Heteroscedastic Gaussian Processes and Random Features: Scalable Motion Primitives with Guarantees

Anonymous Author(s)
Affiliation
Address
email

Abstract: Heteroscedastic Gaussian processes (HGPs) are kernel-based, non-parametric models that can be used to infer nonlinear functions with time-varying noise. In robotics, they can be employed for learning from demonstration as motion primitives, i.e. as a model of the trajectories to be executed by the robot. HGPs provide variance estimates around the reference signal modeling the trajectory, capturing both the predictive uncertainty and the motion variability. However, similarly to standard Gaussian processes they suffer from a cubic complexity in the number of training points, due to the inversion of the kernel matrix. The uncertainty can be leveraged for more complex learning tasks, such as inferring the variable impedance profile required from a robotic manipulator. However, suitable approximations are needed to make HGPs scalable, at the price of potentially worsening the posterior mean and variance profiles. Motivated by these observations, we study the combination of HGPs and random features, which are a popular, data-independent approximation strategy of kernel functions. In a theoretical analysis, we provide novel guarantees on the approximation error of the HGP posterior due to random features. Moreover, we validate this scalable motion primitive on real robot data, related to the problem of variable impedance learning. In this way, we show that random features offer a viable and theoretically sound alternative for speeding up the trajectory processing, without sacrificing accuracy.

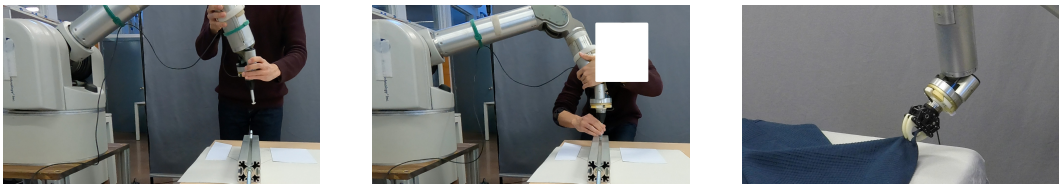
Keywords: Gaussian process regression, random features, motion primitives.

1 Introduction

Learning from demonstration (LfD) is a broadly used technique to transfer skills from humans to robots in a flexible and intuitive way [1]. Within the context of robotics manipulation, LfD consists of recording the movement of an arm performing a specific task multiple times. This data is then used to fit a model of the trajectory, called a *motion primitive* in this context, which allows the robot to reproduce the skill of interest. One popular way to achieve this objective is by means of the so-called *Gaussian process* (GP) regression [2]. Being a Bayesian model, a GP naturally provides a time-varying reference signal to be followed by the robot (the *GP posterior mean*), and an uncertainty quantification (the *GP posterior variance*) around the reference signal.

When used for LfD, GPs are usually *heteroscedastic* (HGP), i.e., the variance of the noise corrupting the recorded trajectories is not constant [3, 4]. As discussed by Arduengo et al. [5], such a time-varying noise variance is a key asset for modeling motion variability when HGPs are used as motion primitives. Moreover, HGPs are able to quantify the input-dependent uncertainty on the predictions, which is relevant when the testing points are far from the training data due to gaps in the human demonstrations. Such a rich uncertainty model is encoded in the posterior variance of the HGP. From a robotics perspective, the posterior variance is particularly relevant in the context of *variable impedance control* (VIC), that is, when the robotic manipulator is controlled with a time-dependent stiffness. In such a context, the uncertainty can be used to compute time-varying

Submitted to the 7th Conference on Robot Learning (CoRL 2023). Do not distribute.



(a) Free motion. (b) Assembly task. (c) Bed-making skill.

Figure 1: The tasks involved in our experiments. In the first two pictures, the human is guiding the robot to perform: a free motion above a metal piece; the insertion of the end-effector in the trail inside the piece. In the last picture, the robot is pulling a bed sheet to remove the wrinkles.

profiles from the human demonstrations [6, 7]. While this approach has been shown to be effective, it has severe limitations in presence of constraints on the manipulator's motion, as discussed by Peternel et al. [8] and Doornebosch et al. [9]. To overcome these issues, Caldarelli et al. [10] proposed an *active learning* algorithm of the stiffness using the posterior variance as initialization.

While being attractive for the aforementioned reasons, GP regression suffers from a cubic scaling in the number of training samples as it requires an inversion of the kernel matrix. For this reason, several approximation techniques have been deployed, that allow for a *linear* scaling in the number of observations. In particular, these approximations can be grouped into variational ones [11, 12, 13] and the spectral ones. The former methods are data-dependent and rely on a variational approximation of the posterior distribution of the GP, based on the choice of *inducing points*, i.e. of a subset of the data summarizing the whole training set. The latter methods, on the other hand, aim at approximating the GP's kernel function in the posterior expression directly. Feature approximations are potentially data-independent, and can rely, e.g., on a polynomial approximation of the kernel [14, 15], a truncation of the kernel's spectrum [16], or a randomized approximation of an integral representation of the kernel function. The latter technique, called *random features* (RFs) approximation [17], is a broadly used algorithm that can be combined with GPs [18, 19, 20, 21, 22]. Considering the VIC scenario previously described, these approximations should not deteriorate the posterior mean and variance as these quantities fully specify the robot's motion and stiffness. In this sense, a GP is an appealing motion primitive, as its approximations can be rigorously analysed [16, 23, 24].

