Deep state-space Gaussian processes

Zheng Zhao[1] · Muhammad Emzir[1] · Simo Särkkä[1]

© The Author(s) 2021

Abstract

This paper is concerned with a state-space approach to deep Gaussian process (DGP) regression. We construct the DGP by hierarchically putting transformed Gaussian process (GP) priors on the length scales and magnitudes of the next level of Gaussian processes in the hierarchy. The idea of the state-space approach is to represent the DGP as a non-linear hierarchical system of linear stochastic differential equations (SDEs), where each SDE corresponds to a conditional GP. The DGP regression problem then becomes a state estimation problem, and we can estimate the state efficiently with sequential methods by using the Markov property of the state-space DGP. The computational complexity scales linearly with respect to the number of measurements. Based on this, we formulate state-space MAP as well as Bayesian filtering and smoothing solutions to the DGP regression problem. We demonstrate the performance of the proposed models and methods on synthetic non-stationary signals and apply the state-space DGP to detection of the gravitational waves from LIGO measurements.

1 Introduction

Gaussian processes (GP) are popular models for probabilistic non-parametric regression, especially in the machine learning field (Rasmussen and Williams 2006). As opposed to parametric models, such as deep neural networks (Goodfellow et al. 2016), GPs put prior distributions on the unknown functions. As the mean and covariance functions characterize a GP entirely, the design of those two functions determines how well the GP learns the structure of data. However, GPs by using, for example, radial basis functions (RBFs) and Matérn class of covariance functions are stationary, and hence those conventional GPs have limitations on learning non-stationary structures in data. Heteroscedastic GPs (Le et al. 2005; Lazaro-Gredilla and Titsias 2011) are designed to tackle with the non-stationarity in measurement noise. To model the non-stationary of the unknown process, we

often need to manipulate the covariance function to give non-stationary GPs.

One approach to construct non-stationary GPs is to transform the domain/input space by compositions. For example, Wilson et al. (2016a, b); Al-Shedivat et al. (2017) transform the inputs by deterministic deep architectures and then feed to GPs, where the deep transformations are responsible for capturing the non-stationarity from data. The resulting GP posterior distribution is in closed form. Similarly, Calandra et al. (2016) transform the input space to manifold feature spaces. Damianou and Lawrence (2013) construct deep Gaussian processes (DGPs) by feeding the outputs of GPs to another layer of GPs as (transformed) inputs. However, the posterior inference requires complicated approximations and does not scale well with a large number of measurements (Salimbeni and Deisenroth 2017a).

Another commonly used non-stationary GP construction is to have input-dependent covariance function hyperparameters, so that the resulting covariance function is non-stationary (Sampson and Guttorp 1992; Higdon et al. 1999; Paciorek and Schervish 2004). For example, one can parametrize the lengthscale as a function of time. This method grants GPs the capability of changing behaviour depending on the input. However, one needs to be careful to ensure that the construction leads to valid (positive defi-

✉ Zheng Zhao
    zheng.zhao@aalto.fi

[1] Department of Electrical Engineering and Automation, Aalto University, Espoo, Finland

nite) covariance functions (Paciorek and Schervish 2004). It is also possible to put GP priors on the covariance parameters (Tolvanen et al. 2014; Lazaro-Gredilla and Titsias 2011; Heinonen et al. 2016; Roininen et al. 2019; Monterrubio-Gómez et al. 2020). For example, Salimbeni and Deisenroth (2017b) model the lengthscale as a GP by using the non-stationary covariance function of Paciorek and Schervish (2006), and they approximate the posterior density via the variational Bayes approach.

The idea of putting GP priors on the hyperpameters of a GP (Heinonen et al. 2016; Roininen et al. 2019; Salimbeni and Deisenroth 2017b) can be continued hierarchically, which leads to one type of deep Gaussian process (DGP) construction (Dunlop et al. 2018; Emzir et al. 2019, 2020). Namely, the GP is conditioned on another GP, which again depends on another GP, and so forth. It is worth emphasizing that this hyperparameter-based (or covariance-operator) construction of DGP is different from the compositional DGPs as introduced by Damianou and Lawrence (2013) and Duvenaud et al. (2014). In these composition-based DGPs, the output of each GP is fed as an input to another GP. Despite the differences, these two types of DGP constructions are similar in many aspects and are often analyzed under the same framework (Dunlop et al. 2018).

This paper focuses on hyperparameter-based (or covariance-operator) temporal DGPs. In particular, we introduce the state-space representations of DGPs by using non-linear stochastic differential equations (SDEs). The SDEs form a hierarchical non-linear system of conditionally linear SDEs which results from the property that a temporal GP can be constructed as a solution to a linear SDE (Hartikainen and Särkkä 2010; Särkkä et al. 2013; Särkkä and Solin 2019). More generally, it is related to the connection of Gaussian fields and stochastic partial differential equations (SPDEs, Lindgren et al. 2011). (D)GP regression then becomes equivalent to the smoothing problem on the corresponding continuous-discrete state-space model (Särkkä et al. 2013). Additionally, by using the SDE representations of DGPs we can avoid to explicitly choose/design the covariance function.

However, the posterior distribution of (state-space) DGPs does not admit a closed-form solution as a plain GP does. Hence we need to use approximations such as maximum a posteriori (MAP) estimates, Laplace approximations, Markov chain Monte Carlo (MCMC, Heinonen et al. 2016; Brooks et al. 2011; Luengo et al. 2020), or variational Bayes methods (Lazaro-Gredilla and Titsias 2011; Salimbeni and Deisenroth 2017a; Chang et al. 2020). However, those methods are often computationally heavy. The another benefit of using state-space DGPs is that we can use the Bayesian filtering and smoothing solvers which are particularly efficient for solving temporal regression/smoothing problems (Särkkä 2013).

In short, we formulate the (temporal) DGP regression as a state-estimation problem on a non-linear continuous-discrete state-space model. For this purpose, various well-established filters and smoothers are available, for example, the Gaussian (assumed density) filters and smoothers (Särkkä 2013; Särkkä and Sarmavuori 2013; Zhao et al. 2021). For temporal data, the computational complexity of using filtering and smoothing approaches is $\mathcal{O}(N)$, where N is the number of measurements.

The contributions of the paper are the follows. (1) We construct a general hyperparameter-based deep Gaussian process (DGP) model and formulate a batch MAP solution for it as a standard reference approach. (2) We convert the DGP into a state-space form consisting of a system of stochastic differential equations. (3) For the state-space DGP, we formulate the MAP and Bayesian filtering and smoothing solutions. The resulting computational complexity scales linearly with respect to the number of measurements. (4) We prove that for a class of DGP constructions and Gaussian approximations on the DGP posterior, certain nodes of the DGP (e.g., the magnitude σ of Matérn GP) will not be asymptotically updated from measurements. (5) We conduct experiments on synthetic data and also apply the methods to gravitational wave detection.

2 Deep Gaussian processes

2.1 Non-stationary Gaussian processes

We start by reviewing the classical Gaussian process (GP) regression problem (Rasmussen and Williams 2006). We consider the model

$$\begin{aligned} f(t) &\sim \mathcal{GP}(0, C(t, t')), \\ y_k &= f(t_k) + r_k, \end{aligned} \tag{1}$$

where $f : \mathbb{T} \to \mathbb{R}$ is a zero-mean GP on $\mathbb{T} = \{t \in \mathbb{R} : t \geq t_0\}$ with a covariance function C. The observation $y_k := y(t_k)$ of $f(t_k)$ is contaminated by a Gaussian noise $r_k \sim \mathcal{N}(0, R_k)$. We let $\mathbf{R} = \text{diag}(R_1, \ldots, R_N)$. Given a set of N measurements $\mathbf{y}_{1:N} = \{y_1, \ldots, y_N\}$, GP regression aims at obtaining the posterior distribution

$$p(f \mid \mathbf{y}_{1:N}),$$

which is again Gaussian with closed-form mean and covariance functions (Rasmussen and Williams 2006). In this model, the choice of covariance function C is crucial to the GP regression as it determines, for example, the smoothness and stationarity of the process. Typical choices, such as radial basis or Matérn covariance functions, give stationary GPs.

However, it is difficult for a stationary GPs to tackle with non-stationary data. The main problem arises from the covariance function (Rasmussen and Williams 2006) as the value of a stationary covariance function only depends on the difference of inputs. That is to say, the correlations of any pairs of two inputs are the same when the differences are the same. This feature is not beneficial for non-stationary signals, as the correlation might vary depending on the input.

A solution to this problem is using a non-stationary covariance function (Higdon et al. 1999; Paciorek and Schervish 2004, 2006; Lindgren et al. 2011). That grants GP with the capability of adaption by learning hyperparameter functions from data. However, one needs to carefully design the non-stationary covariance function such that it is positive definite. Recent studies by, for example, Heinonen et al. (2016); Roininen et al. (2019) and Monterrubio-Gómez et al. (2020), propose to put GP priors on the covariance function hyperparameters. In this article, we follow these approaches to construct hierarchy of GPs which becomes the construction of the deep GP model.

2.2 Deep Gaussian process construction

We define a deep Gaussian process (DGP) in a general perspective as follows. Suppose that the DGP has L layers, and each layer ($i = 1, \ldots, L$) is composed of L_i nodes. Each node of the DGP is conditionally a GP, denoted by $u^i_{j,k}$, where $k = 1, 2, \ldots, L_i$. We give three indices for the node. The indices i and k specify the layer and the position of the GP, respectively. As an example, $u^i_{j,k}$ is located in the i-th layer of the DGP and is the k-th node in the i-th layer. The index j is introduced to indicate the conditional connection to its unique child node on the previous layer. That is to say, $u^i_{j,k}$ is the child of nodes $u^{i+1}_{k,k'}$ for all suitable k'. The terminologies "child" and "parent" follow from the graphical model conventions (Bishop 2006). To keep the notation consistent, we also use $u^1_{1,1} := f$ for the top layer GP. The nodes $u^{L+1}_{j,k}$ outside of the DGP, we treat as degenerate random variables (i.e., constants or trainable hyperparameters). Remark that every node in the DGP is uniquely indexed by i and k, whereas j only serves the purpose of showing the dependency instead of indexing.

We call the vector process

$$U : \mathbb{T} \to \mathbb{R}^{\sum_{i=1}^{L} L_i},$$

the DGP, where each element of U corresponds (one to one and onto) to the element of the set of all nodes $\left\{ u^i_{j,k} : i = 1, \ldots, L, k = 1, 2, \ldots, L_i \right\}$. Similarly, each element of the vector $U^i : \mathbb{T} \to \mathbb{R}^{L_i}$ corresponds to the element of the set of all nodes from the i-th layer. We denote by
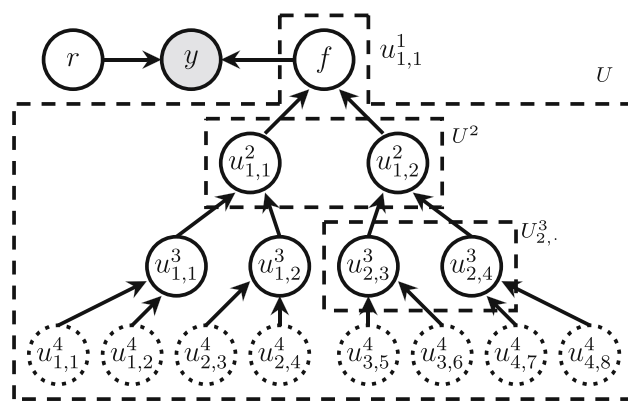

Fig. 1 Example of a 3-layer DGP regression model, where each (conditional) GP depends on two other GPs. Variable y is the measurement, and the nodes in U^4 are degenerate random variables

$$U^i_{k,\cdot} = \left\{ u^i_{k,k'} : \text{for all suitable } k' \right\}$$ the set of all parent nodes of $u^{i-1}_{j,k}$.

In this tree-like general construction of DGP U, there are $\sum_{i=1}^{L} L_i$ nodes in total. Every $u^i_{j,k}$ is independent of other nodes in the same i-th layer, and depends on the nodes $U^{i+1}_{j,\cdot}$ on the next ($i + 1$)-th layer. When there is only one layer, the DGP reduces to a conventional GP. Figure 1 illustrates the DGP construction.

The realization of the DGP depends on how each of the conditionally GP nodes is constructed. In the following sections, we discuss two realizations of this DGP, by either constructing the conditional GPs by specifying the mean and covariance functions, or by stochastic differential equations. These two constructions lead to DGP regression in batch and sequential forms, respectively.

3 A batch deep Gaussian process regression model

In this section, we present a batch DGP construction which uses the construction of non-stationary GPs presented in Paciorek and Schervish (2006) to form the DGP. To emphasize the difference to the SDE construction which is the main topic of this article, we call this the batch-DGP. Let us assume that every conditional GP in the DGP has zero mean and we observe the top GP f with additive Gaussian noise. We write down the DGP regression model as

$$
\begin{aligned}
f \mid U^2 &\sim \mathcal{GP}(0, C(t, t'; U^2)) \\
u^2_{1,1} \mid U^3_{1,\cdot} &\sim \mathcal{GP}(0, C^2_1(t, t'; U^3_{1,\cdot})) \\
u^2_{1,2} \mid U^3_{2,\cdot} &\sim \mathcal{GP}(0, C^2_2(t, t'; U^3_{2,\cdot})) \\
&\vdots \\
u^i_{\cdot,k} \mid U^{i+1}_{k,\cdot} &\sim \mathcal{GP}(0, C^i_k(t, t'; U^{i+1}_{k,\cdot}))
\end{aligned}
$$

$$\vdots$$
$$y_k = f(t_k) + r_k, \tag{2}$$

where each covariance function $C_k^i \colon \mathbb{T} \times \mathbb{T} \to \mathbb{R}$ is parameterized by next layer's (conditional) GPs. That is to say, the covariance function C_k^i takes the nodes in $U_{k,\cdot}^{i+1}$ as parameters.

This DGP construction requires positive covariance function at each node. One option is the non-stationary exponential covariance function which has the form (cf. Paciorek and Schervish [2006])

$$
\begin{aligned}
C_{NS}(t, t'; \ell, \sigma) &= \frac{\sigma(t)\sigma(t')}{\Gamma(\nu)2^{\nu-1}} \ell^{\frac{1}{4}}(t)\ell^{\frac{1}{4}}(t') \\
&\quad \times \sqrt{2} \exp\left(\frac{-\sqrt{2}\,|t-t'|}{\sqrt{\ell(t)+\ell(t')}}\right) (\ell(t)+\ell(t'))^{-\frac{1}{2}}.
\end{aligned}
$$

In the above covariance function C_{NS}, the length scale $\ell(t)$ and magnitude $\sigma(t)$ are functions of input t. Paciorek and Schervish ([2006]) also generalize C_{NS} to the Matérn class.

For the DGP construction in (2) we need to ensure the positivity of the hyperparameter functions. For that purpose we introduce a wrapping function $g \colon \mathbb{R} \to (0, \infty)$ which is positive and smooth, and we put $\ell(t) = g(u_\ell(t))$ and $\sigma(t) = g(u_\sigma(t))$ where u_ℓ and u_σ are the conditionally Gaussian processes from the next layer. The exponential or squaring functions are typical options for g. In Example 1, we show a two-layer DGP by using the covariance function C_{NS}.

Example 1 Consider a two layer exponential (Matérn) DGP

$$
\begin{aligned}
f \mid u_{1,1}^2, u_{1,2}^2 &\sim \mathcal{GP}(0, C_{NS}(t, t'; g(u_{1,1}^2), g(u_{1,2}^2))), \\
u_{1,1}^2 &\sim \mathcal{GP}(0, C_{NS}(t, t'; g(u_{1,1}^3), g(u_{1,2}^3))), \\
u_{1,2}^2 &\sim \mathcal{GP}(0, C_{NS}(t, t'; g(u_{2,3}^3), g(u_{2,4}^3))).
\end{aligned}
$$

In this case, we have the so-called length scale $\ell_{1,1}^2 = g(u_{1,1}^2)$ and magnitude $\sigma_{1,2}^2 = g(u_{1,2}^2)$. Also, $U = \begin{bmatrix} f\ u_{1,1}^2\ u_{1,2}^2 \end{bmatrix}^{\mathsf{T}}$ and $U^2 = U_{1,\cdot}^2 = \begin{bmatrix} u_{1,1}^2\ u_{1,2}^2 \end{bmatrix}^{\mathsf{T}}$.

Given a set of measurements $\mathbf{y}_{1:N} = \{y_1, y_2, \ldots, y_N\}$, the aim of DGP regression is to obtain the posterior density

$$
p(U \mid \mathbf{y}_{1:N}) = \frac{p(\mathbf{y}_{1:N} \mid U)\,p(U)}{p(\mathbf{y}_{1:N})}, \tag{3}
$$

for any input $t \in \mathbb{T}$. Moreover, by the construction of DGP (conditional independence) we have

$$
p(U) = p(f \mid U^2) \prod_{i=2}^{L} \prod_{k=1}^{L_i} p\left(u_{j,k}^i \mid U_{k,\cdot}^{i+1}\right), \tag{4}
$$

where each $u_{j,k}^i \mid U_{k,\cdot}^{i+1}$ is a GP as defined in (2). We isolate $f \mid U^2$ out of the above factorization because we are particularly interested in the observed f. It is important to remark that the distribution of U is (usually) not Gaussian because of the non-Gaussianility induced by the conditional hierarchy of Gaussian processes which depend on each other non-linearly.

3.1 Batch MAP solution

The maximum a posteriori (MAP) estimate gives a point estimate of U as the maximum of the posterior distribution (3). Let us denote $f_{1:N} = \begin{bmatrix} f(t_1)\ f(t_2)\ \cdots\ f(t_N) \end{bmatrix}^{\mathsf{T}} \in \mathbb{R}^N$, $U_{1:N} = \left\{ u_{j,k|1:N}^i \colon \text{for all } i, k \right\}$, where $u_{j,k|1:N}^i = \begin{bmatrix} u_{j,k}^i(t_1)\ \cdots\ u_{j,k}^i(t_N) \end{bmatrix}^{\mathsf{T}} \in \mathbb{R}^N$. We are targeting at the posterior density $p(U_{1:N} \mid \mathbf{y}_{1:N})$ evaluated at $t_1, \ldots, t_N$. The MAP estimate is then obtained by

$$
U_{1:N}^{\mathrm{BMAP}} = \underset{U_{1:N}}{\arg\min}\, \mathcal{L}^{\mathrm{BMAP}}(U_{1:N}; \mathbf{y}_{1:N}), \tag{5}
$$

where $\mathcal{L}^{\mathrm{BMAP}}$ is the negative logarithm of the unnormalized posterior distribution given by

$$
\begin{aligned}
&\mathcal{L}^{\mathrm{BMAP}}(U_{1:N}; \mathbf{y}_{1:N}) \\
&= -\log\left[p(\mathbf{y}_{1:N} \mid U_{1:N})\,p(U_{1:N})\right] \\
&= \frac{1}{2}\left[(\mathbf{y}_{1:N} - f_{1:N})^{\mathsf{T}}\,\mathbf{R}^{-1}\,(\mathbf{y}_{1:N} - f_{1:N}) + \log|2\pi\,\mathbf{R}|\right] \\
&\quad + \frac{1}{2}\left[f_{1:N}^{\mathsf{T}}\,\mathbf{C}^{-1}\,f_{1:N} + \log|2\pi\,\mathbf{C}|\right] \\
&\quad + \frac{1}{2}\sum_{i=2}^{L}\sum_{k=1}^{L_i}\left[(u_{j,k|1:N}^i)^{\mathsf{T}}(\mathbf{C}_k^i)^{-1}u_{j,k|1:N}^i + \log\left|2\pi\,\mathbf{C}_k^i\right|\right].
\end{aligned}
\tag{6}
$$

In the above Eq. (6), $\mathbf{C}$ and $\mathbf{C}_k^i$ are the covariance matrices formed by evaluating the corresponding GP covariance functions at $(t_1, \ldots, t_N) \times (t_1, \ldots, t_N)$. The computational complexity for computing (6) is $\mathcal{O}(N^3 \sum_{i=1}^{L} L_i)$, which scales cubically with the number of measurements.

