

Gaussian Process Latent Force Models for Learning and Stochastic Control of Physical Systems

Simo Särkkä, *Senior Member, IEEE*, Mauricio A. Álvarez, and Neil D. Lawrence

Abstract

This paper is concerned with estimation and stochastic control in physical systems which contain unknown input signals or forces. These unknown signals are modeled as Gaussian processes (GP) in the sense that GP models are used in machine learning. The resulting latent force models (LFMs) can be seen as hybrid models that contain a first-principles physical model part and a non-parametric GP model part. The aim of this paper is to collect and extend the statistical inference and learning methods for this kind of models, provide new theoretical results for the models, and to extend the methodology and theory to stochastic control of LFMs.

Index Terms

Gaussian process, latent force model, stochastic control, statistical learning, statistical inference.

I. INTRODUCTION

Consider, as an example, a second order differential equation model of a physical system of the form

$$\frac{d^2 f(t)}{dt^2} + \lambda \frac{df(t)}{dt} + \gamma f(t) = u(t) + c(t), \quad (1)$$

Simo Särkkä is with the Department of Electrical Engineering and Automation (EEA), Aalto University, Rakentajanaukio 2c, 02150 Espoo, Finland (simo.sarkka@aalto.fi). Tel. +358 50 512 4393

Mauricio A. Álvarez is with the Department of Computer Science, University of Sheffield, Sheffield, UK S1 4DP

Neil D. Lawrence is with Amazon, Cambridge, UK and the Department of Computer Science, University of Sheffield, Sheffield, UK S1 4DP

Manuscript received XXX; revised XXX.

where the functions $u(t)$ and $f(t)$ are unknown and $c(t)$ is a control function (known or to be optimized). Further assume that we measure the function $f(t)$ via noisy measurements at discrete instants of time t_1, t_2, \dots, t_n :

$$y_k = f(t_k) + \epsilon_k. \quad (2)$$

The aim of this article is to consider the unknown force or input u as unknown and model it as a non-parametric Gaussian process model [1] with a given covariance function in order to estimate, or in machine learning terms, *learn* f and u . The generalizations of this kind of models to arbitrary differential equations are called latent force models (LFM) [2]–[6] in machine learning literature. In addition to learning problem on the LFMs, we also consider the problem of controlling the LFM using the control function $c(t)$. In particular, we consider the problem of optimal stochastic control design for LFMs.

The present problem is also closely related to so called input estimation problem that has previously been addressed in target tracking literature (e.g. [7]) by replacing the input or force with a white or colored noise. However, here we will specifically concentrate on the machine learning point of view which allows for encoding prior information into the driving force as well as the use of modern machine learning methods for coping with the related hyperparameter estimation problems and model extensions.

In addition to (1), other examples of problems of interest are spatio-temporal partial differential equations driven by (structured) Gaussian processes such as the controlled heat equation

$$\begin{aligned} \frac{\partial f(\mathbf{x}, t)}{\partial t} &= D \nabla^2 f(\mathbf{x}, t) - \lambda f(\mathbf{x}, t) + u(\mathbf{x}, t) + c(\mathbf{x}, t), \\ y_k &= f(\mathbf{x}_k, t_k) + \epsilon_k, \end{aligned} \quad (3)$$

where again the aim is to learn both the Gaussian force field $u(\mathbf{x}, t)$ and the function $f(\mathbf{x}, t)$ from the noisy observations y_k as well as design an optimal control for regulating the heat.

A different kind of example of a problem studied in this article is the Poisson equation

$$\begin{aligned} \nabla^2 f(\mathbf{x}) &= u(\mathbf{x}), \\ y_k &= f(\mathbf{x}_k) + \epsilon_k, \end{aligned} \quad (4)$$

where again the function f and the force field u are unknown. The difference is that here is no concept of time in this equation, nor a possibility for controlling the equation. However, the problem of estimating the function f and the force field u from the observations y_k is still well defined.

A. General problem formulation

The models considered in this article can be seen to belong to the following three classes:

- 1) Basic latent force models which are ordinary differential equations (ODEs) driven by Gaussian input processes $u(t)$ and control inputs $c(t)$. They can be written in form

$$\mathcal{L} f(t) = u(t) + c(t), \quad (5)$$

where the linear operator is typically $\mathcal{L} \equiv a_m \frac{d^m}{dt^m} + \dots + a_1 \frac{d}{dt} + a_0$ for some fixed constants $a_i, i = 0, \dots, m$. This kind of models can often be rewritten in *state-space form*

$$\frac{d\mathbf{f}(t)}{dt} = \mathbf{A}_f \mathbf{f}(t) + \mathbf{B}_f \mathbf{u}(t) + \mathbf{M}_f \mathbf{c}(t), \quad (6)$$

which is not only equivalent to the representation above, but strictly more general model formulation than (5); the control $\mathbf{c}(t)$ can be vector-valued and it does not need to enter the system via the same route as the latent force, and the operator \mathcal{L} can also be an integro-differential operator.