Contributions In this work, we study the combination of vector-valued HGPs with RFs for fast trajectory processing in LfD. Motivated by the VIC setup, in which the posterior mean and variance are of paramount importance, we perform a theoretical analysis and propose novel bounds on the approximation error of the posterior mean and variance of the HGP. This is achieved by leveraging the deep connection between approximate GP regression and its frequentist counterpart, namely *kernel ridge regression* with RFs [25]. In our experiments with real robot data, we further assess the quality of the approximation for the three VIC tasks described by Caldarelli et al. [10], and shown in Figure 1. Overall, we demonstrate that the combination of RFs and HGPs is theoretically sound, and significantly improves the running time of the motion primitive fitting, without sacrificing accuracy.

Layout The structure of the paper is as follows. In Section 2 we describe the main background of our work, related to HGPs and RFs. In Section 3, we detail the combination of HGPs with RFs, motivated from a robotics perspective. In Section 4, we report the results of our theoretical analysis. The experiments are reported in Section 5, while Section 6 reports a discussion of the main limitations of our work. Lastly, the conclusions are reported in Section 7.

2 Background

HGP regression for trajectory encoding Let $\mathcal{X}$ be an input space, and $x : \mathcal{X} \to \mathbb{R}$ be a scalar-valued function. In LfD, the function x depends on time, i.e., $\mathcal{X} = \mathbb{R}_{\geq 0}$. A GP [2] specifies a prior distribution over the function x, which depends on a mean function $\mu : \mathcal{X} \to \mathbb{R}$ and a kernel $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$. We say that x follows a GP with mean $\boldsymbol{\mu}$ and kernel function k if for any vector of time-steps $\mathbf{t} \in \mathbb{R}^n$, the vector $\mathbf{x}(t)$ of evaluations of x at $\mathbf{t}$ follows the multivariate normal distribution

$\mathbf{x}(\mathbf{t}) \sim \mathcal{N}(\boldsymbol{\mu}(\mathbf{t}), K)$, where $K \in \mathbb{R}^{n \times n}$ is defined as $K_{i,j} = k(t_i, t_j)$. The mean function is usually assumed to be 0, as the function values are standardized. The kernel function depends on a set of hyperparameters, $\boldsymbol{\theta}$, which can be fixed or inferred from data. In LfD, the time-dependent trajectory to be encoded consists of d degrees of freedom (DOFs). If all DOFs are fully observed, each of them is usually modeled by an independent GP [26]. This type of estimation can be linked to a reproducing kernel Hilbert space (RKHS) of vector-valued functions [27]. We assume to have access to $\mathbf{y} \in \mathbb{R}^n$, a vector of noisy measurements of $\mathbf{x}(\mathbf{t})$. As shown by Arduengo et al. [26], assuming a constant noise variance in LfD might severely limit the quality of the posterior prediction. Therefore, the noise is time-dependent, turning the GP into an HGP. The posterior distribution of an HGP, conditioned on the noise variance values at the training points and the function's observations, is analytically available. Let t^* be a testing point, and $\mathbf{k}_{t^*} := [k(t_1, t^*), \ldots, k(t_n, t^*)]^T \in \mathbb{R}^n$. Lastly, let $\Sigma_{\text{noise}} \in \mathbb{R}^{n \times n}$ be the noise variance at the training points, and $\sigma^2_{\text{noise},t^*}$ be the noise variance at the testing point. The posterior mean and variance of the associated HGP are, respectively, [26]

$$\mu_{\text{post}}(t^*) = \mathbf{k}_{t^*}^T (K + \Sigma_{\text{noise}})^{-1} \mathbf{y}, \tag{1}$$

$$\sigma^2_{\text{post}}(t^*) = k(t^*, t^*) + \sigma^2_{\text{noise},t^*} - \mathbf{k}_{t^*}^T (K + \Sigma_{\text{noise}})^{-1} \mathbf{k}_{t^*}. \tag{2}$$

Random features Let $(\Omega, \mathcal{A}, \pi)$ be a probability space over the sample space Ω, and let $\psi : \Omega \times \mathcal{X} \to \mathbb{R}$. Random features (RFs) are a class of randomized methods that can be used to approximate a kernel function admitting an integral representation of the form

$$k(t, t') = \int_\Omega \psi(\omega, t) \psi(\omega, t') d\pi(\omega) \tag{3}$$

by dicretizing it. This is the case for standard kernels such as the ones belonging to the Matérn family or the radial-basis-function (RBF) kernel, as discussed by Rahimi and Recht [17]. The expectation above is approximated by sampling m vectors $(\omega_j)_{1 \leq j \leq m} \sim \pi(\omega)$. For $\tilde{\boldsymbol{\phi}}(t) := \frac{1}{\sqrt{m}} [\psi(\omega_1, t), \ldots, \psi(\omega_m, t)]^T$, the kernel is approximated as $\tilde{k}(t, t') := \tilde{\boldsymbol{\phi}}(t)^T \tilde{\boldsymbol{\phi}}(t')$. For instance for the case of a positive definite stationary kernel $k(t, t') = k(t - t')$, Bochner's theorem ensures that k has a non-negative Fourier transform, which can thus be used in place of π while defining ψ to be a trigonometric function [17]. Rahimi and Recht [17] show that, for the RBF kernel $k(t, t') = \sigma^2_{\text{RBF}} \exp\left(-\|t - t'\|^2 / (2l^2)\right)$, the values of ω are s.t. $\omega_i \sim (2\pi l^{-2})^{-\frac{m}{2}} \exp\left(-l^2 |\omega|^2 / 2\right)$. Then, for b uniformly sampled from $[0, 2\pi)$, they define $\psi(\omega, t) := \sigma_{\text{RBF}} \cos(\omega t + b)$. To retrieve closed-form expressions for the posterior mean and variance of an HGP approximated with RFs, we can define the matrix $\tilde{K} \in \mathbb{R}^{n \times n}$ with entries $\tilde{K}_{i,j} = \tilde{k}(t_i, t_j)$. For any testing point t^* we define $\tilde{\mathbf{k}}_{t^*} = [\tilde{k}(t_1, t^*), \ldots, \tilde{k}(t_n, t^*)]^T \in \mathbb{R}^n$. Lastly, let Σ_{noise} and $\sigma^2_{\text{noise},t^*}$ be as in the previous paragraph. The posterior mean and variance of the associated HGP are, respectively [26]

