adding uncertainty to the inputs.

To quantitatively evaluate the estimation quality of the different approaches we performed 100 simulation runs, where each simulation run included sampling new GP realization, new training positions, new observation noises and new MC training positions samples. Evaluating the RMSE over all the simulation runs and all the test positions for different GPR methods resulted in Table I. Here we see that the best results were achieved by the standard GPR using known training positions. The worst results were attained by the GPR method using observed training positions naively. Compared to that, a significant performance improvement is achieved by the MC-GPR method, which accounts for training positions uncertainty via MC sampling.

TABLE I
RMSE of the considered GPR methods.

GPR method	RMSE
GPR, true training positions	0.1228
GPR, observed training positions	0.4027
MC-GPR	0.2810

VIII. Conclusion

We presented a derivation of the GPR operating on uncertain training positions. Having obtained an expression without closed-form solution, we presented its approximation using MC samples, which results in the MC-GPR method, which accounts for uncertain training positions. We further developed a simulation environment to compare the analyzed spatial function estimation approaches and to verify the performance improvements. In the individual simulation runs, it was demonstrated that MC-GPR method provides posterior distribution with higher variance than the usage of standard GPR with uncertain inputs, which corresponds to the increased uncertainty of the provided estimates and shows that the standard GPR is unreasonably confident about the provided estimates. Moreover, evaluating a higher number of simulation runs in the selected scenario setup and comparing the average RMSE of the estimates shows that the MC-GPR method achieves a significant improvement over the standard GPR operating on uncertain training positions. On the other hand, the performance of MC-GPR is still inferior to the standard GPR operating on known training positions and the performance gain comes at the cost of increased computation complexity linear with the number of MC samples. This makes the GPR with uncertain training positions an open problem with plausible improvements, which may be inspired by [6], [7], [2], [8].

References

[1] 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, Jan. 2013.

[2] 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.

[3] C. E. Rasmussen and C. K. I. Williams, *Gaussian processes for machine learning* (Adaptive computation and machine learning), en. Cambridge, Mass: MIT Press, 2006, OCLC: ocm61285753.

[4] C. M. Bishop, *Pattern recognition and machine learning* (Information science and statistics), en. New York: Springer, 2006.

[5] M. Ptáček, *Publications*, original-date: 2023-02-21T09:41:21Z, Mar. 2023. [Online]. Available: https://github.com/MarPta/Research (visited on 03/09/2023).

[6] A. Girard, “Approximate Methods for Propagation of Uncertainty with Gaussian Process Models,” en, p. 167,

[7] D. Cervone and N. S. Pillai, “Gaussian Process Regression with Location Errors,” *arXiv:1506.08256 [math, stat]*, Jun. 2015, arXiv: 1506.08256. [Online]. Available: http://arxiv.org/abs/1506.08256 (visited on 03/08/2021).

[8] M. Fröhle, L. S. Muppirisetty, and H. Wymeersch, “Channel gain prediction for multi-agent networks in the presence of location uncertainty,” in *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Mar. 2016, pp. 3911–3915.