- 2) We also consider are dynamic partial and pseudo differential equation (PDE) based models that can generally be written in form

$$\mathcal{L} f(\mathbf{x}, t) = u(\mathbf{x}, t) + c(\mathbf{x}, t), \quad (7)$$

where \mathcal{L} is a linear operator in space and time. The input Gaussian process $u(\mathbf{x}, t)$ and control input $c(\mathbf{x}, t)$ are also space-time processes. Typically, the operator has the form

$$\mathcal{L} \equiv \mathcal{A}_m \frac{d^m}{dt^m} + \dots + \mathcal{A}_1 \frac{d}{dt} + \mathcal{A}_0, \quad (8)$$

where $\mathcal{A}_0, \dots, \mathcal{A}_m$ are some spatial partial differential or pseudo-differential operators. This kind of models can often be also written in form of *spatio-temporal state-space models*

$$\frac{\partial \mathbf{f}(\mathbf{x}, t)}{\partial t} = \mathcal{A}_f \mathbf{f}(\mathbf{x}, t) + \mathbf{B}_f u(\mathbf{x}, t) + \mathbf{M}_f \mathbf{c}(\mathbf{x}, t), \quad (9)$$

which again is strictly more general than the model (8).

- 3) As a static special case of the above models we have the following class of partial and pseudo differential equation (PDE) models driven by spatial Gaussian processes $u(\mathbf{x})$:

$$\mathcal{L} f(\mathbf{x}) = u(\mathbf{x}), \quad (10)$$

where \mathcal{L} is a spatial partial differential operator or pseudo-differential operator. For this kind of models there is no control problem per se, because there is no time dependence. These models do not naturally allow for a state-space representation either.

In each of the models we obtain noisy observations of the form

$$y_k = f(\bullet) + \epsilon_k, \quad (11)$$

where in the case of state-space models we have $f(\bullet) = \mathbf{C}_f \mathbf{f}(\bullet)$ for some given matrix \mathbf{C}_f . Above, the argument \bullet is either $t_k \in \mathbb{R}_+$, $(t_k, x_k) \in \mathbb{R}_+ \times \mathbb{R}^p$, or $x_k \in \mathbb{R}^d$. The measurement noise is assumed to be Gaussian $\epsilon_k \sim \mathcal{N}(0, \sigma^2)$. We also briefly consider a number of generalizations, including operators in the measurement model, non-linear measurement models, non-Gaussian noises, as well as switching dynamic models.

The aim of this article is to consider the problem of inferring the functions $f(\bullet)$ and $u(\bullet)$ from a set of noisy measurements $\{y_k\}$. We consider inference problems both in batch form and with recursive Kalman filters and smoothers. We also consider the parameter estimation problems as well as the optimal control problems (i.e., optimal selection of $\mathbf{c}(\bullet)$) in these models. Some of the inference results have previously been presented in conference and journal articles [2]–[6], but the contribution of this article is to extend the inference methods, give new theoretical results, and extend the methodology to allow for stochastic control of the LFM.

II. LATENT FORCE MODELS AS GAUSSIAN PROCESS REGRESSION WITH SPECIFIC COVARIANCE FUNCTIONS

In this section, we provide a brief introduction to Gaussian process regression, latent force models in covariance-function and state-space forms.

A. Gaussian process regression

In machine learning [1], Gaussian processes (GPs) are commonly used as prior distributions over functions $f : \mathbb{R}^d \rightarrow \mathbb{R}$. When used for regression, the GP encodes the uncertainty we have over a function, before seeing the data. Once we have specified a particular relationship between the function and the data we observe, we can update our beliefs over the function using the Bayes' rule, and compute the posterior distribution. Although the dimensionality of the input domain is often greater than one, that is, $d > 1$, in what follows we use $d = 1$ and refer to it

as the time domain. Later on, the input domain will refer to a purely spatial domain (as in the Poisson equation), or to a spatio-temporal domain, (as in the heat equation).