$$\mu_{\text{post}}(t^*) = \tilde{\mathbf{k}}_{t^*}^T (\tilde{K} + \Sigma_{\text{noise}})^{-1} \mathbf{y}, \tag{4}$$

$$\sigma^2_{\text{post}}(t^*) = \tilde{k}(t^*, t^*) + \sigma^2_{\text{noise},t^*} - \tilde{\mathbf{k}}_{t^*}^T (\tilde{K} + \Sigma_{\text{noise}})^{-1} \tilde{\mathbf{k}}_{t^*}. \tag{5}$$

We can observe that these equations are structurally the same as Equations (1) and (2), with the approximated kernel $\tilde{k}$ replacing the exact kernel k.

3 Scalable Motion Primitive Fitting with RFs and HGPs

Within the context of VIC, Caldarelli et al. [10] have recently proposed an interactive learning pipeline for inferring the stiffness profiles over time associated to a given task. After collecting a set of task demonstrations with kinesthetic teaching [28], the demonstrations are temporarily aligned, and the samples that constitute them are merged into a unique training set, independently of the trial in which they were recorded. This training set is processed with an HGP, whose posterior mean is used as a reference signal for the task being executed in Cartesian space. Moreover, to estimate the desired stiffness, the posterior mean is intentionally corrupted by a series of additive perturbations having peaks in the regions of low variance. Such low values might either indicate the need of a high stiffness, or the presence of a constraint, potentially requiring a compliant behavior. The perturbed trials are then executed by the robot under human supervision, and human corrections to the perturbations are used to infer the stiffness. Computing an accurate estimate of the variance is critical for

this procedure, as it serves as initialization of the perturbation routine. Moreover, the posterior mean needs to be accurate, as it is the trajectory executed by the robot when performing a task. On the other hand, the computation of the posterior mean and variance should be efficient considering that the algorithm involves human-robot interaction, and the user experience can significantly benefit from a faster processing algorithm, especially with long demonstrations and online learning.

Scalable HGPs In this work, we focus on the first step of the learning pipeline presented by Caldarelli et al. [10], i.e. the motion primitive fitting. To make it scalable, we study the combination of the HGP with RFs (RF-HGP). Fixing the dimensionality of the RF vector, this approach allows for linear complexity in the number of samples when computing the posterior distribution of the HGP, as the posterior expressions in Equations (4) and (5) can be simplified by the well-known *Woodbury identity* [29]. Such an identity allows to perform matrix inversion in $\mathcal{O}(m^3)$, $m \ll n$ being the dimension of the RF vector, as shown in Appendix A. Moreover, the kernel approximation is data-independent, as the functions $\psi(\omega_j, \cdot)$ from Section 2 can be computed prior to seeing any data.

Workflow with hyperparameters tuning For our experiments in Section 5, we will consider two different settings. In the first scenario, the hyperparameters $\boldsymbol{\theta}$ of the kernel function (which typically contains at least a lengthscale parameter), as well as the noise variance values at the training points, Σ_{noise}, and at a testing point, $\sigma^2_{\text{noise},t^*}$, are fixed *a priori* (**oracle setup**). In the second scenario, $\boldsymbol{\theta}$, Σ_{noise}, and $\sigma^2_{\text{noise},t^*}$ are directly inferred from data (**heuristic setup**). In the first case, the error introduced by the RF approximation can be rigorously quantified, as we will show in Section 4. In the second case, convergence to the exact HGP can be attained empirically, and a suitable inference algorithm needs to be used. In this work, we consider the expectation-maximization (EM) training proposed by Kersting et al. [4], adapted to the RF-HGP. The algorithm comprises the following steps, which are performed independently for each DOF (all the GPs involved take time as input):

1. train a first homoscedastic GP ($\mathcal{GP}_1$), approximated with RFs (RF-GP), on the demonstrations' data with a maximum likelihood estimate (MLE), to retrieve $\boldsymbol{\theta}$;
2. compute the mean-squared-error (MSE) between the posterior mean of $\mathcal{GP}_1$ and the demonstrations' data, at each time-step;
3. train a second RF-GP ($\mathcal{GP}_2$) by means of MLE, using the MSE as training data; $\mathcal{GP}_2$ is a surrogate model of the time-dependent noise variance function;
4. compute the posterior mean and variance of a new RF-HGP ($\mathcal{GP}_3$) as of Equations (4) and (5), with the kernel hyperparameters $\boldsymbol{\theta}$ from step 1 and the noise variance values in Σ_{noise} and $\sigma^2_{\text{noise},t^*}$ given by the posterior mean of $\mathcal{GP}_2$, evaluated at the training and testing time-steps respectively;
5. compute MSE between the posterior mean of $\mathcal{GP}_3$ and the demonstrations, at each time-step;
6. repeat from step 3 until convergence or until the maximum number of iterations is reached.