It is important to recall from (2) that the covariance matrices also depend on the other GP nodes (i.e., $f_{1:N}$ and $U_{1:N}$ are in $\mathbf{C}_k^i$). Therefore the objective function $\mathcal{L}^{\mathrm{BMAP}}$ is non-quadratic. Additional non-linear terms are also introduced by the determinants of the covariance matrices. However, quasi-Newton methods (Nocedal and Wright [2006]) can be used to solve the optimization problem. The required gradients of (6) are provided in Appendix 1.

There are two major challenges in this MAP solution. Firstly, the optimization of (5) is not computationally cheap. It requires to evaluate and store $\sum_{i=1}^{L} L_i$ inversions of N-dimensional matrices for every optimization iteration. This

prevents the use of the DGP on large-scale datasets and large models. Moreover, Paciorek and Schervish (2006) state that the optimization of (5) is difficult and prone to overfitting, which we also confirm in the experiment section. Another problem is the uncertainty quantification and prediction (interpolation) with the MAP estimate which is degenerate.

4 Deep Gaussian processes in state-space

Stochastic differential equations (SDEs) are common models to construct stochastic processes (Friedman 1975; Rogers and Williams 2000a, b; Särkkä and Solin 2019). Instead of constructing the process by specifying, for example, the mean and covariance functions, an SDEs characterizes a process by describing the dynamics with respect to a Wiener process. In this section, we show how a DGP as defined in Sect. 2.2 can be realized using a hierarchy of SDEs. To highlight the difference to the previous batch-DGP realization, we call this the SS-DGP. The regression problem on this class of DGPs can be seen as a state estimation problem.

4.1 Gaussian processes as solutions of linear SDEs

Consider a linear time invariant (LTI) SDE

$$
\begin{aligned}
d\mathbf{f}(t) &= \mathbf{A}\,\mathbf{f}(t)\,dt + \mathbf{L}\,d\mathbf{W}_f(t), \\
\mathbf{f}(t_0) &\sim \mathcal{N}(\mathbf{0}, \mathbf{P}_\infty),
\end{aligned} \tag{7}
$$

where coefficients $\mathbf{A} \in \mathbb{R}^{d \times d}$ and $\mathbf{L}^{d \times S}$ are constant matrices, $\mathbf{W}_f(t) \in \mathbb{R}^S$ is a Wiener process with unit spectral density, and $\mathbf{f}(t_0)$ is a Gaussian initial condition with zero mean and covariance $\mathbf{P}_\infty$, which is obtained as the solution to

$$
\mathbf{A}\,\mathbf{P}_\infty + \mathbf{P}_\infty\,\mathbf{A}^\mathsf{T} + \mathbf{L}\,\mathbf{L}^\mathsf{T} = \mathbf{0}. \tag{8}
$$

When the stationary covariance $\mathbf{P}_\infty$ exists, the vector process $\mathbf{f}$ is a stationary Gaussian process with the (cross-)covariance function

$$
\mathrm{Cov}\left[\mathbf{f}(t), \mathbf{f}(t')\right] = \begin{cases} \mathbf{P}_\infty \exp\left[(t' - t)\,\mathbf{A}\right]^\mathsf{T}, & t < t', \\ \exp\left[-(t' - t)\,\mathbf{A}\right]\mathbf{P}_\infty, & t \geq t'. \end{cases} \tag{9}
$$

It now turns out that we can construct matrices $\mathbf{A}$, $\mathbf{L}$, and $\mathbf{H}$ such that $f = \mathbf{H}\mathbf{f}$ is a Gaussian process with a given covariance function (Hartikainen and Särkkä 2010; Särkkä et al. 2013; Särkkä and Solin 2019). The marginal covariance of f can be extracted by $\mathrm{Cov}[f(t), f(t')] = \mathbf{H}\,\mathrm{Cov}\left[\mathbf{f}(t), \mathbf{f}(t')\right]\mathbf{H}^\mathsf{T}$. In order to construct non-stationary GPs, we can let the SDE coefficients (i.e., $\mathbf{A}$ and $\mathbf{L}$) be functions of time.

In particular, if f is a Matérn GP, then we can select the state

$$
\mathbf{f}(t) = \begin{bmatrix} f(t)\; Df(t)\; \cdots\; D^\alpha f(t) \end{bmatrix}^\mathsf{T} \in \mathbb{R}^d, \tag{10}
$$

and the corresponding $\mathbf{H} = \begin{bmatrix} 1\; 0\; 0\; \cdots \end{bmatrix}$, where D is the time derivative, α is the smoothness factor, and dimension $d = \alpha + 1$. We can also generalize the results to spatial-temporal Gaussian processes, and hence the corresponding SDEs will become stochastic partial differential equations (SPDEs, Särkkä and Hartikainen 2012; Särkkä et al. 2013).

When constructing a GP using SDEs, we sometimes need to select the SDE coefficients suitably so that the resulting covariance function (9) admits a desired form (e.g., Matérn). One way to proceed is to find the spectral density function of the GP covariance function (via Wiener–Khinchin theorem) and translate the resulting transfer function into the state-space form (Hartikainen and Särkkä 2010). The results are known for many classes of GPs, for example, the Mátern and periodic GPs (Särkkä and Solin 2019).

As an alternative to the batch-GP construction in Sect. 3, the SDE approach offers more freedom to certain extent because the corresponding covariance functions are positive definite and non-stationary by construction. It is also computationally beneficial in regression, as we can leverage the Markov properties of the SDEs in the computations.

4.2 Deep Gaussian processes as hierarchy of SDEs

So far, we have only considered the SDE construction of a single stationary/non-stationary GP. To realize a DGP as defined in Sect. 2.2, we need to formulate a hierarchical system composed of linear SDEs. Namely, we parametrize the SDE coefficients as functions of other GPs in a hierarchical structure. Followed from the SDE expression of GP f in Eq. (7), let us similarly define the state

$$
\mathbf{u}^i_{j,k}: \mathbb{T} \to \mathbb{R}^d,
$$

for any GP $u^i_{j,k}$ in the DGP U. We then construct the DGP by finding the SDE representation for each $u^i_{j,k}$ to yield

$$
\begin{aligned}
d\mathbf{f} &= \mathbf{A}(\mathbf{U}^2_{1,.})\,\mathbf{f}\,dt + \mathbf{L}(\mathbf{U}^2_{1,.})\,d\mathbf{W}_f, \\
d\mathbf{u}^2_{1,1} &= \mathbf{A}^2_1(\mathbf{U}^3_{1,.})\,\mathbf{u}^2_{1,1}\,dt + \mathbf{L}^2_1(\mathbf{U}^3_{1,.})\,d\mathbf{W}^2_1, \\
d\mathbf{u}^2_{1,2} &= \mathbf{A}^2_2(\mathbf{U}^3_{2,.})\,\mathbf{u}^2_{1,2}\,dt + \mathbf{L}^2_2(\mathbf{U}^3_{2,.})\,d\mathbf{W}^2_2, \\
&\;\;\vdots \\
d\mathbf{u}^i_{j,k} &= \mathbf{A}^i_k(\mathbf{U}^{i+1}_{k,.})\,\mathbf{u}^i_{j,k}\,dt + \mathbf{L}^i_k(\mathbf{U}^{i+1}_{k,.})\,d\mathbf{W}^i_k, \\
&\;\;\vdots
\end{aligned} \tag{11}
$$

where $\mathbf{W}_f \in \mathbb{R}^S$ and $\mathbf{W}_k^i \in \mathbb{R}^S$ for all i and k are mutually independent standard Wiener processes. Note that the above SDE system (11) is *non-linear*, and the coefficients are state-dependent. We denote by $\mathbf{U}_{k,\cdot}^{i+1}$ the collection states for all parent states of $\mathbf{u}_{j,k}^i$. For example, if $\mathbf{u}_{1,1}^2$ is conditioned on $\mathbf{u}_{1,1}^3$ and $\mathbf{u}_{1,2}^3$, then $\mathbf{U}_{1,\cdot}^3 = \left[\left(\mathbf{u}_{1,1}^3\right)^{\mathsf{T}} \left(\mathbf{u}_{1,2}^3\right)^{\mathsf{T}} \right]^{\mathsf{T}}$. To further condense the notation, we rearrange the above SDEs (11) into

$$
\begin{aligned}
&\mathrm{d}\mathbf{U}(t) = \boldsymbol{\Lambda}(\mathbf{U}(t))\,\mathrm{d}t + \boldsymbol{\beta}(\mathbf{U}(t))\,\mathrm{d}\mathbf{W}(t), \\
&\mathbf{U}(t_0) \sim \mathcal{N}(\mathbf{0}, \mathbf{P}_0),
\end{aligned}
\tag{12}
$$

where

$$
\mathbf{U}(t) = \left[\mathbf{f}^{\mathsf{T}} \left(\mathbf{u}_{1,1}^2\right)^{\mathsf{T}} \left(\mathbf{u}_{1,2}^2\right)^{\mathsf{T}} \cdots \left(\mathbf{u}_{j,k}^i\right)^{\mathsf{T}} \cdots \right]^{\mathsf{T}} \in \mathbb{R}^\varrho,
$$

is the SDE state of the entire DGP, $\mathbf{U}(t_0)$ is the Gaussian initial condition, $\varrho = d \sum_{i=1}^L L_i$ is the total dimension of the state, and

$$
\mathbf{W}(t) = \left[\mathbf{W}_f^{\mathsf{T}} \, (\mathbf{W}_1^2)^{\mathsf{T}} \cdots (\mathbf{W}_k^i)^{\mathsf{T}} \cdots \right]^{\mathsf{T}} \in \mathbb{R}^\varrho.
$$

The drift $\boldsymbol{\Lambda} \circ \mathbf{U} : \mathbb{T} \to \mathbb{R}^\varrho$ and dispersion $\boldsymbol{\beta} \circ \mathbf{U} : \mathbb{T} \to \mathbb{R}^\varrho$ functions can be written as

$$
\boldsymbol{\Lambda}(\mathbf{U}(t)) = \begin{bmatrix} \mathbf{A}(\mathbf{U}_{1,\cdot}^2) & & & & \\ & \mathbf{A}_1^2(\mathbf{U}_{1,\cdot}^3) & & & \\ & & \ddots & & \\ & & & \mathbf{A}_k^i(\mathbf{U}_{k,\cdot}^{i+1}) & \\ & & & & \ddots \end{bmatrix} \mathbf{U}(t),
$$

and

$$
\boldsymbol{\beta}(\mathbf{U}(t)) = \mathrm{diag}\left(\mathbf{L}(\mathbf{U}_{1,\cdot}^2), \mathbf{L}_1^2(\mathbf{U}_{1,\cdot}^3), \ldots, \mathbf{L}_k^i(\mathbf{U}_{k,\cdot}^{i+1}), \ldots \right),
$$

respectively.

The above SDE representation of DGP is general in the sense that the SDE coefficients of each GP and the number of layers are free. However, they cannot be completely arbitrary as we at least need to require that the SDE has a weakly unique solution. A classical sufficient condition is to have the coefficients globally Lipschitz continuous and have at most linear growth (Friedman 1975; Xu et al. 2008; Mao 2008; Øksendal 2003). These restrictive conditions can be further weakened, for example, to locally Lipschitz (Friedman 1975, Ch. 5) and weaker growth condition (Shen et al. 2006, Theorem 2.2). Alternatively, requiring the coefficients to be Borel measurable and locally bounded is enough for a unique solution (Rogers and Williams 2000b, Theorem 21.1 and Equation 21.9).

It is also worth remarking that the SDE system (12) and hence the DGP is a well-defined Itô diffusion, provided that the coefficients are regular enough (Definition 7.1.1, Øksendal 2003). This feature is valuable, as being an Itô diffusion offers many fruitful properties that we can use in practice, for example, continuity, Markov property, and the existence of infinitesimal generator (Øksendal 2003). The Markov property is needed to ensure the existence of transition density and also enables the use of Bayesian filtering and smoothing for regression. The infinitesimal generator can be used to discretize the SDEs as we do in Sect. 5.

It is also possible to extend the SDE representations of temporal DGPs to stochastic partial differential equation (SPDE) representations of spatio-temporal DGPs. Särkkä et al. (2013) given the following result. Suppose $v : \mathbb{X} \times \mathbb{T} \to \mathbb{R}$ is a spatio-temporal stationary GP on a suitable domain, such that $v(\mathbf{x}, t) \sim \mathcal{GP}(0, C(\mathbf{x}, \mathbf{x}', t, t'))$. Then $\mathbf{v}(\mathbf{x}, t)$ can be constructed as a solution to an evolution type of SPDE

$$
\frac{\partial \mathbf{v}(\mathbf{x}, t)}{\partial t} = \mathcal{A}\mathbf{v}(\mathbf{x}, t) + \mathcal{B}w(\mathbf{x}, t),
$$

where $\mathbf{v}(\mathbf{x}, t)$ is the state of v, $\mathcal{A}$ and $\mathcal{B}$ are spatial operators, and $w(\mathbf{x}, t)$ is the spatio-temporal white noise. Emzir et al. (2020) build a deep Gaussian field based on the Matérn SPDE by Lindgren et al. (2011), which provides another path to the spatio-temporal case.

4.3 Deep Matérn process

In this section, we present a Matérn construction of SS-DGP (12). The coefficients are chosen such that each SDE corresponds to a conditional GP with the Matérn covariance function. The idea is to find an equivalent SDE representation for each Matérn GP node, and then parametrize the covariance parameters (i.e., length-scale ℓ and magnitude σ) with another layer of Matérn GPs. We are interested in a GP

$$
u_{j,k}^i \mid \ell, \sigma \sim \mathcal{GP}(0, C(t, t'; \ell, \sigma)),
\tag{13}
$$

with the Matérn covariance function

$$
C(t, t') = \frac{\sigma^2 \, 2^{1-\nu}}{\Gamma(\nu)} \left(\kappa \, |t - t'| \right)^\nu K_\nu \left(\kappa \, |t - t'| \right),
\tag{14}
$$

where K_ν is the modified Bessel function of the second kind and Γ is the Gamma function. We denote $\kappa = \sqrt{2\nu}/\ell$ and $\nu = \alpha + 1/2$.

As shown by Hartikainen and Särkkä (2010) and Särkkä et al. (2013), one possible SDE representation of Matérn GP $u_{j,k}^i$ in Eq. (13) is

$$
\mathrm{d}\mathbf{u}_{j,k}^i = \mathbf{A}_k^i \, \mathbf{u}_{j,k}^i \, \mathrm{d}t + \mathbf{L}_k^i \, \mathrm{d}W_k^i,
\tag{15}
$$

where the state

$$\mathbf{u}_{j,k}^i(t) = \left[u_{j,k}^i(t) \; Du_{j,k}^i(t) \; \cdots \; D^\alpha u_{j,k}^i(t) \right]^\mathsf{T} \in \mathbb{R}^d,$$

and the SDE coefficients $\mathbf{A}_k^i$ and $\mathbf{L}_k^i$ admit the form

$$\mathbf{A}_k^i = \begin{bmatrix} 0 & 1 & & \\ & 0 & 1 & \\ \vdots & & & \ddots \\ -\binom{\alpha}{0}\kappa^\alpha & -\binom{\alpha}{1}\kappa^{\alpha-1} & \cdots & -\binom{\alpha}{\alpha-1}\kappa \end{bmatrix},$$

and

$$\mathbf{L}_k^i = \left[0 \; 0 \; \cdots \; \frac{\sigma\,\Gamma(\alpha+1)}{\sqrt{\Gamma(2\alpha+1)}}\,(2\kappa)^{(\alpha+\frac{1}{2})} \right]^\mathsf{T}, \quad (16)$$

respectively. Above, we denote by $\binom{\alpha}{1}$ a binomial coefficient and $W_k^i \in \mathbb{R}$ is a standard Wiener process. Next, to contruct the deep Matérn process, we need to parametrize the length scale ℓ and magnitude σ by the states of parent GPs and build the system as in Eq. (11). For example, if we want to use $u_{1,1}^3$ and $u_{1,2}^3$ to model the length scale and magnitude of $u_{1,1}^2$, then $\ell_{1,1}^2 = g(u_{1,1}^3)$ and $\sigma_{1,1}^2 = g(u_{1,2}^3)$. The wrapping function $g: \mathbb{R} \to (0, \infty)$ is mandatory to ensure the positivity of Matérn parameters. The minimal requirement for function g is to be positive and Borel measurable. For instance, we can let $g(u) = \exp(u) + c$ or $g(u) = u^2 + c$ for some $c > 0$. Another choice is to let $g(u) = 1/(u^2+c)$ that is bounded and Lipschitz on $\mathbb{R}$, which makes the deep Matérn process an Itô diffusion and we have the SS-DGP well-defined (Øksendal 2003).

Note that the resulting state-space model composed of (15) has a canonical form from control theory (Glad and Ljung 2000), and the dimensionality is determined by the smoothness parameter α. Moreover, the coefficient $\mathbf{A}_k^i$ is Hurwitz, because all the eigenvalues have strictly negative real part. The stability of such system is studied, for example, in Khasminskii (2012).

Example 2 Corresponding to Example 1, the SDE construction of the two layer (exponential) Matérn process is formulated as follows:

$$df = -\frac{1}{g(u_{1,1}^2)}\,f\,dt + \frac{\sqrt{2}\,g(u_{1,2}^2)}{\sqrt{g(u_{1,1}^2)}}\,dW_f,$$

$$du_{1,1}^2 = -\frac{1}{g(u_{1,1}^3)}\,u_{1,1}^2\,dt + \frac{\sqrt{2}\,g(u_{1,2}^3)}{\sqrt{g(u_{1,1}^3)}}\,dW_{1,1}^2,$$

$$du_{1,2}^2 = -\frac{1}{g(u_{2,3}^3)}\,u_{1,2}^2\,dt + \frac{\sqrt{2}\,g(u_{2,4}^3)}{\sqrt{g(u_{2,3}^3)}}\,dW_{1,2}^2,$$

where we have states $\mathbf{U} = \left[f \; u_{1,1}^2 \; u_{1,2}^2 \right]^\mathsf{T}$ and the SDE coefficient functions

$$\boldsymbol{\Lambda}(\mathbf{U}) = \begin{bmatrix} -\frac{1}{g(u_{1,1}^2)} & & \\ & -\frac{1}{g(u_{1,1}^3)} & \\ & & -\frac{1}{g(u_{2,3}^3)} \end{bmatrix} \mathbf{U},$$

and $\boldsymbol{\beta}(\mathbf{U}) = \mathrm{diag}\left(\frac{\sqrt{2}\,g(u_{1,2}^2)}{\sqrt{g(u_{1,1}^2)}}, \frac{\sqrt{2}\,g(u_{1,2}^3)}{\sqrt{g(u_{1,1}^3)}}, \frac{\sqrt{2}\,g(u_{2,4}^3)}{\sqrt{g(u_{2,3}^3)}} \right)$. The length scale $\ell_{1,1}^2$ and magnitude $\sigma_{1,2}^2$ of f are given by $\ell_{1,1}^2 = g(u_{1,1}^2)$ and $\sigma_{1,2}^2 = g(u_{1,2}^2)$, respectively.