Let us now consider predicting the value of an unknown function $f(t)$ at a certain test point t^* . The prediction is based on a finite set of previously seen data points or training samples $\mathcal{D} = \{(t_1, y_1), \dots, (t_n, y_n)\}$ which are corrupted with $\epsilon_k \sim \mathcal{N}(0, \sigma^2)$ Gaussian noise $y_k = f(t_k) + \epsilon_k$. Assume that $f(t)$ is modeled as a priori Gaussian process with zero mean and a covariance function $k(t, t')$.

Under the Gaussian process assumption, the joint distribution between y_1, \dots, y_n and $f(t^*)$ follows a Gaussian distribution. Using \mathbf{y} to denote the set of training outputs y_1, \dots, y_n , and \mathbf{t} the set of training inputs t_1, \dots, t_n , we have

$$\begin{pmatrix} \mathbf{y} \\ f(t^*) \end{pmatrix} \sim \mathcal{N} \left(\begin{pmatrix} \mathbf{0} \\ 0 \end{pmatrix}, \begin{pmatrix} \mathbf{K}(\mathbf{t}, \mathbf{t}) + \sigma^2 \mathbf{I} & \mathbf{k}(\mathbf{t}, t^*) \\ \mathbf{k}^\top(\mathbf{t}, t^*) & k(t^*, t^*) \end{pmatrix} \right),$$

where $\mathbf{K}(\mathbf{t}, \mathbf{t}) \in \mathbb{R}^{n \times n}$ is a matrix with elements $\{k(t_i, t_j)\}_{i=1, j=1}^{n, n}$, and $\mathbf{k}(\mathbf{t}, t^*) \in \mathbb{R}^n$ is a vector with elements $\{k(t_i, t^*)\}_{i=1}^n$.

By making use of the properties of multivariate Gaussian distributions, the posterior distribution for $f(t^*)$ given \mathbf{y} follows

$$p(f(t^*) | \mathbf{y}) = \mathcal{N}(f(t^*) | \mu_{f(t^*)|\mathbf{y}}, \sigma_{f(t^*)|\mathbf{y}}^2),$$

where

$$\begin{aligned} \mu_{f(t^*)|\mathbf{y}} &= \mathbf{k}^\top(\mathbf{t}, t^*) [\mathbf{K}(\mathbf{t}, \mathbf{t}) + \sigma^2 \mathbf{I}]^{-1} \mathbf{y}, \\ \sigma_{f(t^*)|\mathbf{y}}^2 &= k(t^*, t^*) - \mathbf{k}^\top(\mathbf{t}, t^*) [\mathbf{K}(\mathbf{t}, \mathbf{t}) + \sigma^2 \mathbf{I}]^{-1} \mathbf{k}(\mathbf{t}, t^*), \end{aligned}$$

being $\mathbf{I} \in \mathbb{R}^{n \times n}$ the identity matrix.

Fig. 1 shows an example of Gaussian process regression. The dashed line is an unknown function $f(t)$. The dots correspond to noisy observed data $\mathcal{D} = \{t_k, y_k\}_{k=1}^n$. The solid line is the posterior mean $\mu_{f(t^*)|\mathbf{y}}$ computed at several test inputs, and the shaded region represents two standard deviations (obtained from $\sigma_{f(t^*)|\mathbf{y}}^2$) away from the posterior mean.

An additional element that is usually considered in GP regression is the estimation of the parameters of the covariance function $k(t, t')$, also known as hyperparameters. However, the GP regression formulation allows for the following simple expression for the likelihood of measurement given the parameters:

$$p(\mathbf{y} | \theta, \sigma^2) = \mathcal{N}(\mathbf{y} | \mathbf{0}, \mathbf{K}(\mathbf{t}, \mathbf{t}; \theta) + \sigma^2 \mathbf{I}). \quad (12)$$