Both in the oracle and in the heuristic setup, the RF-HGP's posterior distribution is computed at a set of testing time-steps $\mathcal{T} = \{t_1^*, \dots, t_T^*\}$, which are fixed *a priori* based on the desired task duration.

4 Quantification of the Approximation Errors

In this section, we derive error bounds on posterior mean and variance for the RF-HGP, at the test points in $\mathcal{T}$. These results guarantee that RFs are a suitable approximation strategy, that can be used in robotics scenarios where both a reference signal and uncertainty are needed, such as the VIC problem presented in Section 3. The proofs, reported in Appendix B.2 and Appendix B.3, rely on techniques developed by Rudi and Rosasco [25]. As a start, we will introduce some assumptions.

4.1 Assumptions

In the following, we set $\psi_j(\cdot) \coloneqq \psi(\omega_j, \cdot)$, where the latter function is defined in Section 2. Moreover, we make the following assumptions.

Assumption 4.1. The kernel is bounded, i.e., $k(t, t') \leq \kappa$. Moreover, the kernel admits the integral representation of Equation (3), in terms of a suitable function $\psi(\omega, \cdot)$.

Assumption 4.2. The entries of the RF vector are bounded, i.e., $|\psi_j(t)| \leq \alpha, \forall j \in \{1, \dots, m\}, \forall t$.

Assumption 4.3. The noise variance values in the diagonal matrix Σ_{noise} are bounded from below, i.e. $\exists \gamma \in (0, 1]$ s.t. $(\Sigma_{\text{noise}})_{ii} \geq \gamma n, \ \forall i \in \{1, \dots, n\}$.

Assumption 4.1 is standard, and holds, e.g., for the RBF kernel ($\kappa = \sigma^2_{\text{RBF}}$). Assumption 4.2 holds, e.g., for the RFFs described in Section 2 (with $\alpha = \sigma_{\text{RBF}}$). Assumption 4.3 is obtained by replacing, e.g, $K + \Sigma_{\text{noise}}$ in Equations (1) and (2) with $K + (\Sigma_{\text{noise}} + \gamma n I)$, a common practice aimed at avoiding numerical instabilities in matrix inversion. Lastly, for the RBF kernel and RFFs, $\kappa = \alpha^2$.

4.2 Concentration of Posterior Mean and Variance

Under these assumptions, we can derive the two main results of this paper, bounding the deviation of the posterior mean and variance of our RF-HGP w.r.t. its exact counterpart.

Theorem 4.4. *Let $\delta \in (0, 1]$, and m be the dimension of the RF vector. Let $m \geq 8\left(\frac{1}{3} + \frac{\alpha^2}{\gamma}\right)\log(\frac{8\alpha^2}{\gamma\delta})$, and consider a vector-valued HGP with d independent components. Let $\boldsymbol{\mu}_{post}$ and $\tilde{\boldsymbol{\mu}}_{post}$ denote its exact and RF-approximated posterior mean as of Equations (1) and (4). Let $\nu = \max_{1 \leq i \leq d} \left\| \frac{1}{\sqrt{n}} \mathbf{y}_i \right\|$, where $\mathbf{y}_i$ denotes the observations of DOF i. Lastly, let $\mathcal{T}$ be a set of testing points. Under Assumptions 4.1 to 4.3, with probability at least $1 - d\delta$, $\forall t^* \in \mathcal{T}$, it holds that*

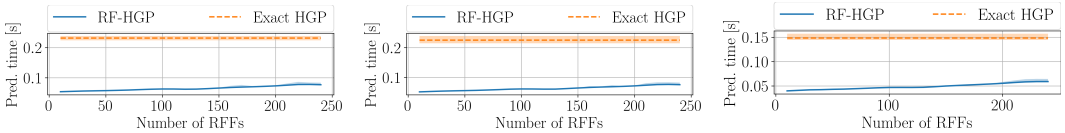
$$\|\boldsymbol{\mu}_{post}(t^*) - \tilde{\boldsymbol{\mu}}_{post}(t^*)\|_2 \leq \nu\alpha^2 \sqrt{\frac{2d \log \frac{2|\mathcal{T}|n}{\delta}}{m\gamma^2}} + \frac{\sqrt{2d}\kappa\nu}{\sqrt{\gamma}} \left[\frac{2 \log \frac{8\kappa^2}{\gamma\delta}(1 + \alpha^2/\gamma)}{3m} + \alpha \sqrt{\frac{2 \log \frac{8\kappa^2}{\gamma\delta}}{\gamma m}} \right].$$

Theorem 4.5. *Under the assumptions of Theorem 4.4, denoting $\boldsymbol{\sigma}^2_{post}$ and $\tilde{\boldsymbol{\sigma}}^2_{post}$ respectively the exact and RF-approximated posterior variances as of Equations (2) and (5), it holds with probability at least $1 - d\delta$, $\forall t^* \in \mathcal{T}$,*

$$\begin{aligned}\|\boldsymbol{\sigma}^2_{post}(t^*) - \tilde{\boldsymbol{\sigma}}^2_{post}(t^*)\|_2 \leq & \alpha^2 \sqrt{\frac{2d \log \frac{2|\mathcal{T}|}{\delta}}{m}} + \left(\frac{\kappa\alpha^2}{\sqrt{\gamma}} + \frac{\alpha^4}{\gamma} \right) \sqrt{\frac{2d \log \frac{2|\mathcal{T}|n}{\delta}}{m}} \\ & + \frac{\alpha^3\sqrt{2d}}{\sqrt{\gamma}} \left[\frac{2 \log \frac{8\kappa^2}{\gamma\delta}(1 + \alpha^2/\gamma)}{3m} + \alpha \sqrt{\frac{2 \log \frac{8\kappa^2}{\gamma\delta}}{\gamma m}} \right].\end{aligned}$$