5 State-space deep Gaussian process regression

In this section, we formulate sequential state-space regression by DGPs. By using the result in Eq. (12), the state-space regression model is

$$\begin{aligned} d\mathbf{U}(t) &= \boldsymbol{\Lambda}(\mathbf{U}(t))\,dt + \boldsymbol{\beta}(\mathbf{U}(t))\,d\mathbf{W}(t), \\ y_k &= \mathbf{H}\,\mathbf{U}(t_k) + r_k, \end{aligned} \quad (17)$$

where the initial condition $\mathbf{U}(t_0) \sim \mathcal{N}(\mathbf{0}, \mathbf{P}_0)$ is independent of $\mathbf{W}(t)$ for $t \geq 0$, and $\mathbf{H}\,\mathbf{U}(t_k) = f(t_k)$ extracts the top GP f from the state. We also assume that the functions $\boldsymbol{\Lambda}$ and $\boldsymbol{\beta}$ are selected suitably such that the SDE (17) has a weakly unique solution and imply Markov property (Friedman 1975). The deep Matérn process and Example 2 satisfy the required two conditions, provided that function g is chosen properly.

Suppose we have a set of observations $\mathbf{y}_{1:N} = \{y_1, y_2, \ldots, y_N\}$, then the posterior density of interests is

$$p(\mathbf{U}(t) \mid \mathbf{y}_{1:N}), \quad (18)$$

for any $t_1 \leq t \leq t_N$. Since we have discrete-time measurements, let us denote by

$$\mathbf{U}_k := \mathbf{U}(t_k),$$

for $k = 1, 2, \ldots, N$ and use $\mathbf{U}_{1:N} = \{\mathbf{U}_1, \ldots, \mathbf{U}_N\}$. Also, it would be is possible to extend the regression to classification by using a categorical measurement model (Rasmussen and Williams 2006; Garcia-Fernández et al. 2019).

5.1 SDE discretization

To obtain the posterior density with discrete-time observations, we need the transition density of the SDE, such that $\mathbf{U}_{k+1} \sim p(\mathbf{U}_{k+1} \mid \mathbf{U}_k)$. It is known that the transition density is the solution to the Fokker–Planck–Kolmogorov (FPK)

partial differential equation (PDE, Särkkä and Solin 2019). However, solving a PDE is not computationally cheap, and does not scale well for large-dimensional state. It is often more convenient to discretize the SDEs and approximate the continuous-discrete state-space model (17) with a discretized version

$$
\begin{aligned}
\mathbf{U}_{k+1} &= \mathbf{a}(\mathbf{U}_k) + \mathbf{q}(\mathbf{U}_k), \\
y_k &= \mathbf{H}\,\mathbf{U}_k + r_k,
\end{aligned}
\tag{19}
$$

where $\mathbf{a}\colon \mathbb{R}^\varrho \to \mathbb{R}^\varrho$ is a function of state, and $\mathbf{q}\colon \mathbb{R}^\varrho \to \mathbb{R}^\varrho$ is a zero-mean random variable depending on the state. One of the most commonly used methods to derive functions $\mathbf{a}$ and $\mathbf{q}$ is the Euler–Maruyama scheme (Kloeden and Platen 1992). Unfortunately, the Euler–Maruyama is not applicable for many DGP models, as the covariance $\mathbf{q}$ would be singular. As an example, a smooth Matérn ($\alpha \geq 1$) GP's SDE representation gives a singular $\boldsymbol{\beta}(\mathbf{U}_k)\,\boldsymbol{\beta}^\mathsf{T}(\mathbf{U}_k)$ (see Eq. (16)), thus the transition density $p(\mathbf{U}_{k+1} \mid \mathbf{U}_k)$ is ill-defined.

The Taylor moment expansion (TME) is one way to proceed instead of Euler–Maruyama (Zhao et al. 2021; Kessler 1997; Florens-Zmirou 1989). This method requires that the SDE coefficients $\boldsymbol{\Lambda}$ and $\boldsymbol{\beta}$ are differentiable and there exists an infinitesimal generator for the SDE (Zhao et al. 2021). The deep Matérn process satisfies these conditions provided that the wrapping function g is chosen suitably.

We remark that at this point that we have formed an approximation to the SS-DGP in order to use its Markov property. This is different from the batch-DGP model where we do not utilize the Markov property for regression. In summary, we approximate the transition density

$$
\begin{aligned}
p(\mathbf{U}_{k+1} \mid \mathbf{U}_k) &\approx \mathcal{N}(\mathbf{U}_{k+1} \mid \mathbf{a}(\mathbf{U}_k), \mathbf{Q}(\mathbf{U}_k)), \\
\mathbf{Q}(\mathbf{U}_k) &= \mathrm{Cov}\left[\mathbf{q}(\mathbf{U}_k) \mid \mathbf{U}_k\right],
\end{aligned}
$$

as a non-linear Gaussian, where a discretization such as Euler–Maruyama or TME is used. With the transition density formulated, we can now approximate the posterior density (18) of SS-DGP using sequential methods in state-space.

5.2 State-space MAP solution

The MAP solution to the SS-DGP model is fairly similar to the batch-DGP model, except that we factorize the posterior density with the Markov property. Suppose that we are interested in the posterior density $p(\mathbf{U}_{0:N} \mid \mathbf{y}_{1:N})$ at N discrete observation points, then we factorize the posterior density by

$$
\begin{aligned}
&p(\mathbf{U}_{0:N} \mid \mathbf{y}_{1:N}) \\
&\propto p(\mathbf{y}_{1:N} \mid \mathbf{U}_{0:N})\, p(\mathbf{U}_{0:N}) \\
&= \prod_{k=1}^{N} \mathcal{N}(y_k \mid \mathbf{H}\,\mathbf{U}_k, R_k)\, p(\mathbf{U}_0) \prod_{k=1}^{N} p(\mathbf{U}_k \mid \mathbf{U}_{k-1}).
\end{aligned}
\tag{20}
$$

By taking the negative logarithm on both sides of Eq. (20), the MAP estimate of SS-DGP is given by

$$
\mathbf{U}_{0:N}^{\mathrm{SMAP}} = \underset{\mathbf{U}_{0:N}}{\arg\min}\; \mathcal{L}^{\mathrm{SMAP}}(\mathbf{U}_{0:N}; \mathbf{y}_{1:N}),
\tag{21}
$$

where

$$
\begin{aligned}
&\mathcal{L}^{\mathrm{SMAP}}(\mathbf{U}_{0:N}; \mathbf{y}_{1:N}) \\
&= -\log\left[p(\mathbf{y}_{1:N} \mid \mathbf{U}_{1:N})\, p(\mathbf{U}_0) \prod_{k=1}^{N} p(\mathbf{U}_k \mid \mathbf{U}_{k-1}) \right] \\
&= \frac{1}{2} \sum_{k=1}^{N} \left[\frac{1}{R_k}(y_k - \mathbf{H}\,\mathbf{U}_k)^2 + \log\det(2\,\pi\,R_k) \right] \\
&\quad + \sum_{k=1}^{N} \Big[(\mathbf{U}_k - \mathbf{a}(\mathbf{U}_{k-1}))^\mathsf{T}\,\mathbf{Q}^{-1}(\mathbf{U}_{k-1})\,(\mathbf{U}_k - \mathbf{a}(\mathbf{U}_{k-1})) \\
&\quad + \log\det(2\,\pi\,\mathbf{Q}(\mathbf{U}_{k-1})) \Big] \times \frac{1}{2} \\
&\quad + \frac{1}{2}\left[\mathbf{U}_0^\mathsf{T}\,\mathbf{P}_0^{-1}\,\mathbf{U}_0 + \log\det(2\,\pi\,\mathbf{P}_0) \right].
\end{aligned}
\tag{22}
$$

The corresponding gradient of (22) is given in Appendix 2. The computational complexity of this SS-DGP MAP estimation is $\mathcal{O}(N\,(d\sum_{i=1}^{L} L_i)^3)$ which is in contrast with the complexity $\mathcal{O}(N^3 \sum_{i=1}^{L} L_i)$ of the batch-DGP. We see that the state-space MAP solution has an advantage with large dataset, as the computational complexity is linear with respect to the number of data points N.

The state-space MAP method also has the problem that it is inherently a point estimate. One way to proceed is to use a Bayesian filter and smoother instead of the MAP estimates (Särkkä 2013).

5.3 Bayesian filtering and smoothing solution

Recall the original SS-DGP model (17). The estimation of the state from an observed process is equivalent to computing the posterior distribution (18) which in turn is equivalent to a continuous-discrete filtering and smoothing problem (Jazwinski 1970; Särkkä and Solin 2019). Compared to the MAP solution, the Bayesian smoothing approaches offer the full posterior distribution instead of a point estimate.

The core idea of Bayesian smoothing is to utilize the Markov property of the process and approximate the posterior density recursively at each time step. In particular, we are interested in the filtering posterior

$$
p(\mathbf{U}_k \mid \mathbf{y}_{1:k}),
\tag{23}
$$

and the smoothing posterior

$$p(\mathbf{U}_k \mid \mathbf{y}_{1:N}), \tag{24}$$

for any $k = 1, 2, \ldots, N$. There are many well-known methods to obtain the above posterior densities, such as the Kalman filter and Rauch–Tung–Striebel smoother for linear Gaussian state-space models. Typical methods for non-linear SS-DGP models are the Gaussian filters and smoothers (Särkkä and Sarmavuori 2013; Kushner 1967; Itô and Xiong 2000). Some popular examples are the extended Kalman filter and smoother (EKF/EKS), and the unscented or cubature Kalman filter and smoothers (UKF/UKS/CKF/CKS). The significant benefit of Gaussian filters and smoothers is the computational efficiency, as they scale linearly with the number of measurements.

Remark 1 Although the Gaussian filters and smoothers are beneficial choices in terms of computation, there are certain limitations when applying them to DGP regression. We elucidate this peculiar characteristic in Sect. 6.

Instead of Gaussian filters and smoothers, we can use a particle filter and smoother on a more general setting of DGPs (Godsill et al. 2004; Andrieu et al. 2010). Typical choices are the bootstrap particle filter (PF, Gordon et al. 1993) with resampling procedures (Kitagawa 1996) and the backward-simulation particle smoother (PF-BS, Godsill et al. 2004). However, particle filters and smoothers do not usually scale well with the dimension of state-space, as we need more particles to represent the probability densities in higher dimensions. Other non-Gaussian assumed density filters and smoothers might also apply, for example, the projection filter and smoother (Brigo et al. 1998; Koyama 2018).

6 Analysis on Gaussian approximated DGP posterior

Gaussian filters are particularly efficient methods, which approximate the DGP posterior (23) and the predictive density $p(\mathbf{U}_k \mid \mathbf{y}_{1:k-1})$ as Gaussian (Itô and Xiong 2000). Under linear additive Gaussian measurement models, the posterior density is approximated analytically by applying Gaussian identities. However, we are going to show that this type of Gaussian approximation is not useful for all constructions of DGPs. In particular, we show that the estimated posterior covariance of the observed GP $f(t)$ and an inner GP $\sigma(t)$ approaches to zero as $t \to \infty$. As a consequence, the Gaussian filtering update for $\sigma(t)$ will not use information from measurements as $t \to \infty$.

Hereafter, we restrict our analysis to a certain construction of DGPs and a class of Gaussian approximations (filters) for which we can prove the covariance vanishing property.

Therefore, in Sect. 6.1 we define a construction of DGPs, and in Algorithm 1 we formulate a type of Gaussian filters. The main result is revealed in Theorem 1.

We organize the proofs as follows. First we show that at every time step the predictions from DGPs give vanishing prior covariance (in Lemma 1). Then we show that the Gaussian filter update step also shrinks the covariance (in Theorem 1). Finally we prove the vanishing posterior covariance by mathematical induction over all time steps as in Theorem 1.

6.1 Preliminaries and assumptions

Let $f: \mathbb{T} \to \mathbb{R}$ and $u_\sigma: \mathbb{T} \to \mathbb{R}$ be the solution to the pair of SDEs

$$
\begin{aligned}
\mathrm{d}f(t) &= \mu(u_\ell(t))\, f(t)\, \mathrm{d}t + \theta(u_\ell(t), u_\sigma(t))\, \mathrm{d}W_f(t), \\
\mathrm{d}u_\sigma(t) &= a(u_v(t))\, u_\sigma(t)\, \mathrm{d}t + b(u_v(t))\, \mathrm{d}W_\sigma(t),
\end{aligned}
\tag{25}
$$

for $t \geq t_0$ starting from random initial conditions $f(t_0), u_\sigma(t_0)$ which are independent of the Wiener processes $W_f(t) \in \mathbb{R}$ and $W_\sigma(t) \in \mathbb{R}$. In addition, $u_\ell: \mathbb{T} \to \mathbb{R}$ and $u_v: \mathbb{T} \to \mathbb{R}$ are two independent processes driving the SDEs (25), which are also independent of $W_f(t) \in \mathbb{R}$ and $W_\sigma(t) \in \mathbb{R}$ for $t \geq t_0$.

Let $y(t_k) = f(t_k) + r(t_k)$ be the noisy observation of $f(t)$ at time t_k, where $r(t_k) \sim \mathcal{N}(0, R_k)$ and $k = 1, 2, \ldots$. Also let $\mathbf{y}_{1:k} = \{y_1, \ldots, y_k\}$ and $\Delta t = t_k - t_{k-1} > 0$ for all k. We make the following assumptions.

Assumption 1 The functions $\mu: \mathbb{R} \to (-\infty, 0)$, $\theta: \mathbb{R} \times \mathbb{R} \to \mathbb{R}$, $a: \mathbb{R} \to (-\infty, 0)$, and $b: \mathbb{R} \times \mathbb{R} \to \mathbb{R}$ and the initial conditions $f(t_0), u_\sigma(t_0), u_\ell(t_0)$, and $u_v(t_0)$ are chosen regular enough so that the solution to SDEs (25) exists.

Assumption 2 $\mathbb{E}\left[f^2(t_0)\right] < \infty$, $\mathbb{E}\left[u_\sigma^2(t_0)\right] < \infty$, and $\mathbb{E}\left[(f(t_0)\, u_\sigma(t_0))^2\right] < \infty$.

Assumption 3 There exists constants $C_\mu < 0$ and $C_a < 0$ such that $(\mu \circ u_\ell)(t) \leq C_\mu$ and $(a \circ u_v)(t) \leq C_a$ almost surely.

Assumption 4 $\mathbb{E}\left[(\mu(u_\ell(t))f(t))^2\right] \leq C < \infty$ almost everywhere and $\mathbb{E}\left[\theta^2(u_\ell(t), u_\sigma(t))\right] < \infty$. Also $\mathbb{E}\left[\theta^2(u_\ell(t), u_\sigma(t))\right] \geq C_\theta > 0$ almost everywhere.

Assumption 5 There exists a constant $C_R > 0$ such that $R_k \geq C_R$ for all $k = 1, 2, \ldots$, or there exists a k such that $R_k = 0$.

The solution existence in Assumption 1 is the prerequisite for the analysis of SDEs (25) (Kuo 2006; Øksendal 2003). Assumption 2 ensures that the SDEs start from a reasonable condition which is used in Lemma 1. Assumption 3 postulates negativity on functions μ and a. It implies that the sub-processes f and u_σ stay near zero. Also, the

negativity guarantees the positivity of lengthscale (e.g., the lengthscale of $f(t)$ is $-\mu(u_\ell(t))$). Assumption 4 yields a lower bound on the variance of f as stated in Corollary 1. Finally, Assumption 5 means that the measurement noise admits a lower bound uniformly which is used in Theorem 1. This assumption also allows for perfect measurements (i.e., no measurement noises).

The above SDEs (25) and Assumptions 1-5 correspond to a type of DGP constructions. In particular, f is a conditional GP given u_σ and u_ℓ. Also, u_σ is another conditional GP given u_v. The processes u_ℓ and u_v are two independent processes that drive f and u_σ. The Matérn DGP in Example 2 satisfies the above assumptions, if we choose Gaussian initial conditions and a regular wrapping function by, for example, $g(u) = u^2 + c$ and $c > 0$.

6.2 Theoretical results

The following Lemma 1 shows that the covariance of $f(t)$ and $u_\sigma(t)$ approaches to zero as $t \to \infty$.

Lemma 1 *Under Assumptions 1 to 3,*

$$\lim_{t \to \infty} \mathrm{Cov}[f(t), u_\sigma(t)] = 0. \tag{26}$$

Proof Let $m_f(t) := \mathbb{E}[f(t)]$, $m_\sigma(t) := \mathbb{E}[u_\sigma(t)]$. By Itô's lemma (see, e.g., Theorem 4.2 of Särkkä and Solin 2019),

$$
\begin{aligned}
&\mathrm{d}\left(f(t)\, u_\sigma(t)\right) \\
&= [u_\sigma(t)\, \mu(u_\ell(t))\, f(t) + f(t)\, a(u_v(t))\, u_\sigma(t)]\, \mathrm{d}t \\
&\quad + \frac{1}{2}[u_\sigma(t)\, \theta(u_\ell(t), u_\sigma(t))\, \mathrm{d}W_f(t) \\
&\quad + f(t)\, b(u_v(t))\, \mathrm{d}W_\sigma(t)].
\end{aligned}
\tag{27}
$$

To analyze the relation between f and u_σ, we need to fix the information from u_v and u_ℓ. Hence, let $\mathcal{F}_t^v$ and $\mathcal{F}_t^\ell$ be the generated filtrations of $u_v(t)$ and $u_\ell(t)$, respectively. Taking conditional expectations on the above Eq. (27) gives

$$
\begin{aligned}
&\mathrm{d}\mathbb{E}\left[f(t)\, u_\sigma(t) \mid \mathcal{F}_t^v, \mathcal{F}_t^\ell\right] \\
&= \mathbb{E}\left[(\mu(u_\ell(t)) + a(u_v(t)))\, f(t)\, u_\sigma(t) \mid \mathcal{F}_t^v, \mathcal{F}_t^\ell\right] \mathrm{d}t \\
&= (\mu(u_\ell(t)) + a(u_v(t)))\, \mathbb{E}\left[f(t)\, u_\sigma(t) \mid \mathcal{F}_t^v, \mathcal{F}_t^\ell\right] \mathrm{d}t.
\end{aligned}
$$

Thus

$$
\begin{aligned}
&\mathbb{E}\left[f(t)\, u_\sigma(t) \mid \mathcal{F}_t^v, \mathcal{F}_t^\ell\right] \\
&= \mathbb{E}[f(t_0)\, u_\sigma(t_0) \mid u_v(t_0), u_\ell(t_0)]\, e^{\int_{t_0}^t \mu(u_\ell(s)) + a(u_v(s))\, \mathrm{d}s}.
\end{aligned}
$$

Using the same approach, we derive

$$
\begin{aligned}
\mathbb{E}\left[f(t) \mid \mathcal{F}_t^\ell\right] &= \mathbb{E}[f(t_0) \mid u_\ell(t_0)]\, e^{\int_{t_0}^t \mu(u_\ell(s))\, \mathrm{d}s}, \\
\mathbb{E}\left[u_\sigma(t) \mid \mathcal{F}_t^v\right] &= \mathbb{E}[u_\sigma(t_0) \mid u_v(t_0)]\, e^{\int_{t_0}^t a(u_v(s))\, \mathrm{d}s}.
\end{aligned}
\tag{28}
$$

Then by law of total expectation, we recover

$$
\begin{aligned}
&\mathrm{Cov}[f(t), u_\sigma(t)] \\
&= \mathbb{E}\Big[\mathbb{E}[f(t_0)\, u_\sigma(t_0) \mid u_\ell(t_0), u_v(t_0)] \\
&\qquad \times e^{\int_{t_0}^t \mu(u_\ell(s)) + a(u_v(s))\, \mathrm{d}s}\Big] \\
&\quad - \mathbb{E}\left[\mathbb{E}[f(t_0) \mid u_\ell(t_0)]\, e^{\int_{t_0}^t \mu(u_\ell(s))\, \mathrm{d}s}\right] \\
&\qquad \times \mathbb{E}\left[\mathbb{E}[u_\sigma(t_0) \mid u_v(t_0)]\, e^{\int_{t_0}^t a(u_v(s))\, \mathrm{d}s}\right].
\end{aligned}
\tag{29}
$$

Taking the limit of Eq. (29) gives

$$
\begin{aligned}
&\lim_{t \to \infty} \mathrm{Cov}[f(t), u_\sigma(t)] \\
&= \lim_{t \to \infty} \mathbb{E}\Big[\mathbb{E}[f(t_0)\, u_\sigma(t_0) \mid u_\ell(t_0), u_v(t_0)] \\
&\qquad \times e^{\int_{t_0}^t \mu(u_\ell(s)) + a(u_v(s))\, \mathrm{d}s}\Big] \\
&\quad - \lim_{t \to \infty} \mathbb{E}\left[\mathbb{E}[f(t_0) \mid u_\ell(t_0)]\, e^{\int_{t_0}^t \mu(u_\ell(s))\, \mathrm{d}s}\right] \\
&\qquad \times \lim_{t \to \infty} \mathbb{E}\left[\mathbb{E}[u_\sigma(t_0) \mid u_v(t_0)]\, e^{\int_{t_0}^t a(u_v(s))\, \mathrm{d}s}\right],
\end{aligned}
\tag{30}
$$

where all the three limits on the right side turn out to be zero. Let us first focus on $\mathbb{E}\left[\mathbb{E}[f(t_0) \mid u_\ell(t_0)]\, e^{\int_{t_0}^t \mu(u_\ell(s))\, \mathrm{d}s}\right]$. By Jensen's inequality ([see, e.g., Theorem 7.9 of Klenke 2014)

$$
\begin{aligned}
&\left|\mathbb{E}\left[\mathbb{E}[f(t_0) \mid u_\ell(t_0)]\, e^{\int_{t_0}^t \mu(u_\ell(s))\, \mathrm{d}s}\right]\right| \\
&\leq \mathbb{E}\left[\left|\mathbb{E}[f(t_0) \mid u_\ell(t_0)]\, e^{\int_{t_0}^t \mu(u_\ell(s))\, \mathrm{d}s}\right|\right],
\end{aligned}
$$

for $t \in \mathbb{T}$. Then by Hölder's inequality (see, e.g., Theorem 7.16 of Klenke 2014), the above inequality continues as

$$
\begin{aligned}
&\mathbb{E}\left[\left|\mathbb{E}[f(t_0) \mid u_\ell(t_0)]\, e^{\int_{t_0}^t \mu(u_\ell(s))\, \mathrm{d}s}\right|\right] \\
&\leq \sqrt{\mathbb{E}\left[\mathbb{E}^2[f(t_0) \mid u_\ell(t_0)]\right]}\, \sqrt{\mathbb{E}\left[e^{2\int_{t_0}^t \mu(u_\ell(s))\, \mathrm{d}s}\right]}.
\end{aligned}
$$

Now by using Assumption 3, we know that there exists a constant $C_\mu < 0$ such that $(\mu \circ u_\ell)(t) \le C_\mu$ almost surely. Hence

$$\mathbb{E}\left[e^{\int_{t_0}^t \mu(u_\ell(s))\,\mathrm{d}s}\right] \le \mathbb{E}\left[e^{C_\mu\,(t-t_0)}\right] = e^{C_\mu\,(t-t_0)},$$

for all $t > t_0$. Therefore

$$\lim_{t\to\infty} \mathbb{E}\left[\mathbb{E}[f(t_0) \mid u_\ell(t_0)]\, e^{\int_{t_0}^t \mu(u_\ell(s))\,\mathrm{d}s}\right]$$
$$= \sqrt{\mathbb{E}\left[\mathbb{E}^2[f(t_0) \mid u_\ell(t_0)]\right]}\, \lim_{t\to\infty} e^{2\,C_\mu\,(t-t_0)} = 0.$$

Assumption 2 ensures that $\mathbb{E}\left[\mathbb{E}^2[f(t_0) \mid u_\ell(t_0)]\right]$ is finite. Similarly, we obtain the zero limits for the rest of the terms in Eq. (30). Thus limit (26) holds. □

The almost sure negativity (i.e., Assumption 3) on functions $\mu(\cdot)$ and $a(\cdot)$ is the key condition we need to have for the covariance to vanish to zero in infinite time. These conditions are often true in an SDE representation of a DGP because $\mu(\cdot)$ and $a(\cdot)$ ensure the positivity of lengthscales.

Before analyzing the posterior covariance, we need to construct a positive lower bound on the variance of $f(t)$, which is given in Lemma 2 and Corollary 1.

Lemma 2 *Under Assumption 1, for any $\epsilon > 0$, there is $\zeta > 0$ such that*

$$\mathrm{Var}[f(t)] \ge \frac{1}{z(t)} \int_{t_0}^t z(s) \left(\mathbb{E}\left[\theta^2(u_\ell(s), u_\sigma(s))\right]\right.$$
$$\left. - 2\epsilon \sqrt{\mathbb{E}\left[(\mu(u_\ell(s))\, f(s))^2\right]}\right) \mathrm{d}s,$$
(31)

where $z(t) = \exp \int_{t_0}^t 2\zeta \sqrt{\mathbb{E}[(\mu(u_\ell(s))\, f(s))^2]}\,\mathrm{d}s$.

Proof Denote by $P(t) := \mathrm{Var}[f(t)] = \mathbb{E}[(f(t) - \mathbb{E}[f(t)])^2]$. By applying Itô's lemma on $(f(t) - \mathbb{E}[f(t)])^2$ and taking expectation, we obtain

$$P(t) = P(t_0) + 2\int_{t_0}^t \mathbb{E}[\mu(u_\ell(s))\, f(s)(f(s) - \mathbb{E}[f(s)])]\,\mathrm{d}s$$
$$+ \int_{t_0}^t \mathbb{E}\left[\theta^2(u_\ell(s), u_\sigma(s))\right]\,\mathrm{d}s,$$
(32)

where the initial $P(t_0) > 0$. By Jensen's and Hölder's inequalities (Klenke 2014),

$$|\mathbb{E}[\mu(u_\ell(t))\, f(t)(f(t) - \mathbb{E}[f(t)])]|$$
$$\le \sqrt{\mathbb{E}[(\mu(u_\ell(t))\, f(t))^2]}\sqrt{P(t)}.$$

We now form a linear bound on $\sqrt{P(t)}$ such that for any $\epsilon > 0$, there is $\zeta > 0$ such that $\sqrt{P(t)} \le \epsilon + \zeta P(t)$. Next, to prove the bound (31), we use the differential form of (32) and get

$$\frac{\mathrm{d}P(t)}{\mathrm{d}t}$$
$$\ge -2\sqrt{\mathbb{E}[(\mu(u_\ell(t))\, f(t))^2]}\sqrt{P(t)} + \mathbb{E}[\theta^2(u_\ell(t), u_\sigma(t))]$$
$$\ge -2\zeta\sqrt{\mathbb{E}[(\mu(u_\ell(t))\, f(t))^2]}\, P(t)$$
$$+ \left(\mathbb{E}\left[\theta^2(u_\ell(t), u_\sigma(t))\right] - 2\epsilon\sqrt{\mathbb{E}[(\mu(u_\ell(t))\, f(t))^2]}\right),$$

Now, we introduce $z(t) = \exp\int_{t_0}^t 2\zeta\sqrt{\mathbb{E}[(\mu(u_\ell(s))\, f(s))^2]}\,\mathrm{d}s$, and then by integrating factor method on $\frac{\mathrm{d}}{\mathrm{d}t}(z(t)\, P(t))$, we recover the bound (31). □

Corollary 1 *Under Assumptions 1 and 4, there exists $\epsilon > 0$ and $C_F(\Delta t) > 0$ such that*

$$\mathrm{Var}[f(t)] \ge C_F(\Delta t).$$
(33)

Proof From Lemma 2, we know that for any $\epsilon > 0$, there is $\zeta > 0$ such that Eq. (31) holds. By Assumption 4, we have $1 \le z(t) \le \exp(2\zeta\,\Delta t\,\sqrt{C})$. Also, we have $\mathbb{E}\left[\theta^2(u_\ell(t), u_\sigma(t))\right] - 2\epsilon\sqrt{\mathbb{E}\left[(\mu(u_\ell(t))\, f(t))^2\right]} \ge C_\theta - 2\epsilon\sqrt{C}$ almost everywhere. Thus let us choose any small enough $\epsilon < \frac{C_\theta}{2\sqrt{C}}$ so that $C_\theta - 2\epsilon\sqrt{C} > 0$. Now let $C_F = \frac{(C_\theta - 2\epsilon\sqrt{C})\Delta t}{\exp(2\zeta\,\Delta t\,\sqrt{C})}$ hence Eq. (33) holds. Note that the inequality (33) only depends on Δt and some fixed parameters of the SDEs. □

The following Algorithm 1 formulates a partial procedure for estimating the posterior density using a Gaussian approximation. In particular, Algorithm 1 gives an approximation

$$P_k^{f,\sigma} \approx \mathrm{Cov}[f(t_k), u_\sigma(t_k) \mid \mathbf{y}_{1:k}],$$

to the posterior covariance for $k = 1, 2, \ldots$. In order to do so, we need to make predictions through SDEs (25) based on different starting conditions at each time step. Hence let us introduce two notations as following. We denote by

$$\mathrm{Cov}[f(t), u_\sigma(t)](c_0),$$

and

$$\mathrm{Var}[f(t)](s_0),$$

the functions of t in Eqs. (29) and (32) starting from initial values $c_0 \in \mathbb{R}$ and $s_0 \in (0, +\infty)$ at t_0, respectively.

Algorithm 1 *(Gaussian posterior approximation for $P_k^{f,\sigma}$)
Let us approximate the posterior densities $p(f(t_k), u_\sigma(t_k) \mid \mathbf{y}_{1:k})$ by Gaussian densities for $k = 1, 2, \ldots$. Suppose that the initial condition is known and particularly $P_0^{f,\sigma} := \mathrm{Cov}[f(t_0), u_\sigma(t_0)]$ and $P_0^{f,f} := \mathrm{Var}[f(t_0)]$. Then starting from $k = 1$ we calculate*

$$\bar{P}_k^{f,\sigma} = \mathrm{Cov}[f(t_k), u_\sigma(t_k)](P_{k-1}^{f,\sigma}), \tag{34}$$

and

$$\bar{P}_k^{f,f} = \mathrm{Var}[f(t_k)](P_{k-1}^{f,f}), \tag{35}$$

through the SDEs (25) and update

$$P_k^{f,\sigma} = \bar{P}_k^{f,\sigma} - \frac{\bar{P}_k^{f,f} \, \bar{P}_k^{f,\sigma}}{\bar{P}_k^{f,f} + R_k}, \tag{36}$$

for $k = 1, 2, \ldots$.

Remark 2 The above Algorithm 1 is a abstraction of continuous-discrete Gaussian filters (Itô and Xiong 2000; Särkkä and Solin 2019), except that the predictions through SDEs (25) are done exactly in Eqs. (34) and (35). The derivation of Eq. (36) is shown in Appendix 3. Note that in practice the predictions might also involve various types of Gaussian approximations and even numerical integrations (e.g., sigma-point methods).

Theorem 1 *Suppose that Assumptions 1 to 5 hold. Further assume that $|\mathrm{Cov}[f(t), u_\sigma(t)](c_0)| \leq |c_0|$ for all $c_0 \in \mathbb{R}$, then Algorithm 1 gives*

$$\lim_{k \to \infty} P_k^{f,\sigma} = 0. \tag{37}$$

Proof We are going to use induction to prove that the claim

$$\left| P_k^{f,\sigma} \right| \leq \left| P_0^{f,\sigma} \right| \prod_{i=1}^{k} M_i \tag{38}$$

holds for all $k = 1, 2, \ldots$, where $M_i = \frac{R_i}{\bar{P}_i^{f,f} + R_i}$. To do so, we expand $\left| P_k^{f,\sigma} \right|$ by

$$\left| P_k^{f,\sigma} \right| = \left| \bar{P}_k^{f,\sigma} - \frac{\bar{P}_k^{f,f} \, \bar{P}_k^{f,\sigma}}{\bar{P}_k^{f,f} + R_k} \right|$$
$$= \left| \frac{\bar{P}_k^{f,\sigma} R_k}{\bar{P}_k^{f,f} + R_k} \right| = M_k \left| \bar{P}_k^{f,\sigma} \right| \leq M_k \left| P_{k-1}^{f,\sigma} \right|. \tag{39}$$

Now we can verify that $\left| P_1^{f,\sigma} \right| \leq \left| P_0^{f,\sigma} \right| M_1$ when $k = 1$, which satisfies the induction claim (38). Suppose that

Eq. (38) holds for a given $k > 1$, then we can calculate Eq. (39) at $k + 1$ giving

$$\left| P_{k+1}^{f,\sigma} \right| = M_{k+1} \left| \bar{P}_{k+1}^{f,\sigma} \right| \leq M_{k+1} \left| P_k^{f,\sigma} \right|$$
$$\leq M_{k+1} \left| P_0^{f,\sigma} \right| \prod_{i=1}^{k} M_i = \left| P_0^{f,\sigma} \right| \prod_{i=1}^{k+1} M_i, \tag{40}$$

which satisfies the induction claim (38). Thus Eq. (38) holds. Above, we used the assumption $|\mathrm{Cov}[f(t), u_\sigma(t)](c_0)| \leq |c_0|$ for all $c_0 \in \mathbb{R}$ to get $\left| \bar{P}_{k+1}^{f,\sigma} \right| \leq \left| P_k^{f,\sigma} \right|$ for any k.

By Corollary 1, Assumption 5, and a fixed non-zero Δt, we know that $\bar{P}_k^{f,f}$ are lower bounded uniformly over all k, thus $\lim_{k \to \infty} \prod_{i=1}^{k} M_i = 0$. Hence, by taking the limit on Eq. (38), the Eq. (37) holds. Also, this theorem trivially holds if $R_k = 0$ for some k or $P_0^{f,\sigma} = 0$ because $M_k = 0$ for all $k = 1, 2, \ldots$. □

Remark 3 Note that in Theorem 1, the initial bounding assumption $|\mathrm{Cov}[f(t), u_\sigma(t)](c_0)| \leq |c_0|$ for all $c_0 \in \mathbb{R}$ is needed because it is not always followed from Lemma 1. On the other hand, for any choice of $c_0 \in \mathbb{R}$, there always exists a threshold $\eta > 0$ such that for all $t > \eta$ we have $|\mathrm{Cov}[f(t), u_\sigma(t)](c_0)| \leq |c_0|$ because of Lemma 1.

Under the result of bounded $\mathrm{Var}[f(t)]$ in Corollary 1, the consequence of the vanishing posterior covariance in Theorem 1 is that the so-called Kalman gain for $u_\sigma(t)$ approaches zero asymptotically. It entails that the Kalman update for $u_\sigma(t)$ will use no information from measurements when $t \to \infty$. In the later experiment as shown in Fig. 8 we see that the corresponding estimated $u_\sigma(t)$ and covariance rapidly stabilizes to zero.

The previous Theorem 1 is formulated in a general sense which applies to DGP methods that use Algorithm 1 and satisfy Assumptions 1 to 5. A concrete example is shown in the following Example 3.
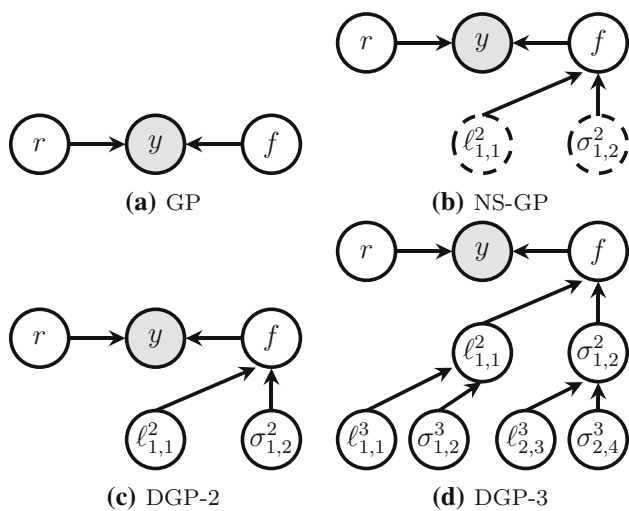
Example 3 Consider a system of SDEs,

$$\mathrm{d}f(t) = \mu \, f(t) \, \mathrm{d}t + u_\sigma(t) \, \mathrm{d}W_f(t),$$
$$\mathrm{d}u_\sigma(t) = a \, u_\sigma(t) \, \mathrm{d}t + b \, \mathrm{d}W_u(t), \tag{41}$$

starting from a Gaussian initial condition $f(t_0), u_\sigma(t_0)$, where constants $\mu < 0$, $a < 0$, and $b > 0$. The conditions of Theorem 1 are now satisfied, and thus $\lim_{k \to \infty} P_k^{f,\sigma} = 0$.

7 Experiments

In this section we numerically evaluate the proposed methods. The specific objectives of the experiments are as follows.
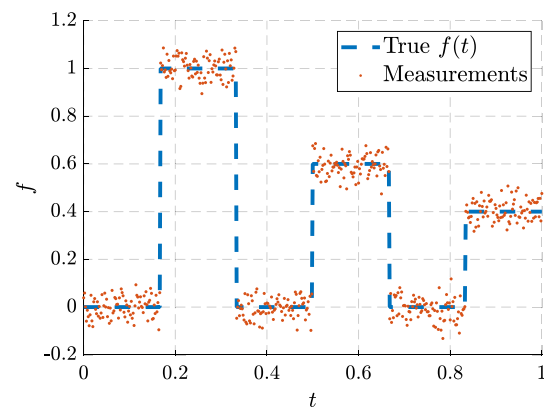
Fig. 2 Graphs of four regression models. We denote by y as the measurement of function f contaminated by noise r. In (**b**), the processes $\ell^2_{1,1}$ and $\sigma^2_{1,2}$ in dashed circles are degenerate (learnable hyperparameters)

Fig. 3 Demonstration of the magnitude-varying rectangle signal in Eq. (42) with 500 samples

First, we show the advantages of using DGPs over conventional GPs or non-stationary GPs (one-layer DGPs) in non-stationary regression. Then, we compare the batch and state-space constructions of DGPs. Finally, we examine the efficiencies of different DGP regression methods.