Our bounds show that both the mean and the variance errors for a vector-valued RF-HGP scale as $\mathcal{O}(m^{-1/2})$. Moreover, our results cover the RF-based approximation of the homoscedastic GP as a special case taking $\Sigma_{\text{noise}} = \beta I$ for some $\beta \geq 0$. Our results ensure that RFs are a viable data-independent approximation method, and match the rates of other data-dependent strategies, such as the well-studied *Nyström method* [30] for GPs, as shown by Lu et al. [31]. Moreover, if the testing points are not fixed, one could extend the analysis so as to account for the uniform approximation provided, e.g., by RFFs, as commented in Appendix C.

Feature approximations Some theoretical properties of feature approximations for homoscedastic, scalar-valued GPs have been explored in previous works. Särkkä and Piché [14] propose a feature approximation based on different polynomial approximations of the RBF kernel function. They are able to prove uniform convergence of the kernel, mean and variance, but without convergence rates. Moreover, Solin and Särkkä [15] propose to compute the features based on the Fourier transform of the Laplace operator. They prove convergence of the kernel function values and convergence of the posterior mean and variance, but they have a dependency on the size of the domain of the expansion of the Laplace operator, which cannot grow faster than the number of features, as discussed by Angelis et al. [32]. RFs are also widely used in the Bayesian optimization setting, as they allow to provide scalable algorithms with theoretical guarantees [16, 24]. In this scenario, the *uniform convergence* of the approximate kernel function is of interest as the optimization algorithms may require sampling points in the whole domain of the function being optimized. Compared to our work, these bounds have a worse dependency on the number n of training samples (which is not due to the uniformity of the bounds). For instance, Mutny and Krause [16, Proof of Th. 5 and Prop. 1] in the context of homoscedastic GP approximated with deterministic features provide bounds of the form



(a) Free motion. (b) Assembly task. (c) Bed-making skill.

Figure 2: Time taken to compute the posterior distribution with an exact HGP and an RF-HGP, with demonstrations of three different robotic tasks, in an oracle setup. Median, the 15th and 85th percentiles across all DOFs, 50 seeds.

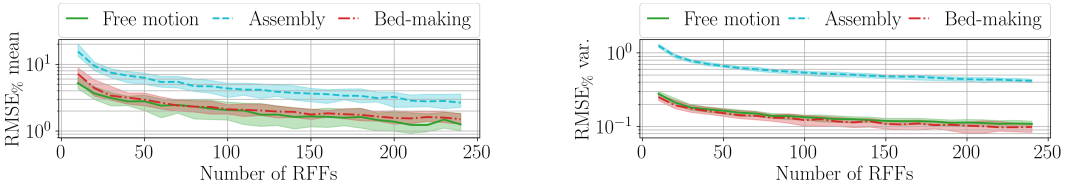


Figure 3: RMSE between the posterior means (left) and posterior variances (right) of an exact and and approximated HGP, in an oracle setup, for three different tasks. Median, 15th and 85th percentiles across all DOFs, 50 seeds.

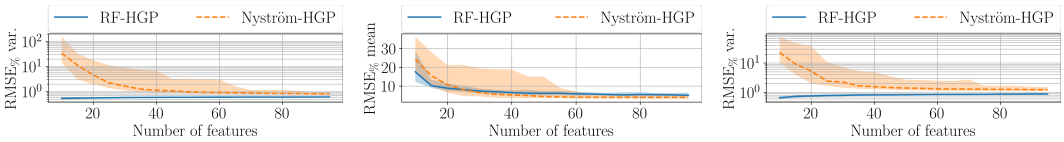
$\sup_{t^*} |\mu_{\text{post}}(t^*) - \tilde{\mu}_{\text{post}}(t^*)| \lesssim \epsilon \nu n^2 \sigma^{-2}$ and $\sup_{t^*} |\sigma^2_{\text{post}}(t^*) - \tilde{\sigma}^2_{\text{post}}(t^*)| \lesssim \epsilon n^3 \sigma^{-2}$ where ϵ denotes the accuracy of a uniform bound on the kernel approximation, i.e. $\sup_{x,y} |k(x,y) - \tilde{k}(x,y)| \leq \epsilon$. In our setting, when $d = 1$ and in the homoscedatic setting $\Sigma_{\text{noise}} = \sigma^2 I$ and $\gamma = \sigma^2/n$ (cf. Assumption 4.3) we obtain a dependence in n of order $\mathcal{O}(n^{3/2}\log(n))$ both for the posterior mean and variance which strictly improves on these bounds.

Variational approximations Considering variational approximations, such as the variational Fourier features [13] or SGPR [12], one fundamental convergence result for the Kullback-Leibler (KL) divergence between the variational and exact posterior of the GP was proven by Burt et al. [23]. Moreover Burt et al. [33, Proposition 1] showed that the error on the KL divergence implies convergence of the variational posterior means and variances, however the bound that they derive depends on the value of the exact posterior variance, which makes the comparison with our result difficult.