We prepare four regression models as shown in Fig. 2. These models are the conventional GP (Rasmussen and Williams 2006), non-stationary GP (NS-GP, Paciorek and Schervish 2006), two-layer DGP (DGP-2), and three-layer DGP (DGP-3). The DGP-2 and DGP-3 are constructed using both the batch and state-space approaches as formulated in Sects. 3 and 4 , respectively. In particular, we consider a Matérn type of GP construction, which only has two hyperparameters (i.e., the length scale ℓ and magnitude σ). That is to say, we use the non-stationary Matérn covariance function (Paciorek and Schervish 2006) for the NS-GP and batch-DGP models, and the deep Matérn process for SS-DGP model. For the wrapping function g, we choose $g(u) = \exp(u)$. For the discretization of SS-DGP, we use the 3rd-order TME method (Zhao et al. 2021). We control the smoothness of f and hyperparameter processes by using $\alpha = 1$ and 0, respectively (see Eq. (14)). In addition, we draw samples from the DGP priors in Appendix 4.

There are unknown model hyperparameters. We use the maximum likelihood estimation (MLE) routine to optimize the hyperparameters for the GP and NS-GP models which have closed-form likelihood functions and gradients. For the DGP models, we find them by grid searches because the gradients are non-trivial to derive. We detail the found hyperparameters in Appendix 5.

As for the batch-DGP models, we use the proposed batch maximum a posteriori (B-MAP) method in Sect. 3.1.

Similarly for the SS-DGP, we apply the state-space MAP (SS-MAP), Gaussian filters and smoothers (Särkkä 2013), and a bootstrap particle filter (PF, Andrieu et al. 2010; Doucet et al. 2000) and a backward-simulation particle smoother (PF-BS, Godsill et al. 2004).

We use the limited-memory Broyden–Fletcher–Goldfarb–Shanno (l-BFGS, Nocedal and Wright 2006) optimizer for MLE and MAP optimizations. For the Gaussian filters and smoothers, we exploit the commonly used linearization (EKFS) and spherical cubature method (CKFS) (Särkkä 2013). As for the PF and PF-BS, we use 200,000 particles and 1600 backward simulations.

The following experiments except the real application are computed with the Triton computing cluster at Aalto University, Finland.[1] We uniformly allocate 4 CPU cores and 4 gigabyte of memory for each of the individual experiment. In addition, the PF-BS method is implemented with CPU-based parallelization. All programs are implemented in MATLAB® 2019b.

7.1 Regression on rectangle signal

In this section, we conduct regression on a magnitude-varying rectangle wave, as shown in Fig. 3. The regression model is formulated by
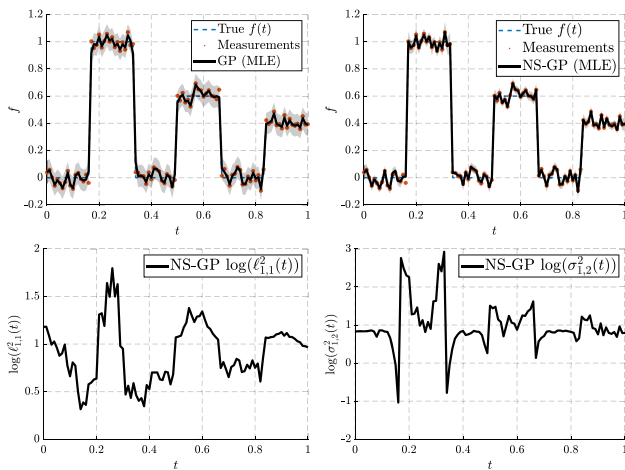
$$
f(t) = \begin{cases}
0, & t \in [0, \frac{1}{6}) \cup [\frac{2}{6}, \frac{3}{6}) \cup [\frac{4}{6}, \frac{5}{6}), \\
1, & t \in [\frac{1}{6}, \frac{2}{6}), \\
0.6, & t \in [\frac{3}{6}, \frac{4}{6}), \\
0.4, & t \in [\frac{5}{6}, 1],
\end{cases} \tag{42}
$$
$$
y(t) = f(t) + r(t),
$$

where f is the true function, y is the measurement, and $r(t) \sim \mathcal{N}(0, 0.002)$. We evenly generate samples $y(t_1), \dots, y(t_T)$,

---

[1] The code is available at https://github.com/zgbkdlm/ssdgp.

Fig. 4 GP and NS-GP regressions on model (42). The shaded area stands for 95% confidence interval

where $T = 100$. The challenge of this type of signal is that the rectangle wave is continuous and flat almost everywhere while it is only right-continuous at a finite number of isolated points. Moreover, the jumps have different heights.

We formulate the commonly used root mean square error (RMSE)

$$\left(\frac{1}{T} \sum_{k=1}^{T} \left(f(t_k) - \tilde{f}(t_k) \right)^2 \right)^{1/2}, \tag{43}$$

as well as the negative log predictive density (NLPD)

$$- \log \int p(\mathbf{y}^\star \mid \mathbf{f}^\star) \, \tilde{p}(\mathbf{f}^\star \mid \mathbf{y}_{1:T}) \, d\mathbf{f}^\star, \tag{44}$$

to numerically demonstrate the methods' effectiveness, where $\tilde{f}$ is the regression estimate, $\mathbf{y}^\star$ are the test data, and $\tilde{p}(\mathbf{f}^\star \mid \mathbf{y}_{1:T})$ is the estimated posterior density. Note that the NLPD metric is not applied to the MAP results. We run 100 independent Monte Carlo trials to average the RMSE and NLPD as well as the computational time. For visualization, we uniformly choose the results under the same random seed.

Figure 4 shows the results of GP and NS-GP regression. Both of GP and NS-GP experience overfitting problem on this rectangle signal, while the estimated posterior variance of NS-GP is significantly smaller than that of GP. The outcome of GP is expected, as the covariance function is stationary. Because there are no constraints (e.g., being time-continuous) on the parameters of NS-GP, the learnt $\ell_{1,1}^2$ and $\sigma_{1,2}^2$ overfit to the likelihood function individually at each time instant (cf. Paciorek and Schervish 2006). From Table 1 we can see that the RMSE and NLPD of GP and NS-GP are very close.

The results of B-MAP on batch-DGPs are shown in Fig. 5. We can see a slight improvement in overfitting compared to GP and NS-GP. However, the learnt function $f(t)$ of B-MAP is not smooth enough and is jittering. For B-MAP on DGP-2, the estimated $\ell_{1,1}^2$ and $\sigma_{1,2}^2$ change abruptly on the jump points, and do not stay at flat levels, especially $\ell_{1,1}^2$. On the contrary, the estimated $\ell_{1,1}^3$ and $\sigma_{1,2}^3$ on the last layer of DGP-3 stay mostly flat while changing sharply on the jump points. From Fig. 5 and the RMSEs of Table 1 we can see that the results of B-MAP on DGP-2 and DGP-3 are almost identical with subtle differences.

Compared to the batch-DGP, the SS-DGP method gives a better fit to the true function. This result is demonstrated in Fig. 7, where SS-MAP is used. There is no noticeable overfitting problem in the SS-MAP estimates. The learnt function
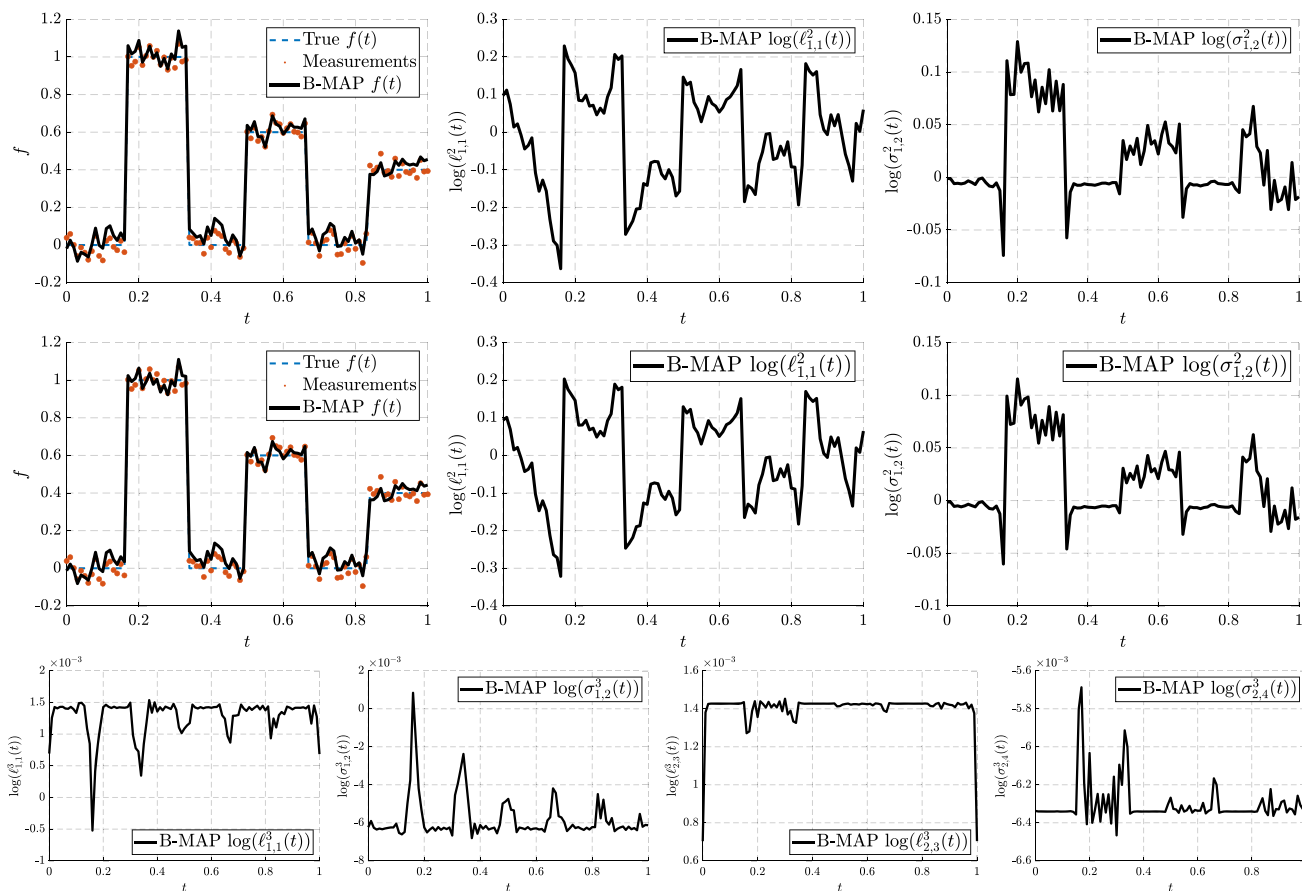
Table 1 Averaged RMSEs ($\times 10^{-2}$), NLPD, and computational time (in seconds) over different regression models and methods

Methods	RMSE (0^{-2})	NLPD	Time (s)
GP (MLE)	4.36 ± 0.3	-136.6 ± 8	2.0 ± 0.5
NS-GP (MLE)	4.28 ± 0.3	-135.9 ± 12	3.3 ± 0.2
B-MAP (DGP-2)	3.89 ± 0.3	N/A	454.9 ± 67
B-MAP (DGP-3)	3.80 ± 0.3	N/A	897.7 ± 37
SS-MAP (DGP-2)	2.04 ± 0.4	N/A	205.4 ± 26
SS-MAP (DGP-3)	$\mathbf{1.69} \pm 0.3$	N/A	479.5 ± 70
CKFS (DGP-2)	4.50 ± 0.3	-136.2 ± 7	0.2 ± 0.03
CKFS (DGP-3)	N/A	N/A	N/A
EKFS (DGP-2)	5.32 ± 0.2	-135.4 ± 8	$\mathbf{0.1} \pm 0.02$
EKFS (DGP-3)	7.77 ± 0.1	-119.6 ± 9	0.2 ± 0.01
PF (DGP-2)	4.25 ± 2.3	-135.5 ± 17	929.6 ± 200
PF (DGP-3)	3.73 ± 0.9	-145.1 ± 12	1460.8 ± 170
PF-BS (DGP-2)	4.08 ± 2.7	-140.6 ± 25	4-7 hrs
PF-BS (DGP-3)	3.35 ± 0.9	$\mathbf{-149.2} \pm 11$	17-20 hrs

Bold represent the best values in each of their columns

Fig. 5 B-MAP regression results on model (42) using DGP-2 (first row) and DGP-3 (second and third rows)
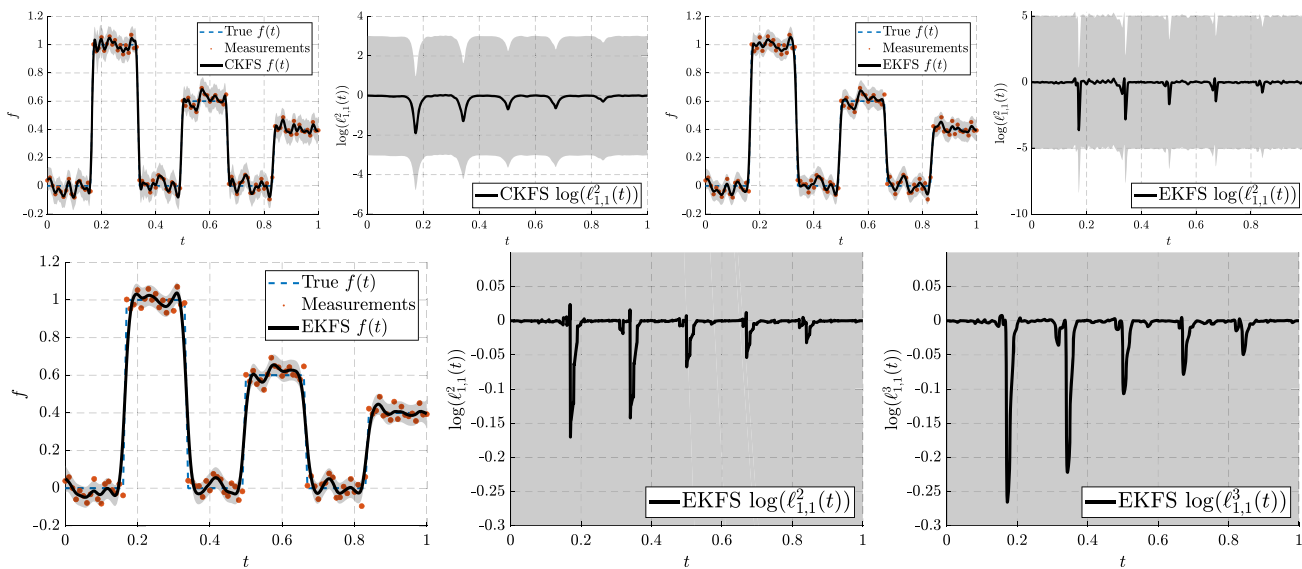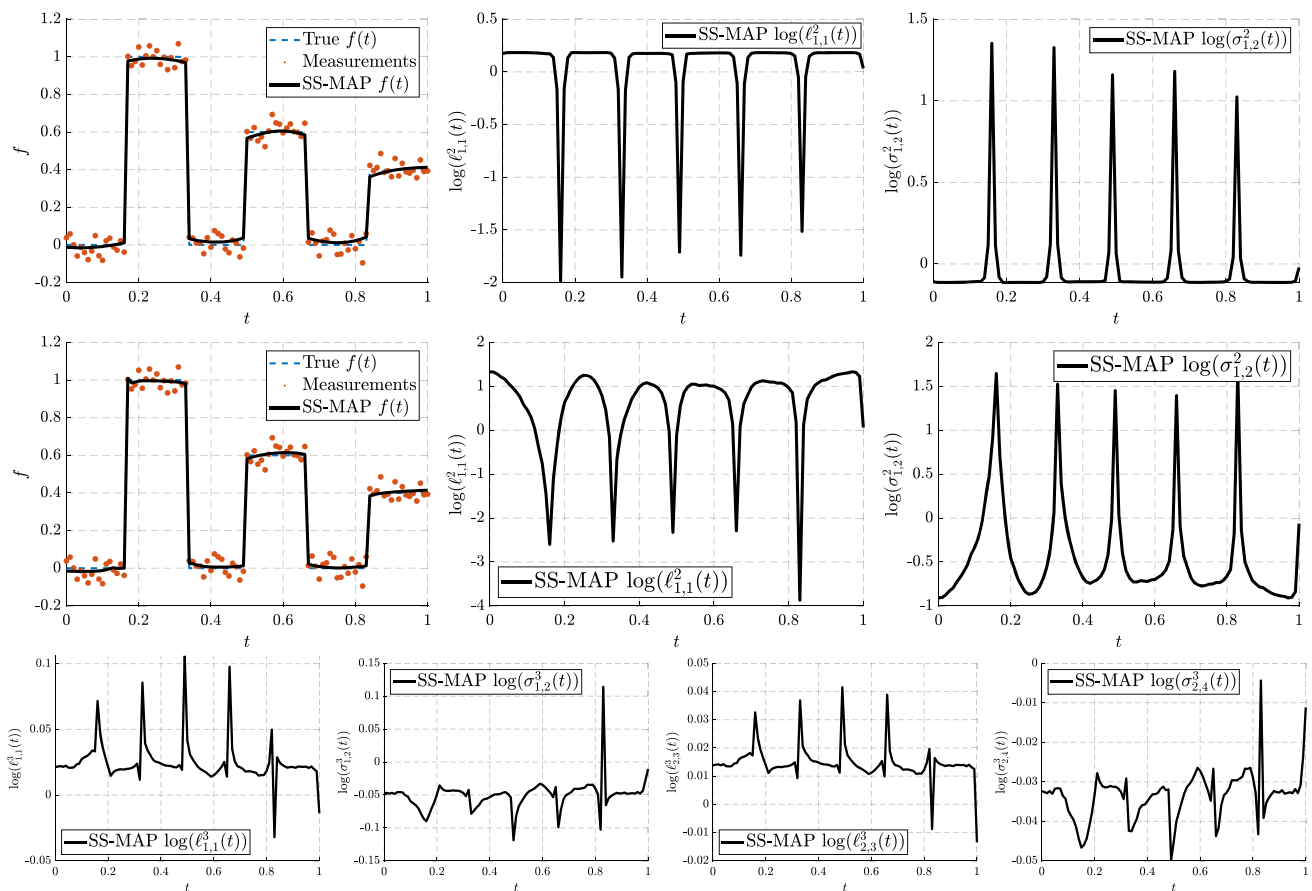

Fig. 6 CKFS and EKFS regression results on model (42) using DGP-2 (first row) and EKFS on DGP-3 (first and second rows). The shaded area stands for 95% confidence interval

Fig. 7 SS-MAP regression results on model (42) using DGP-2 (first row) and DGP-3 (second and third rows)

f is smooth and fits to the actual function to a reasonable extent. For SS-MAP on DGP-2, the estimated $\ell_{1,1}^2$ and $\sigma_{1,2}^2$ mostly stay at a constant level and change rapidly on the leap points. From the second and third rows of Fig. 7 and Table 1, we see that the SS-MAP achieves better result on DGP-3 compared to on DGP-2. We also find that the learnt parameters $\ell_{1,1}^2$ and $\sigma_{1,2}^2$ of DGP-3 appear to be smoother than of DGP-2.

Apart from the SS-MAP solution to the SS-DGP, we also demonstrate the Bayesian filtering and smoothing solutions in Figs. 8, 6, and 9 . Figure 8 shows the results of CKFS on DGP-2. We find that the regression result on DGP-2 is acceptable though the estimate is overly smooth on the jump points. The learnt parameters $\ell_{1,1}^2$ also change significantly on the jump points and stay flat elsewhere. Moreover, we find that the estimated $\log(\sigma_{1,2}^2)$ and $\mathrm{Cov}[f, \sigma_{1,2}^2 \mid \mathbf{y}_{1:k}]$ converge to zero in very fast speeds, especially the covariance estimate. This phenomenon resembles the vanishing covariance in Theorem 1. In this case, the estimated $\log(\sigma_{1,2}^2)$ converges to the prior mean of SS-DGP which is zero, due to the vanishing covariance. Therefore for this experiment and all the following experiments, we treat all the magnitude parameters of Matérn (e.g., $\sigma_{1,2}^2$) as trainable hyperparam-

eters learnt from grid searches. The results are illustrated in Fig. 6. However, we identify that there is a numerical difficulty when applying CKFS on DGP-3. With many hyperparameter settings, the CKFS fails due to numerical problems (e.g., singular matrix). The EKFS still works on DGP-3, thus we plot the results in the second row of Fig. 6. The estimated f of EKFS appears to be over-smooth, especially on the jump points. Also, the estimated variances of $\ell_{1,1}^2$ and $\sigma_{1,2}^2$ are significantly large.

Figure 9 illustrates the result of PF-BS. We find that the regression results are reasonably close to the ground truth. Also, the estimated f is smooth. The estimated parameters $\ell_{1,1}^2$ and $\sigma_{1,2}^2$ for PF-BS on DGP-2 have a similar pattern as the results of SS-MAP, CKFS, and EKFS, which only change abruptly on the jump points. However, the $\ell_{1,1}^2$ of DGP-3 does not stay flat generally, and $\sigma_{1,2}^2$ does not change significantly on the jump points. In Table 1, the RMSEs of PF-BS on DGP-3 are better than on DGP-2. Also, PF-BS is slightly better than PF.

We now summarize the numerical results in terms of the RMSEs, NLPD, and computational time from Table 1. Table 1 demonstrates that the DGP methods using MAP, PF, and PF-BS outperform GP and NS-GP on this non-stationary
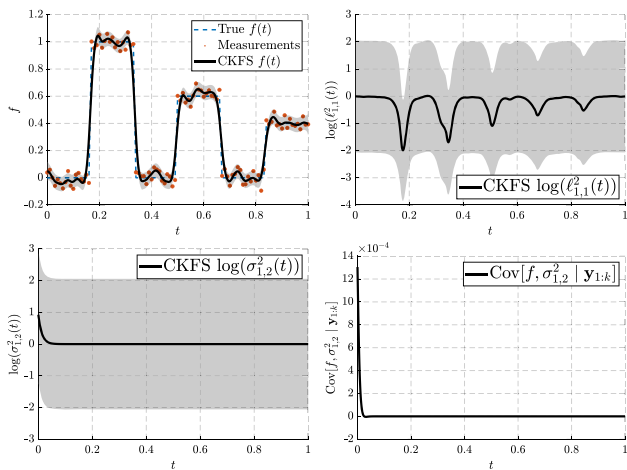
Fig. 8 CKFS regression on model (42) using DGP-2

Fig. 10 Demonstration of the composite sinusoidal signal (45)

signal regression. Moreover, the RMSEs and NLPDs are improved by using DGP-3 over DGP-2, except for Gaussian filters and smoothers. Among all regression methods, the SS-MAP is the best in terms of RMSE, followed by B-MAP and PF-BS. In terms of NLPD, PF-BS admits the lowest
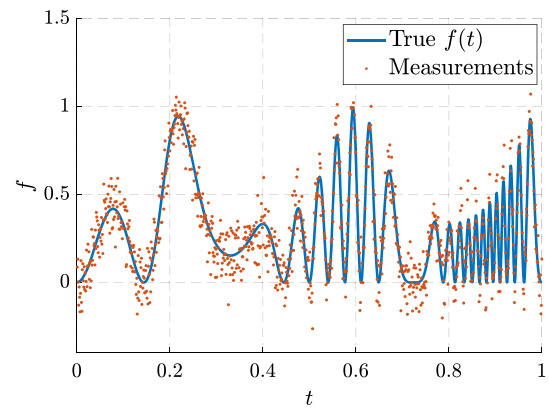
value. However, the NLPD and RMSE results of PF and PF-BS have very large deviations which are improved by using DGP-3 over DGP-2. We found that the Gaussian filters and smoothers (CKFS and EKFS) are the fastest, followed by GP and NS-GP. We also notice that for all methods, DGP-3 is more time-consuming than DGP-2. Even though we implemented PF-BS in CPU-based parallelization the time
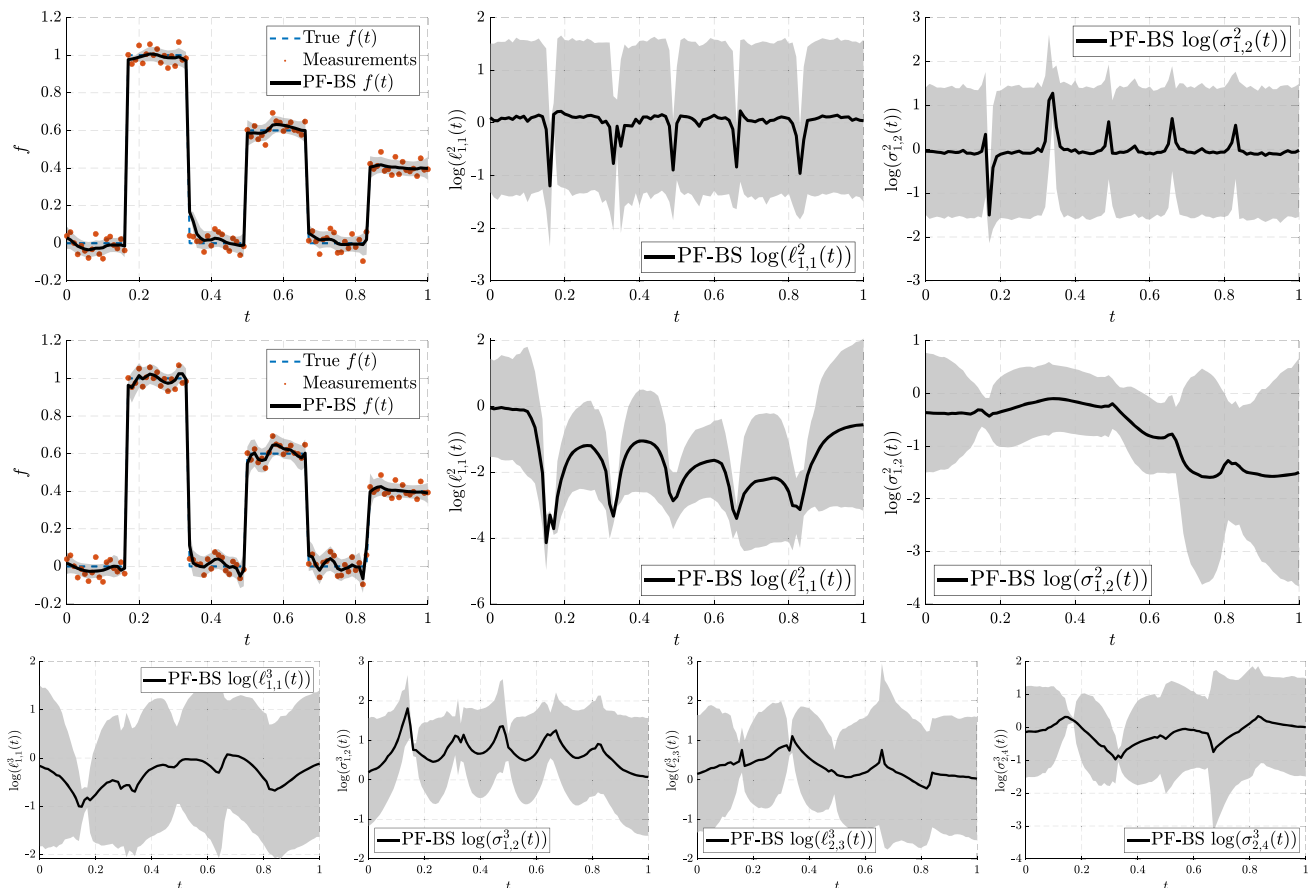
Fig. 9 PF-BS regression results on model DGP-2 (first row) and DGP-3 (second and third rows). The shaded area stands for 95% confidence interval
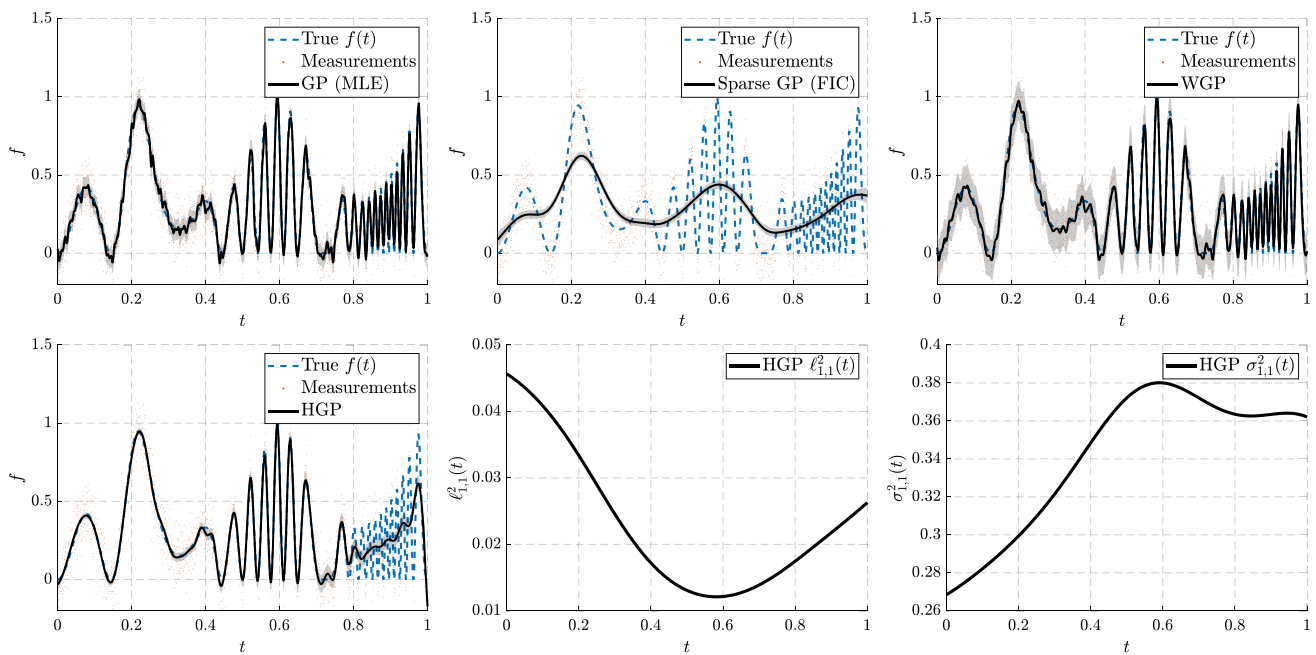
Fig. 11 Regression results of GP, Sparse GP, WGP, and HGP on model (45). The shaded area stands for 95% confidence interval

7.2 Regression on composite sinusoidal signal

In this section, we conduct another experiment on a non-stationary composite sinusoidal signal formulated by

$$f(t) = \frac{\sin^2\left(7\pi\,\cos\left(2\pi\,t^2\right)\,t\right)}{\cos\left(5\pi\,t\right)+2}, \quad t \in [0,1], \tag{45}$$
$$y(t) = f(t) + r(t),$$

where f is the true function, and $r(t) \sim \mathcal{N}(0, 0.01)$. This type of signal has been used by, for example, Rudner et al. (2020); Vannucci and Corradi (1999) and Monterrubio-Gómez et al. (2020). A demonstration is plotted in Fig. 10. In contrast to the discontinuous rectangle wave in Eq. (42), this composite sinusoidal is smooth. Thus it is appropriate to postulate a smooth Matérn prior. This non-stationary signal is challenging in the sense that the frequencies and magnitudes are changing rapidly over time.

The settings of this experiment are the same with the rectangle wave regression in Sect. 7.1, except that we generate the signal with 2, 000 samples. With this number of measurements, the NS-GP and MAP-based solvers fail because they do not converge in a reasonable amount of time. Also, we select three other GP models from the literature for com-

parison, that are, the fully independent conditional (FIC, Quinonero-Candela and Rasmussen 2005) sparse GP with 500 pseudo-inputs, the warped GP (WGP, Snelson et al. 2004), and a non-stationary GP (HGP) by Heinonen et al. (2016).

The results for GP, Sparse GP, WGP, and HGP are shown in Fig. 11. We find that the estimate of GP is overfitted to the measurements, and it is not smooth. On the contrary, the estimate of sparse GP is underfitted. The result of WGP is similar to GP, but the estimated variance of WGP is large. The HGP works well except for the part after $t > 0.8$ s. The learnt $\ell^2_{1,1}$ and $\sigma^2_{1,2}$ from HGP are smooth.

Figures 12 and 13 plot the results of EKFS and CKFS, respectively. From visual inspection, the Gaussian filters and smoothers based DGPs outperform GP, sparse GP, WGP, and HGP. We also find that the estimates from EKFS and CKFS are quite similar, whereas EKFS gives smoother estimate of f compared to CKFS. The learnt $\ell^2_{1,1}$ and $\sigma^2_{1,2}$ also adapt to the frequency changes of the signal. It is worth noticing that the estimated $\ell^3_{1,1}$ in the third layer of DGP-3 is almost flat for both CKFS and EKFS.

The RMSE, NLPD, and computational time are listed in Table 2. This table verifies that the DGPs using Gaussian filters and smoothers (i.e., CKFS and EKFS) outperform other methods in terms of RMSE, NLPD, and computational time. Also, CKFS gives slightly better RMSE and NLPD than EKFS. For this signal, using DGP-3 yields no better RMSE and NLPD compared to DGP-2.
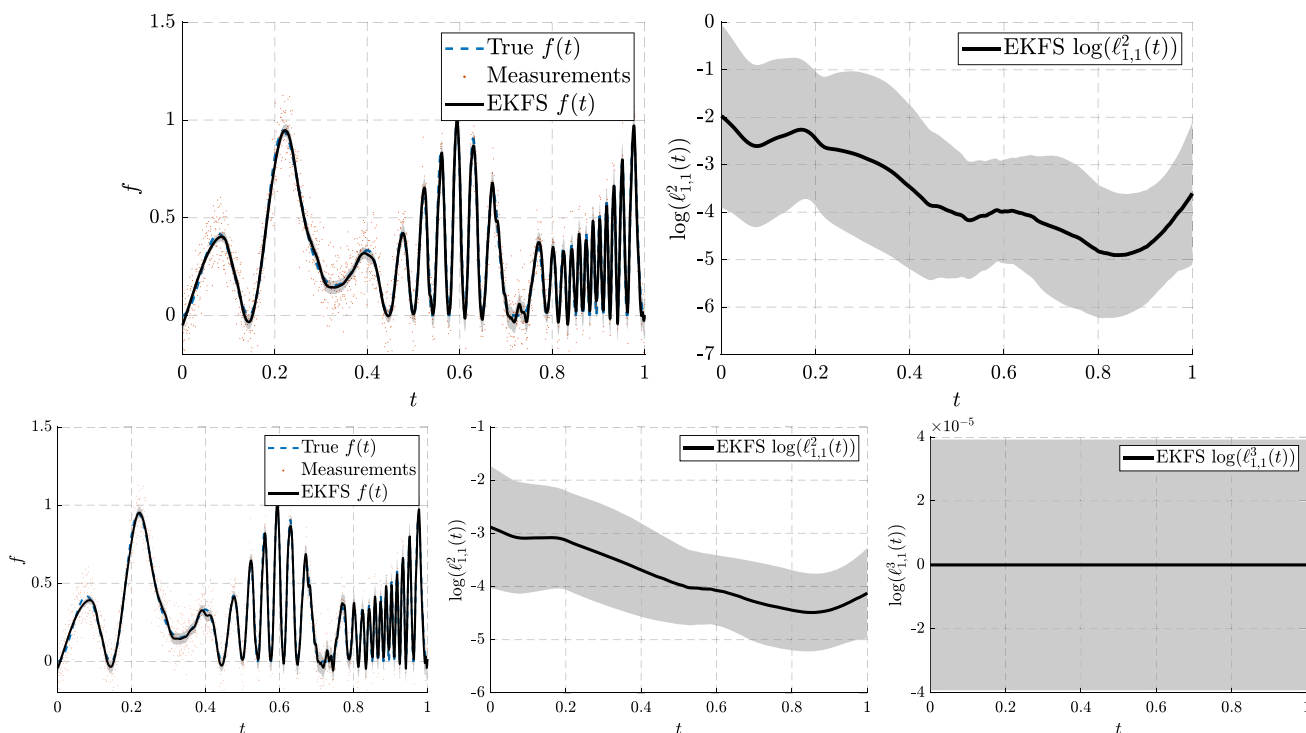
Fig. 12 EKFS regression results on model (45) using DGP-2 (first row) and DGP-3 (second row). The shaded area stands for 95% confidence interval
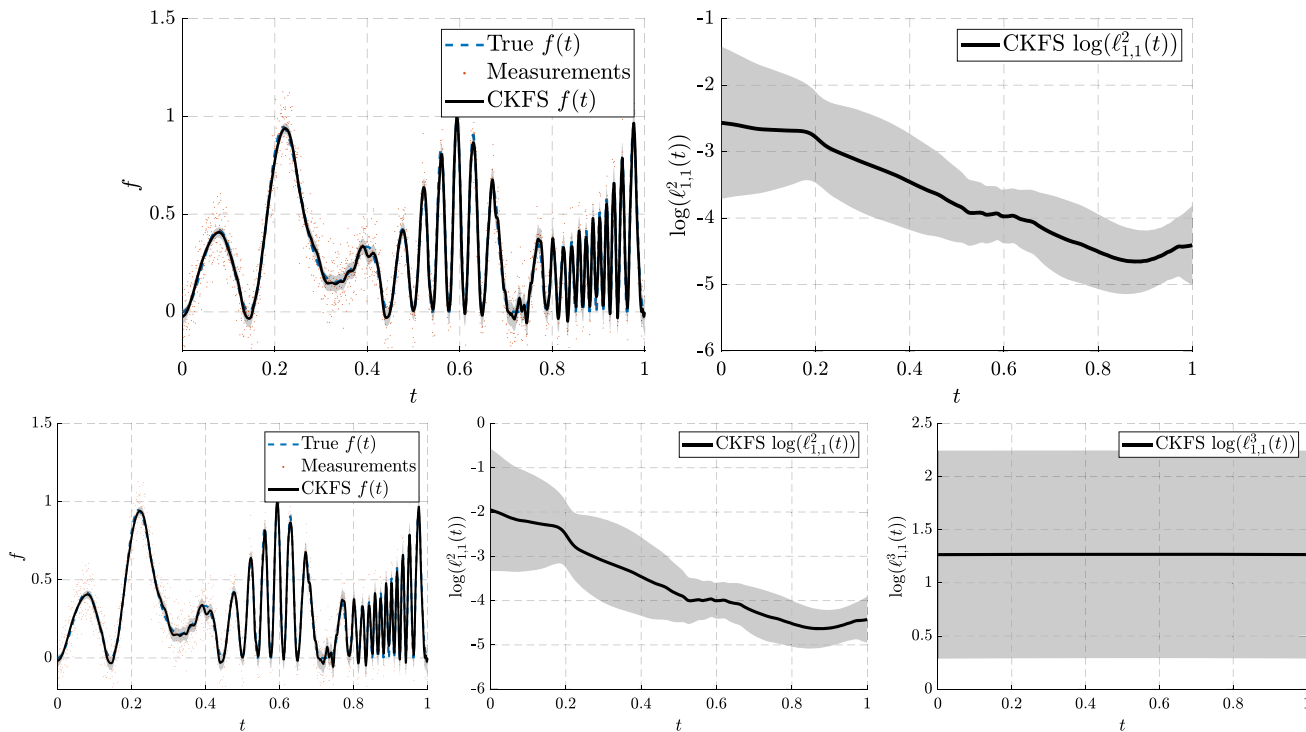
Fig. 13 CKFS regression results on model (45) using DGP-2 (first row) and DGP-3 (second row). The shaded area stands for 95% confidence interval

Table 2 Averaged RMSEs, NLPD, and computational time (in seconds) on model (45) over different regression models and solvers

Methods	RMSE (0^{-2})	NLPD (0^3)	Time (s)
GP (MLE)	3.08 ± 0.1	-1.69 ± 0.03	1.4 ± 0.5
Sparse GP (FIC)	17.52 ± 1.9	1.24 ± 0.71	0.3 ± 0.1
WGP	3.21 ± 2.6	N/A	2.6 ± 1.0
HGP	9.35 ± 1.7	-0.93 ± 0.36	765.5 ± 109
CKFS (DGP-2)	$\mathbf{2.52} \pm 0.2$	$\mathbf{-1.71} \pm 0.03$	0.3 ± 0.1
CKFS (DGP-3)	2.54 ± 0.2	-1.70 ± 0.03	0.9 ± 0.2
EKFS (DGP-2)	2.61 ± 0.2	-1.70 ± 0.03	$\mathbf{0.1} \pm 0.03$
EKFS (DGP-3)	2.73 ± 0.2	-1.70 ± 0.03	0.2 ± 0.02

Bold represent the best values in each of their columns

7.3 Real data application on LIGO gravitational wave detection

The theoretical existence of gravitational waves was predicted by Albert Einstein in 1916 from a linearized field equation of general relativity (Hill et al. 2017; Einstein and Rosen 1937). In 2015, the laser interferometer gravitational-wave observatory (LIGO) team made the first observation of gravitational waves from a collision of two black holes, known as the event GW150914 (Abbott et al. 2016). The detection was originally done by using a matched-filter approach. It is of our interests to test if the GP and DGP approaches can detect the gravitational waves from the LIGO measurements. We now formulate the detection as a regression task.