5 Empirical Evaluation

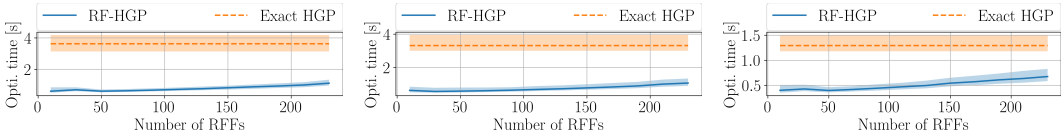
In this section, we empirically validate the RF-HGP. The implementation is built on the package `gpflow` by Matthews et al. [34][1]. We choose RFFs to approximate the RBF kernel, as reported in Section 2. In this work, we do not train the RFF parameters ω's, as it is prone to overfitting [18, 35], and annuls the Monte-Carlo interpretation of RFFs introduced in Section 2. To assess the correctness of the GP approximation, we process real demonstrations of different robotic tasks, obtained by means of kinesthetic teaching with a 7-DOF Barrett WAM manipulator. The trajectories are recorded while performing a movement in free Cartesian space, an assembly task, and a bed-making skill [10]. These trajectories are particularly interesting from the VIC point of view, and summarized in Figure 1. The first task exhibits varying boundaries in which the manipulator can move, while the latter two involve physical constraints on the robot's motion. The assembly task requires a motion with low stiffness when the pieces being mounted are in contact. On the other hand, the bed-making skill requires the robot to stiffen up to remove wrinkles from the sheet's surface, in spite of it opposing the motion of the robot's end-effector. Each task uses a different number of human demonstrations, namely 6, 7 and 5. While the number of demonstration is relatively small, the total number of training points is in the order of 10^3, justifying the need for scalable GPs. Moreover, the number of samples might increase for longer or denser trajectories. We consider an oracle setup in Section 5.1, where all hyperparameters and noise variances are assumed to be known, and then evaluate in Section 5.2 the heuristic method presented in Section 3.

[1]The code for the experiments is open-source and available at *LINK REMOVED*.



(a) Free motion. (b) Assembly task. (c) Bed-making skill.

Figure 4: Incremental learning with missing chunks of data, in an oracle setup. Starting points of the missing chunks are sampled uniformly. RF-HGP vs a Nyström approximation with fixed centers, to achieve the same complexity as RFs. Median, 15th and 85th percentiles across all DOFs, 50 seeds.



(a) Free motion. (b) Assembly task. (c) Bed-making skill.

Figure 5: The time taken to perform an MLE with an exact HGP and an RF-HGP, with demonstrations of three different robotic tasks, in a heuristic setup. We show the median, the 15th and 85th percentiles, across all DOFs and 50 seeds.

5.1 Oracle Setup

As a start, we consider the ideal case in which the GP kernel hyperparameters $\boldsymbol{\theta}$ are known, and the noise variance values Σ_{noise}, $\sigma^2_{\text{noise},t^*}$ are provided by an oracle, as discussed in Section 3. This experiment is useful to assess how the prediction capability of RF-HGP deteriorates due to the kernel spectral approximation, in view of the theoretical analysis carried out in Section 4. The oracle is given by an exact HGP trained with the EM algorithm by Kersting et al. [4] as mentioned in Section 3, for 15 iterations per DOF.

Offline learning In this setup, we are interested in comparing the prediction times (Figure 2) and accuracy (Figure 3) of exact and approximated GP regression. By prediction time, we refer here to the time taken to compute the posterior means and posterior variances at the testing points. The human demonstrations are processed all together, in an offline fashion. The prediction times, as shown in Figure 2, indicate that the RF-HGP is more efficient, as expected. Moreover, to asses the predictive accuracy, we compute the normalized root-mean-squared-error (RMSE) between the posterior means and between the posterior variances of the HGP and RF-HGP:

$$\text{RMSE}_{\%\text{mean}} = \sqrt{\frac{1}{d}\sum_{i=1}^{d}\frac{\sum_{t\in\mathcal{T}}(\tilde{\mu}_{\text{post},i}(t)-\mu_{\text{post},i}(t))^2}{\sum_{t\in\mathcal{T}}\mu_{\text{post},i}(t)^2}}\cdot 100. \tag{6}$$

The $\text{RMSE}_{\%}$ on the variance is computed in the same way. This experiment allows us to observe that the error rate follows the expected rate in $\mathcal{O}(m^{-1/2})$, as shown in Figure 3, indicating that RFs can be used to speed up the inference without sacrificing accuracy in the posterior's calculation. We can also observe that although all errors decrease with an increasing number of features, the assembly task exhibits the largest error among the three tasks. This is likely to be due to the RF approximation being challenged by the sharp changes in the task variability.