We use the observation data provided by LIGO scientific collaboration and the Virgo collaboration.[2] As shown in the first picture of Fig. 14, the data contains 3,441 measurements sampled in frequency of 16,384 Hz. We use time interval 10^{-5} s to interpolate the data, which results in 10, 499 time steps. The reference gravitational wave calculated numerically from the general relativity theory is shown in Fig. 14, and we use it as the ground truth for comparison.

We use the previously formulated regression models GP and DGP-2, as shown in Fig. 2. Unfortunately, the NS-GP and MAP-based solvers are not applicable due to a large number of observations and interpolation steps. Hence, we choose the Gaussian filters and smoothers (i.e., CKFS and EKFS) for DGP regression.

The detection results are shown in the second and third rows of Fig. 14. We find that the DGP-2 model gives a better fit to the gravitational wave compared to GP. The DGP-2 estimate is almost identical to the numerical relativity result. GP however, fails because the estimate overfits to the measurements. Also, the outcomes of DGP-2 are explainable by

reviewing the learnt parameter $\ell_{1,1}^2$. We see that the length scale $\ell_{1,1}^2$ adapts to the frequency changes of the gravitational wave, which is an expected feature by using the DGP model. The results of CKFS and EKFS are similar, while EKFS gives smoother results.

Moreover, the Gaussian filters and smoothers on DGP-2 have significantly smaller time consumption compared to GP. In one single run of the program, CKFS and EKFS take 1.5 s and 0.4 s, respectively, while GP takes 202.2 s (including hyperparameter optimization).
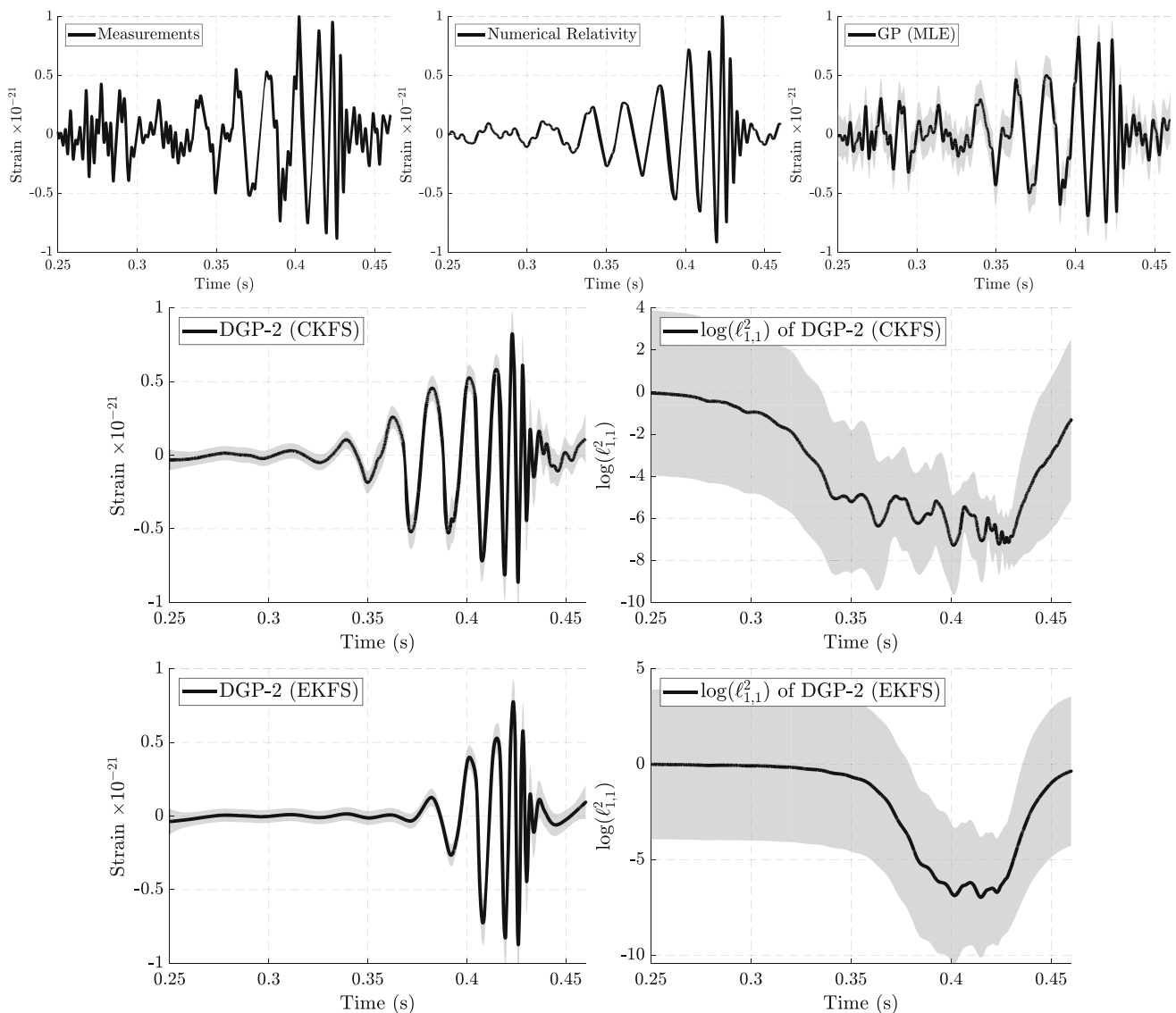
7.4 Summary of experimental results

In this section, we summarize the results of the state-space methods presented in the sections above. In the rectangular signal regression experiment, the state-space MAP and particle smoothing methods are better than Gaussian smoothers (e.g., EKFS and CKFS) in terms of RMSE and NLPD. Based on the results of the composite sinusoidal signal regression experiment, Gaussian smoothers are particularly efficient in computation. However, Gaussian smoothers may not be suitable solvers for SS-DGP models that have both lengthscale and magnitude parameters included in the DGP hierarchy. This is proved in Sect. 6, and it is also numerically shown in Fig. 8.

8 Conclusion

In this paper, we have proposed a state-space approach to deep Gaussian process (DGP) regression. The DGP is formulated as a cascaded collection of conditional Gaussian processes (GPs). By using the state-space representation, we cast the DGP into a non-linear hierarchical system of linear stochastic differential equations (SDEs). Meanwhile, we propose the maximum a posteriori and Bayesian filtering and smoothing solutions to the DGP regression task. The experiment shows significant benefits when applying the DGP methods to simulated non-stationary regression problems as well as to a real data application in gravitational wave detection.

The proposed state-space DGPs (SS-DGPs) have the following major strengths. The DGP priors are capable of modeling larger classes of functions compared to the conventional and non-stationary GPs. In the construction of state-space DGP, one does not need to choose/design valid covariance functions manually like in Paciorek and Schervish (2006) or Salimbeni and Deisenroth (2017b). In DGP regression in state-space form we do not need to evaluate the (full) covariance function either. Moreover, state-space methods are particularly efficient for temporal data as they have linear computational complexity with respect to time.

---

[2] The data is available at https://doi.org/10.7935/K5MW2F23 or https://doi.org/10.7935/82H3-HH23.

Fig. 14 LIGO gravitational wave detection (event GW150914, Hanford, Washington) using (Matérn, $\alpha = 1$) GP and DGP-2. The shaded area stands for 95% confidence interval

In addition, we have identified a wide class of SS-DGPs that are not suitable for Gaussian smothers to solve. More specifically, these SS-DGP models are the ones that have both their lengthscale and magnitude parameters modeled as GP nodes under the assumptions in Section 6. When applying Gaussian smoothers on these SS-DGPs, their Kalman gains converge to zero as time goes to infinity, which makes Gaussian smoothers use no information from data to update their posterior distributions. This is one limitation of SS-DGPs. Although one can use the MAP and particle smoothing methods in place of Gaussian smoothers, these methods can be computationally demanding.

For future investigation, enabling automatic differentiations is of interests. In this paper we have only applied grid search on a large number of trainable hyperparameters which results in a very crude optimization. By using libraries like TensorFlow or JAX we can also obtain Hessians which we can use to quantify the uncertainty in MAP.

Another useful future extension is to exploit data-scalable inference methods, such as sparse variational methods. For example, Chang et al. (2020) solve state-space GP regression problems (possibly with non-Gaussian likelihoods) by using a conjugate variational inference method while still retaining a linear computational complexity in time. Their work is extended by Wilkinson et al. (2021) who introduce sparse inducing points to the said variational state-space GP inference, resulting in a computational complexity that is linear in the number of inducing points. Although these works are mainly concerned with standard state-space GPs (i.e., linear state-space models), it would be possible to apply these

methods on SS-DGPs as well, for example, by linearizing the state-space models of SS-DGPs.

Generalizing the temporal SS-DGPs to spatio-temporal SS-DGPs (see, the end of Section 4.2) would be worth studying as well, by extending the methodologies introduced in Särkkä et al. (2013); Emzir et al. (2020).

Appendix A Derivatives of Loss Function (6)

We define the derivatives in a set

$$\frac{\partial \mathcal{L}^{\text{BMAP}}}{\partial U_{1:N}} = \left\{ \frac{\partial \mathcal{L}}{\partial u^i_{j,k|1:N}} : i = 1, \ldots, L, k = 1, 2, \ldots, L_i \right\}$$

for all nodes, where each element is a column vector. For the top GP $f := u^1_{1,1}$, the derivative is

$$\frac{\partial \mathcal{L}^{\text{BMAP}}}{\partial f_{1:N}} = -\mathbf{R}^{-1} (\mathbf{y}_{1:N} - f_{1:N}) + \mathbf{C}^{-1} f_{1:N}.$$

The derivatives of other GP nodes are given by

$$\frac{\partial \mathcal{L}^{\text{BMAP}}}{\partial u^i_{j,k|1:N}} = \frac{\partial - \log p(u^i_{j,k|1:N} \mid U^{i+1}_{k,\cdot|1:N})}{\partial u^i_{j,k|1:N}}$$
$$+ \frac{\partial - \log p(u^{i-1}_{j,k|1:N} \mid U^i_{k,\cdot|1:N})}{\partial u^i_{j,k|1:N}}$$
$$= \left(\mathbf{C}^i_k\right)^{-1} u^i_{j,k|1:N} + \frac{1}{2} \mathbf{g}^i_k.$$

Above, the m-th element of $\mathbf{g}^i_k \in \mathbb{R}^N$ is

$$\left[\mathbf{g}^i_k\right]_m = \frac{\partial \mathcal{L}^{\text{BMAP}}}{\partial u^i_{j,k|m}}$$
$$= \frac{\partial - \log \left| 2\pi \, \mathbf{C}^{i-1}_k \right|}{\partial u^i_{j,k|m}}$$
$$+ \frac{\partial - (u^{i-1}_{j,k|1:N})^\mathsf{T} (\mathbf{C}^{i-1}_k)^{-1} u^{i-1}_{j,k|1:N}}{\partial u^i_{j,k|m}}$$
$$= \text{tr} \left[\left(\left(\mathbf{C}^{i-1}_k\right)^{-1} - \boldsymbol{\tau} \, \boldsymbol{\tau}^\mathsf{T} \right) \frac{\partial \mathbf{C}^{i-1}_k}{\partial u^i_{j,k|m}} \right],$$

where $u^i_{j,k|m}$ is the m-th element of $u^i_{j,k|1:N}$ and $\boldsymbol{\tau} = \left(\mathbf{C}^{i-1}_k\right)^{-1} u^{i-1}_{j,k|1:N}$.

Appendix B Derivatives of Loss Function (22)

We collect the derivates of the state in a set

$$\frac{\partial \mathcal{L}^{\text{SMAP}}}{\partial U_{1:N}} = \left\{ \frac{\partial \mathcal{L}^{\text{SMAP}}}{\partial U_k}, k = 0, \ldots, N \right\},$$

for all time step, where each element is a column vector. For the initial condition, its derivative is

$$\frac{\partial \mathcal{L}^{\text{SMAP}}}{\partial U_0} = \mathbf{P}^{-1}_0 U_0 + \frac{1}{2} \mathbf{z}_0.$$

For $k = 1, 2, \ldots, N - 1$, the derivative is

$$\frac{\partial \mathcal{L}^{\text{SMAP}}}{\partial U_k} = \frac{1}{R_k} \mathbf{H}^\mathsf{T} (\mathbf{H} U_k - y_k)$$
$$+ \mathbf{Q}^{-1}(U_{k-1}) (U_k - \mathbf{a}(U_{k-1})) + \frac{1}{2} \mathbf{z}_k.$$

Above, $\mathbf{z}_k \in \mathbb{R}^\varrho$ is a vector for $k = 0, 1, \ldots, N - 1$. Now let us temporarily use u^m_k as the m-th component of state U_k, then the m-th element of $\mathbf{z}_k$ is

$$\mathbf{z}_k t_m = -U^\mathsf{T}_{k+1} \mathbf{Q}^{-1}(U_k) \frac{\partial \mathbf{Q}(U_k)}{\partial u^m_k} \mathbf{Q}^{-1}(U_k) U_{k+1} \quad (46)$$
$$+ 2 \frac{\partial \mathbf{a}^\mathsf{T}(U_k)}{\partial u^m_k} \mathbf{Q}^{-1}(U_k) (\mathbf{a}(U_k) - U_{k+1})$$
$$+ \mathbf{a}^\mathsf{T}(U_k) \mathbf{Q}^{-1}(U_k) \frac{\partial \mathbf{Q}(U_k)}{\partial u^m_k} \mathbf{Q}^{-1}(U_k)$$
$$\times (2 U_{k+1} - \mathbf{a}(U_k))$$
$$+ \text{tr} \left(\mathbf{Q}^{-1}(U_k) \frac{\partial \mathbf{Q}(U_k)}{\partial u^m_k} \right).$$
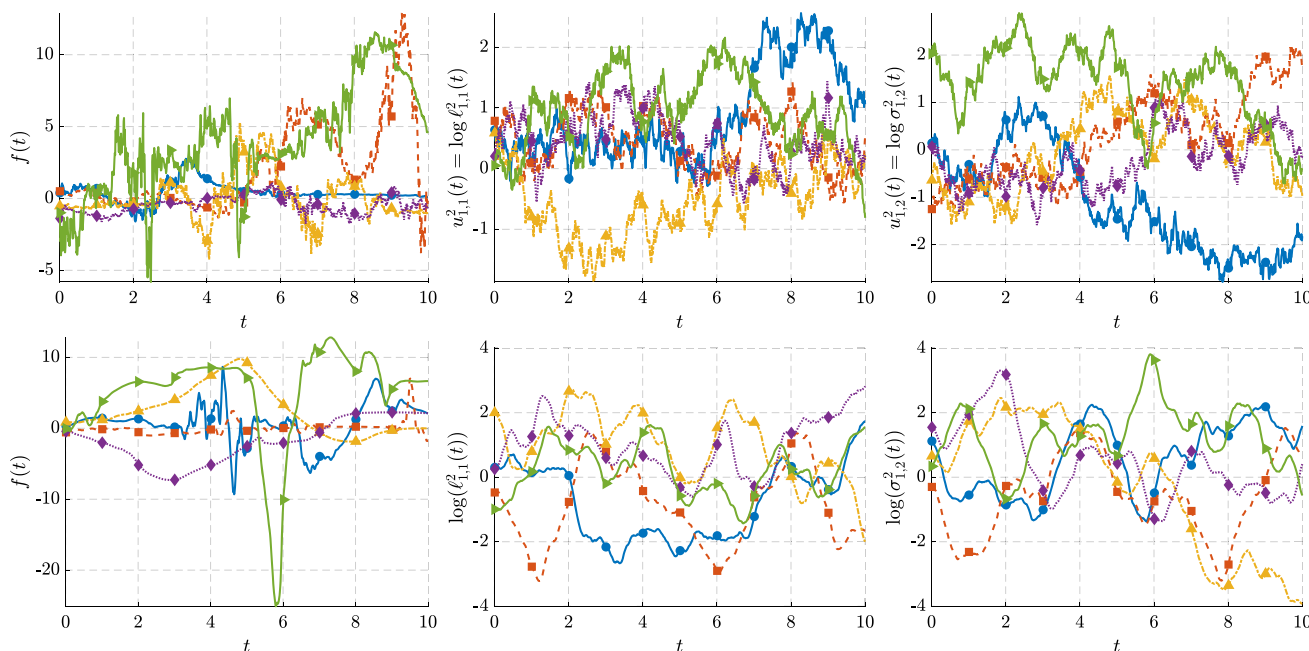
Fig. 15 Samples of DGP-2 models defined in Example 2 (first row) and Fig. 2c (second row)

Finally, for the derivative on the last time step

$$\frac{\partial \mathcal{L}^{\text{SMAP}}}{\partial \mathbf{U}_N} = \frac{1}{R_N} \mathbf{H}^\mathsf{T} (\mathbf{H} \mathbf{U}_N - y_N) + \mathbf{Q}^{-1}(\mathbf{U}_{N-1})(\mathbf{U}_N - \mathbf{a}(\mathbf{U}_{N-1})).$$

(47)

Appendix C Derivation of Eq. (36)

Let us denote by

$$\bar{P}_k = \begin{bmatrix} \bar{P}_k^{f,f} & \bar{P}_k^{f,\sigma} \\ \bar{P}_k^{f,\sigma} & \bar{P}_k^{\sigma,\sigma} \end{bmatrix}.$$

Then by the update step of Gaussian filters (see, e.g., Algorithm 6.3 of Särkkä 2013), we have

$$S_k = H \, \bar{P}_k \, H^\mathsf{T} + R_k,$$
$$K_k = \bar{P}_k \, H^\mathsf{T}/S_k,$$
$$P_k = \bar{P}_k - K_k \, K_k^\mathsf{T}/S_k,$$

where $H = \begin{bmatrix} 1 & 0 \end{bmatrix}$. Substituting K_k and S_k into P_k gives

$$P_k = \bar{P}_k - \begin{bmatrix} \left(\bar{P}_k^{f,f}\right)^2 & \bar{P}_k^{f,f} \, \bar{P}_k^{f,\sigma} \\ \bar{P}_k^{f,f} \, \bar{P}_k^{f,\sigma} & \left(\bar{P}_k^{\sigma,\sigma}\right)^2 \end{bmatrix} / \left(\bar{P}_k^{f,f} + R_k\right)$$

Hence, the $P_k^{f,\sigma}$ of P_k is

$$P_k^{f,\sigma} = \bar{P}_k^{f,\sigma} - \frac{\bar{P}_k^{f,f} \, \bar{P}_k^{f,\sigma}}{\bar{P}_k^{f,f} + R_k}.$$

Appendix D Samples from DGP Priors and Predictions from DGP Posterior Distributions

To demonstrate the non-stationarity of the DGP models, we draw samples from the DGPs priors defined in Example 2 and Fig. 2c. The samples are drawn by using the TME-3 discretization approach (Zhao et al. 2021) on $t \in [0, 10]$ with time interval $\Delta t = 0.01$ s. We show the samples in Figs. 15, 16 where we can clearly see the non-stationary features of process $f(t)$. The samples also switch the stationary and non-stationary behaviour randomly.

It is also of interests to see how does a fitted DGP model behave in the future (i.e., when extrapolated). For this purpose, we select the fitted CKFS DGP-2 on the sinusoidal experiments as the example. We draw prediction samples starting from the end of the smoothing posterior distribution, and predict until $t = 4$ s. We see that at the beginning ($t = 1$ s) the samples of f retain similar features as the fitted f. As t reaches the end, $f(t)$ is gradually becoming smoother because its lengthscale approach the stationary state.
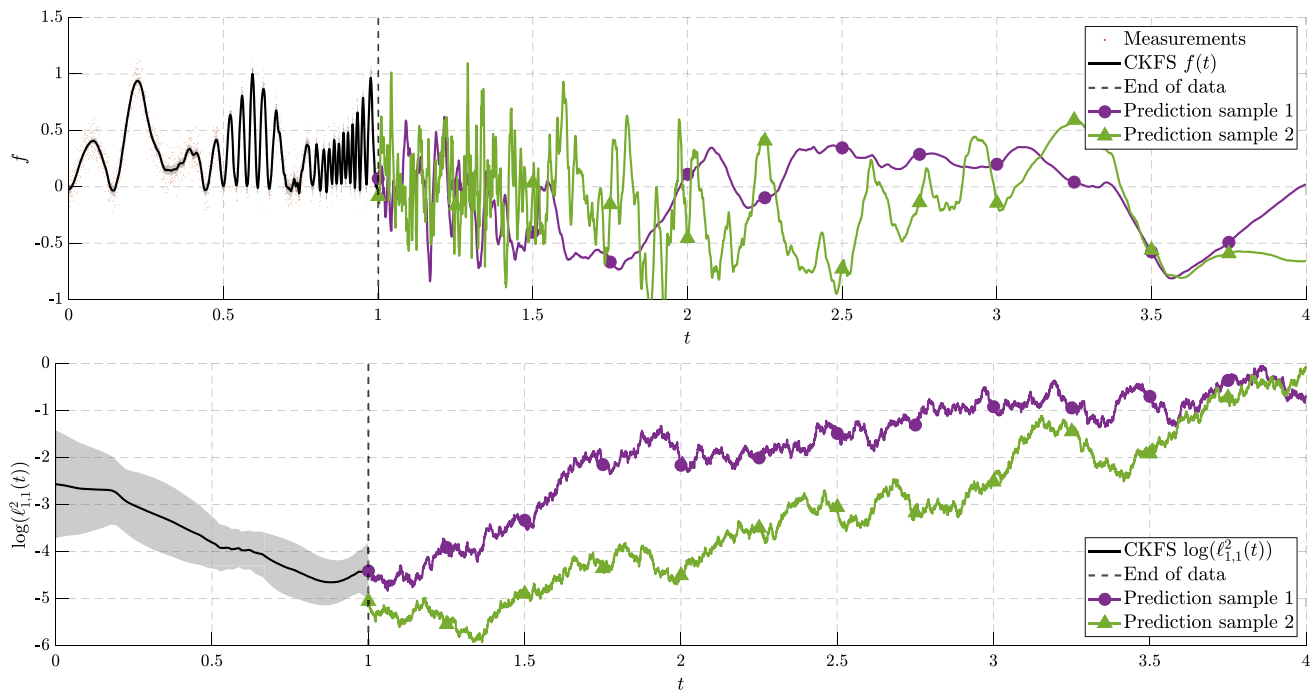
Fig. 16 Prediction samples drawn from the CKFS DGP-2 model (i.e., continuation of Fig. 13). Only two samples are shown for the sake of readability

Table 3 Hyperparameters found via grid search for the rectangle (first block) and sinusoidal (second block) experiments

Method	DGP-2	DGP-3
B-MAP	$\ell = 0.087, \sigma = 0.3$	$\ell = 0.04, \sigma = 0.3$
SS-MAP	$\ell = 0.008, \sigma = 0.14$	$\ell = 0.001, \sigma = 0.92$
EKFS	$\sigma_{I,2}^2 = 2\ell = 0.001, \sigma = 2.1$	$\sigma_{I,2}^2 = 12, \sigma_{I,2}^3 = 0.8 \ \ell = 0.001, \sigma = 9$
CKFS	$\sigma_{I,2}^2 = 2\ell = 0.001, \sigma = 1.54$	N/A
PF-BS	$\ell = 0.008, \sigma = 0.54$	$\ell = 0.098, \sigma = 0.79$
CKFS	$\sigma_{I,2}^2 = 0.4\ell = 2.83, \sigma = 1.49$	$\sigma_{I,2}^2 = 0.4, \sigma_{I,2}^3 = 1.2 \ \ell = 140, \sigma = 0.7$
EKFS	$\sigma_{I,2}^2 = 1.6\ell = 0.23, \sigma = 1.16$	$\sigma_{I,2}^2 = 1.2, \sigma_{I,2}^3 = 0.9\ell = 0.22, \sigma = 0.01$

Appendix E Hyperparameter Values Found via Grid Search

For the sake of reproducibility we list the hyperparameters found by grid search in the following Table 3. Due to a large number of unknown hyperparameters, the grid search routine assumes that GP nodes in the last layer share the same hyperparameters. Hereafter we use notations ℓ and σ to represent the last layer lengthscale and magnitude.

References

Abbott, B.P., et al.: Observation of gravitational waves from a binary black hole merger. Phys. Rev. Lett. **116**(6), 061102 (2016)

Al-Shedivat, M., Wilson, A.G., Saatchi, Y., Hu, Z., Xing, E.P.: Learning scalable deep kernels with recurrent structure. J. Mach. Learn. Res. **18**(82), 1–37 (2017)

Andrieu, C., Doucet, A., Holenstein, R.: Particle Markov chain Monte Carlo methods. J. R. Stat. Soc. Ser. B (Stat. Methodol.) **72**(3), 269–342 (2010)

Bishop, C.M.: Pattern Recognition and Machine Learning. Springer, Berlin (2006)

Brigo, D., Hanzon, B., LeGland, F.: A differential geometric approach to nonlinear filtering: the projection filter. IEEE Trans. Autom. Control **43**(2), 247–252 (1998)

Brooks, S., Gelman, A., Jones, G., Meng, X.L.: Handbook of Markov Chain Monte Carlo. Chapman and Hall/CRC, Cambridge (2011)

Calandra, R., Peters, J., Rasmussen, C.E., Deisenroth, M.P.: Manifold Gaussian processes for regression. In: 2016 International Joint Conference on Neural Networks (IJCNN), pp 3338–3345 (2016)

Chang, P.E., Wilkinson, W.J., Khan, M.E., Solin, A.: Fast variational learning in state-space Gaussian process models. In: 2020 IEEE 30th International Workshop on Machine Learning for Signal Processing (MLSP) (2020)

Damianou, A., Lawrence, N.: Deep Gaussian processes. In: Proceedings of the Sixteenth International Conference on Artificial Intelligence and Statistics, Scottsdale, Arizona, USA, vol. 31, pp. 207–215 (2013)

Doucet, A., Godsill, S., Andrieu, C.: On sequential Monte Carlo sampling methods for Bayesian filtering. Stat. Comput. **10**(3), 197–208 (2000)

Dunlop, M.M., Girolami, M.A., Stuart, A.M., Teckentrup, A.L.: How deep are deep Gaussian processes? J. Mach. Learn. Res. **19**(54), 1–46 (2018)

Duvenaud, D., Rippel, O., Adams, R., Ghahramani, Z.: Avoiding pathologies in very deep networks. Proceedings of the Seventeenth International Conference on Artificial Intelligence and Statistics, PMLR, Reykjavik, Iceland, Proceedings of Machine Learning Research vol. 33, pp. 202–210 (2014)

Einstein, A., Rosen, N.: On gravitational waves. J. Franklin Inst. **223**(1), 43–54 (1937)

Emzir, M., Lasanen, S., Purisha, Z., Roininen, L., Särkkä, S.: Non-stationary multi-layered Gaussian priors for Bayesian inversion. Inverse Prob. **37**(1), 015002 (2020)

Emzir, M.F., Lasanen, S., Purisha, Z., Särkkä, S.: Hilbert-space reduced-rank methods for deep Gaussian processes. In: 2019 IEEE 29th International Workshop on Machine Learning for Signal Processing (MLSP) (2019)

Florens-Zmirou, D.: Approximate discrete-time schemes for statistics of diffusion processes. Statistics **20**(4), 547–557 (1989)

Friedman, A.: Stochastic Differential Equations and Applications. Springer, Berlin (1975)

Garcia-Fernández, A.F., Tronarp, F., Särkkä, S.: Gaussian process classification using posterior linearisation. IEEE Signal Process. Lett. **26**(5), 735–739 (2019)

Glad, T., Ljung, L.: Control Theory: Multivariate and Nonlinear Methods. Taylor & Francis, New York (2000)

Godsill, S.J., Doucet, A., West, M.: Monte Carlo smoothing for nonlinear time series. J. Am. Stat. Assoc. **99**(465), 156–168 (2004)

Goodfellow, I., Bengio, Y., Courville, A.: Deep Learning. MIT Press, Cambridge (2016)

Gordon, N., Salmond, D., Smith, A.: Novel approach to nonlinear/non-Gaussian Bayesian state estimation. IEE Proc. F (Radar and Signal Process.) **140**(2), 107–113 (1993)

Hartikainen, J., Särkkä, S.: Kalman filtering and smoothing solutions to temporal Gaussian process regression models. In: 2010 IEEE International Workshop on Machine Learning for Signal Processing, pp. 379–384 (2010)

Heinonen, M., Mannerström, H., Rousu, J., Kaski, S., Lähdesmäki, H.: Non-stationary Gaussian process regression with Hamiltonian Monte Carlo. In: Proceedings of the 19th International Conference on Artificial Intelligence and Statistics, PMLR, Proceedings of Machine Learning Research, vol 51, pp. 732–740 (2016)

Higdon, D., Swall, J., Kern, J.: Non-stationary spatial modeling. Bayesian. Statistics **6**(1), 761–768 (1999)

Hill, C.D., Nuroski, P., Bieri, L., Garfinkle, D., Yunes, N.: The mathematics of gravitational waves. Notice of the AMS **64**(7), 686–707 (2017)

Itô, K., Xiong, K.: Gaussian filters for nonlinear filtering problems. IEEE Trans. Autom. Control **45**(5), 910–927 (2000)

Jazwinski, A.: Stochastic Processes and Filtering Theory. Academic Press, Cambridge (1970)

Kessler, M.: Estimation of an ergodic diffusion from discrete observations. Scand. J. Stat. **24**(2), 211–229 (1997)

Khasminskii, R.: Stochastic Stability of Differential Equations, 2nd edn. Springer, Berlin (2012)

Kitagawa, G.: Monte Carlo filter and smoother for non-Gaussian nonlinear state space models. J. Comput. Gr. Stat. **5**(1), 1–25 (1996)

Klenke, A.: Probability Theory: A Comprehensive Course, 2nd edn. Springer, London (2014)

Kloeden, P.E., Platen, E.: Numerical Solution of Stochastic Differential Equations. Springer, Berlin (1992)

Koyama, S.: Projection smoothing for continuous and continuous-discrete stochastic dynamic systems. Signal Process. **144**, 333–340 (2018)

Kuo, H.H.: Introduction to Stochastic Integration. Springer, New York (2006)

Kushner, H.J.: Approximations to optimal nonlinear filters. IEEE Trans. Autom. Control **12**(5), 546–556 (1967)

Lazaro-Gredilla, M., Titsias, M.: Variational heteroscedastic Gaussian process regression. In: Proceedings of the 28th International Conference on Machine Learning (ICML-11), ACM, New York, NY, USA, pp 841–848 (2011)

Le, Q.V., Smola, A.J., Canu, S.: Heteroscedastic Gaussian process regression. In: Proceedings of the 22nd International Conference on Machine Learning, Association for Computing Machinery, New York, NY, USA, p 489–496 (2005)

Lindgren, F., Rue, H., Lindström, J.: An explicit link between Gaussian fields and Gaussian Markov random fields: The stochastic partial differential equation approach. J. R. Stat. Soc. Ser. B (Stat. Methodol.) **73**(4), 423–498 (2011)

Luengo, D., Martino, L., Bugallo, M., Elvira, V., Särkkä, S.: A survey of Monte Carlo methods for parameter estimation. EURASIP J. Adv. Signal Process. **25**, 1–62 (2020)

Mao, X.: Stochastic Differential Equations and Applications, 2nd edn. Woodhead Publishing, Oxford (2008)

Monterrubio-Gómez, K., Roininen, L., Wade, S., Damoulas, T., Girolami, M.: Posterior inference for sparse hierarchical non-stationary models. Comput. Stat. Data Anal. **148**, 106954 (2020)

Nocedal, J., Wright, S.J.: Numerical Optimization, 2nd edn. Springer, Berlin (2006)

Øksendal, B.: Stochastic Differential Equations: An Introduction with Applications, 5th edn. Springer, Berlin (2003)

Paciorek, C.J., Schervish, M.J.: Nonstationary covariance functions for Gaussian process regression. In: Advances in Neural Information Processing Systems 16, MIT Press, pp. 273–280 (2004)

Paciorek, C.J., Schervish, M.J.: Spatial modelling using a new class of nonstationary covariance functions. Environmetrics **17**(5), 483–506 (2006)

Quinonero-Candela, J., Rasmussen, C.E.: A unifying view of sparse approximate Gaussian process regression. J. Mach. Learn. Res. 6(Dec):1939–1959 (2005)

Rasmussen, C.E., Williams, C.K.I.: Gaussian Processes for Machine Learning. The MIT Press, Cambridge (2006)

Rogers, C., Williams, D.: Diffusions, Markov Processes, and Martingales, vol. 1, 2nd edn. Cambridge University Press, Cambridge (2000a)

Rogers, C., Williams, D.: Diffusions, Markov Processes, and Martingales, vol. 2, 2nd edn. Cambridge University Press (2000b)

Roininen, L., Girolami, M., Lasanen, S., Markkanen, M.: Hyperpriors for Matérn fields with applications in Bayesian inversion. Inverse Problems Imaging **13**(1), 1–29 (2019)

Rudner, T., Sejdinovic, D., Gal, Y.: Inter-domain deep Gaussian processes with RKHS Fourier features. Proc. Int. Conf. Mach. Learn. **2020**, 10236–10245 (2020)

Salimbeni, H., Deisenroth, M.: Doubly stochastic variational inference for deep Gaussian processes. In: Advances in Neural Information Processing Systems, Curran Associates, Inc., vol 30 (2017a)

Salimbeni, H., Deisenroth, M.P.: Deeply non-stationary Gaussian processes. In: NIPS Workshop on Bayesian Deep Learning (2017b)

Sampson, P.D., Guttorp, P.: Nonparametric estimation of nonstationary spatial covariance structure. J. Am. Stat. Assoc. **87**(417), 108–119 (1992)

Särkkä, S.: Bayesian Filtering and Smoothing. Institute of Mathematical Statistics Textbooks, Cambridge University Press (2013)

Särkkä, S., Hartikainen, J.: Infinite-dimensional Kalman filtering approach to spatio-temporal Gaussian process regression. Proceedings of the Fifteenth International Conference on Artificial

Intelligence and Statistics, La Palma, Canary Islands vol. 22, pp. 993–1001 (2012)

Särkkä, S., Sarmavuori, J.: Gaussian filtering and smoothing for continuous-discrete dynamic systems. Signal Process. **93**(2), 500–510 (2013)

Särkkä, S., Solin, A.: Applied Stochastic Differential Equations. Institute of Mathematical Statistics Textbooks, Cambridge University Press (2019)

Särkkä, S., Solin, A., Hartikainen, J.: Spatiotemporal learning via infinite-dimensional Bayesian filtering and smoothing: A look at Gaussian process regression through Kalman filtering. IEEE Signal Process. Mag. **30**(4), 51–61 (2013)

Shen, Y., Luo, Q., Mao, X.: The improved LaSalle-type theorems for stochastic functional differential equations. J. Math. Anal. Appl. **318**(1), 134–154 (2006)

Snelson, E., Ghahramani, Z., Rasmussen, C.E.: Warped Gaussian processes. In: Advances in Neural Information Processing Systems 16, pp. 337–344. MIT Press (2004)

Tolvanen, V., Jylänki, P., Vehtari, A.: Expectation propagation for non-stationary heteroscedastic Gaussian process regression. In: 2014 IEEE International Workshop on Machine Learning for Signal Processing (MLSP) (2014)

Vannucci, M., Corradi, F.: Covariance structure of wavelet coefficients: theory and models in a Bayesian perspective. J. R. Stat. Soc. Ser. B (Stat. Methodol.) **61**(4), 971–986 (1999)

Wilkinson, W., Solin, A., Adam, V.: Sparse algorithms for Markovian Gaussian processes. In: Proceedings of the 24th International Conference on Artificial Intelligence and Statistics, vol 130, pp 1747–1755 (2021)

Wilson, A.G., Hu, Z., Salakhutdinov, R., Xing, E.P.: Deep kernel learning. In: Proceedings of the 19th International Conference on Artificial Intelligence and Statistics, PMLR, Cadiz, Spain, Proceedings of Machine Learning Research, vol 51, pp 370–378 (2016a)

Wilson, A.G., Hu, Z., Salakhutdinov, R.R., Xing, E.P.: Stochastic variational deep kernel learning. In: Advances in Neural Information Processing Systems, Curran Associates, Inc., vol 29 (2016b)

Xu, D., Yang, Z., Huang, Y.: Existence-uniqueness and continuation theorems for stochastic functional differential equations. J. Differ. Equ. **245**(6), 1681–1703 (2008)

Zhao, Z., Karvonen, T., Hostettler, R., Särkkä, S.: Taylor moments expansion for continuous-discrete Gaussian filtering. IEEE Transactions on Automatic Control In press (2021)

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.