Incremental learning RFs are data-independent, and they can be used to perform incremental learning of the posterior distribution of the HGP [36]. In an oracle setup, the posterior distribution can be updated every time a new demonstration is gathered, as described in Appendix D. Such an approach is also possible, e.g., with the Nyström approximation of the kernel function, provided that the points summarizing the dataset, i.e., the Nyström centers, are sampled from the first demonstration and not updated afterwards, as doing so would imply re-computing the correlation matrices between the centers and the whole set of training points, which has a linear complexity in the size of the training set. Keeping a fixed set of Nyström centers is not an issue in general for LfD, as the domain is scalar and all demonstrations are temporally aligned on the time interval $[0, 1]$,

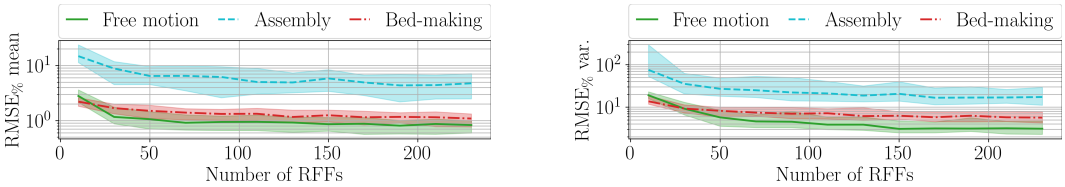


Figure 6: RMSE between the posterior means (left) and posterior variances (right) of an exact and an approximated HGP, in heuristic setup, for three different tasks. Median, the 15th and 85th percentiles, across all DOFs and using 50 seeds.

and thus the first demonstration already covers the whole domain. However, this fact does not hold if, for any reason, portions of the human demonstrations are missing or invalid. In this case, the initialization of the centers might be poor, and the posterior calculation spoiled. This issue is shown in the examples of Figure 4, where chunks of 60 observations, sampled uniformly at random, were removed from each demonstration. RFFs offer a more reliable option, in such a scenario.

5.2 Heuristic Setup

Having validated our theoretical contributions, we are now interested in the more realistic setting described in Section 3, in which the kernel hyperparameters and noise variance are unknown and need to be heuristically estimated from data. Note that this step can serve as preliminary stage with a few demonstrations before applying, e.g., the incremental learning described in the previous section. In this case, the RF-HGP would be trained with the EM method described in Section 3.

Results Figure 5 reports the time taken to complete an MLE for retrieving the GP hyperparameters, both in the exact and in the RFF setup. With 150 features, the optimization time is approximately 23% of the exact counterpart for the assembly task, 23% for the free motion, and 42% for the bed-making skill. Considering the errors reported in Figure 6, we observe that convergence to the exact HGP posterior can be heuristically attained in the worst possible scenario of having no knowledge about the necessary GP priors. Again, the largest errors are attained by the assembly task, as discussed for the oracle setup. Note that variational methods solve more complex optimization problems, as they usually optimize the location of the inducing points, and therefore a comparison with such methods would be unfair as they have a worse complexity.

6 Limitations

The assumptions behind our theoretical analysis were stated in Section 4.1. Moreover, as discussed in Section 3, our theoretical analysis focuses on an oracle setup. This type of scenario is standard in kernel theory [37, 25, 31], as it allows to decouple the error due to the kernel approximation from the uncertainty surrounding the GP hyperparameters. If this oracle setup does not hold, the HGP training method is heuristic, and uses two approximate GPs iteratively in an EM fashion. The errors displayed in the heuristic experiments of Section 5.2 may change with a different training algorithm.

7 Conclusion

In this work, we have studied the combination of heteroscedastic Gaussian processes and random features, used as scalable motion primitives in the context of learning from demonstration. Motivated by the variable impedance learning problem, we derived novel upper bounds on the approximation error, induced by random features, on the posterior mean and variance of a Gaussian process. Moreover, we have validated this approximate motion primitive w.r.t. relevant tasks for variable impedance control of robotic manipulators, namely, a motion in free Cartesian space, an assembly task, and a bed-making skill. Our theoretical and empirical results demonstrate that random features are a theoretically sound approximation method, that can be used to speed up the motion primitive fitting without sacrificing accuracy. Moreover, we have shown that random features are well suited to incremental learning from demonstration, thanks to their data-independent nature.

References

[1] A. G. Billard, S. Calinon, and R. Dillmann. Learning from humans. *Springer handbook of robotics*, pages 1995–2014, 2016.

[2] C. K. Williams and C. E. Rasmussen. *Gaussian processes for machine learning*, volume 2. MIT press Cambridge, MA, 2006.

[3] P. Goldberg, C. Williams, and C. Bishop. Regression with input-dependent noise: A Gaussian process treatment. *Advances in neural information processing systems*, 10, 1997.

[4] K. Kersting, C. Plagemann, P. Pfaff, and W. Burgard. Most likely heteroscedastic Gaussian process regression. In *International conference on Machine Learning (ICML)*, pages 393–400. PMLR, 2007.

[5] M. Arduengo, A. Colomé, J. Lobo-Prat, L. Sentis, and C. Torras. Gaussian-process-based robot learning from demonstration. *Journal of Ambient Intelligence and Humanized Computing*, pages 1–14, 2023.

[6] S. Calinon, I. Sardellitti, and D. G. Caldwell. Learning-based control strategy for safe human-robot interaction exploiting task and robot redundancies. In *IEEE International Conference on Intelligent Robots and Systems (IROS)*, pages 249–254, 2010.

[7] D. Parent, A. Colomé, and C. Torras. Variable impedance control in cartesian latent space while avoiding obstacles in null space. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 9888–9894, 2020.

[8] L. Peternel, T. Petrič, and J. Babič. Robotic assembly solution by human-in-the-loop teaching method based on real-time stiffness modulation. *Autonomous Robots*, 42:1–17, 2018.

[9] L. M. Doornebosch, D. A. Abbink, and L. Peternel. The force-feedback coupling effect in bilateral tele-impedance. In *IEEE International Conference for Biomedical Robotics and Biomechatronics (BioRob)*, pages 152–157, 2020.

[10] E. Caldarelli, A. Colomé, and C. Torras. Perturbation-based stiffness inference in variable impedance control. *IEEE Robotics and automation letters*, 7(4):8823–8830, 2022.

[11] H. Liu, Y.-S. Ong, X. Shen, and J. Cai. When Gaussian process meets big data: A review of scalable GPs. *IEEE Transactions on neural networks and learning systems*, 31(11):4405–4423, 2020.

[12] M. Titsias. Variational learning of inducing variables in sparse Gaussian processes. In *International Conference on Artificial intelligence and Statistics (AISTATS)*, pages 567–574. PMLR, 2009.

[13] J. Hensman, N. Durrande, A. Solin, et al. Variational Fourier features for Gaussian processes. *The Journal of Machine Learning Research*, 18(1):5537–5588, 2017.

[14] S. Särkkä and R. Piché. On convergence and accuracy of state-space approximations of squared exponential covariance functions. In *IEEE International Workshop on Machine Learning for Signal Processing (MLSP)*, 2014.

[15] A. Solin and S. Särkkä. Hilbert space methods for reduced-rank Gaussian process regression. *Statistics and Computing*, 30(2):419–446, 2020.

[16] M. Mutny and A. Krause. Efficient high dimensional Bayesian optimization with additivity and quadrature Fourier features. *Advances in Neural Information Processing Systems*, 31, 2018.

[17] A. Rahimi and B. Recht. Random features for large-scale kernel machines. *Advances in neural information processing systems*, 20, 2007.

[18] M. Lázaro-Gredilla, J. Quinonero-Candela, C. E. Rasmussen, and A. R. Figueiras-Vidal. Sparse spectrum Gaussian process regression. *The Journal of Machine Learning Research*, 11:1865–1881, 2010.

[19] G. V. Karanikolas, Q. Lu, and G. B. Giannakis. Online unsupervised learning using ensemble Gaussian processes with random features. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3190–3194, 2021.

[20] Y. Pan, X. Yan, E. A. Theodorou, and B. Boots. Prediction under uncertainty in sparse spectrum Gaussian processes with applications to filtering and control. In *International Conference on Machine Learning (ICML)*, pages 2760–2768. PMLR, 2017.

[21] K. D. Polyzos, Q. Lu, and G. B. Giannakis. Graph-adaptive incremental learning using an ensemble of Gaussian process experts. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5220–5224, 2021.

[22] K. Cutajar, E. V. Bonilla, P. Michiardi, and M. Filippone. Random feature expansions for deep Gaussian processes. In *International Conference on Machine Learning (ICML)*, pages 884–893. PMLR, 2017.

[23] D. Burt, C. E. Rasmussen, and M. Van Der Wilk. Rates of convergence for sparse variational Gaussian process regression. In *International Conference on Machine Learning (ICML)*, pages 862–871. PMLR, 2019.

[24] A. Dubey. No-regret algorithms for private Gaussian process bandit optimization. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, pages 2062–2070. PMLR, 2021.

[25] A. Rudi and L. Rosasco. Generalization properties of learning with random features. *Advances in neural information processing systems*, 30, 2017.

[26] M. Arduengo, A. Colomé, J. Borràs, L. Sentis, and C. Torras. Task-adaptive robot learning from demonstration with Gaussian process models under replication. *IEEE Robotics and automation letters*, 6(2):966–973, 2021.

[27] M. A. Alvarez, L. Rosasco, N. D. Lawrence, et al. Kernels for vector-valued functions: A review. *Foundations and Trends® in Machine Learning*, 4(3):195–266, 2012.

[28] D. Kushida, M. Nakamura, S. Goto, and N. Kyura. Human direct teaching of industrial articulated robot arms based on force-free control. *Artificial Life and Robotics*, 5:26–32, 2001.

[29] M. A. Woodbury. *Inverting modified matrices*. Statistical Research Group, 1950.

[30] C. Williams and M. Seeger. Using the Nyström method to speed up kernel machines. *Advances in neural information processing systems*, 13, 2000.

[31] X. Lu, A. Rudi, E. Borgonovo, and L. Rosasco. Faster kriging: Facing high-dimensional simulators. *Operations Research*, 68(1):233–249, 2020.

[32] E. Angelis, P. Wenk, B. Schölkopf, S. Bauer, and A. Krause. Sleipnir: Deterministic and provably accurate feature expansion for Gaussian process regression with derivatives. *arXiv preprint arXiv:2003.02658*, 2020.

[33] D. R. Burt, C. E. Rasmussen, and M. Van Der Wilk. Convergence of sparse variational inference in Gaussian processes regression. *The Journal of Machine Learning Research*, 21(1): 5120–5182, 2020.

[34] A. G. d. G. Matthews, M. Van Der Wilk, T. Nickson, K. Fujii, A. Boukouvalas, P. León-Villagrá, Z. Ghahramani, and J. Hensman. Gpflow: A Gaussian process library using tensorflow. *The Journal of Machine Learning Research*, 18(40):1–6, 2017.

[35] Y. Gal and R. Turner. Improving the Gaussian process sparse spectrum approximation by representing uncertainty in frequency inputs. In *International Conference on Machine Learning (ICML)*, pages 655–664. PMLR, 2015.

[36] R. Camoriano, S. Traversaro, L. Rosasco, G. Metta, and F. Nori. Incremental semiparametric inverse dynamics learning. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 544–550, 2016.

[37] A. Rudi, R. Camoriano, and L. Rosasco. Less is more: Nyström computational regularization. *Advances in Neural Information Processing Systems*, 28, 2015.