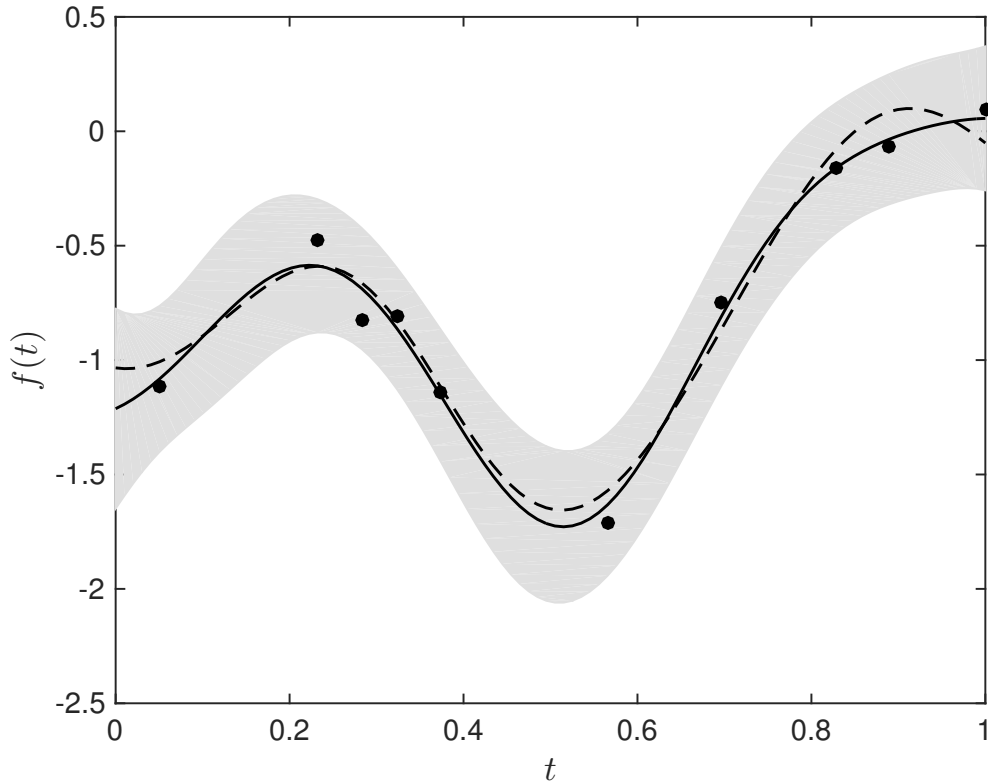


Fig. 1. Example of Gaussian process regression. The dashed line represents the unknown function $f(t)$. The dots are the observed data $\{t_k, y_k\}_{k=1}^n$. The solid line is the posterior mean $\mu_{f(t^*)|\mathbf{y}}$, and the shaded region represents two standard deviations away from the predictive mean.

Given the likelihood there are different ways in which the hyperparameters can be estimated, including maximization of the log-marginal likelihood [1], maximum a posteriori [8], and Markov chain Monte Carlo [9] methods.

B. GP regression solution to basic LFM Inference

Let us now consider the inclusion of a differential equation model to the Gaussian process regression, which leads to a simple latent force model. A canonical example of a linear latent force model (LFM) [2], [4] is the following:

$$u(t) \sim \mathcal{GP}(0, k(t - t')), \quad (13)$$

$$a_m \frac{df^m(t)}{dt^m} + \dots + a_1 \frac{df(t)}{dt} + a_0 = u(t), \quad (14)$$

$$y_k = f(t_k) + \epsilon_k, \quad (15)$$

which contains three parts: a Gaussian process part (13), a differential equation part (14), and a measurement model (15).

Equation (14) can be written as $\mathcal{L} f(t) = u(t)$, where \mathcal{L} is a linear differential operator of the form $\mathcal{L} \equiv a_m \frac{d^m}{dt^m} + \dots + a_1 \frac{d}{dt} + a_0$. The differential operator \mathcal{L} can be related to a so called Green's function or impulse response $h(t)$. On the stationary stage the solution to the differential equation (14) can be expressed as $f(t) = \int_{-\infty}^t h(t-s) u(s) ds$, where $h(t)$ is the impulse response of the system. When $u(s)$ is a Gaussian process with zero mean and covariance function $k(t-t')$ then we can compute the covariance function of the process $f(t)$:

$$k_{f,f}(t, t') = \int_{-\infty}^t \int_{-\infty}^{t'} h(t-s) h(t'-s') k(s-s') ds ds'. \quad (16)$$

Thus we can replace the problem defined by (14) and (13) with an equivalent Gaussian process regression problem

$$f(t) \sim \mathcal{GP}(0, k_{f,f}(t, t')), \quad (17)$$

$$y_k = f(t_k) + \epsilon_k. \quad (18)$$

We may be interested in computing the predictive distribution for $f(t)$ at a test point t^* given the training samples. We can then use the GP regression expressions that we saw in section II-A, where the covariance function is now given by $k_{f,f}(t, t')$.

We may also be interested in computing the posterior distribution for u^* at a test time point t^* given noisy measurements y_1, \dots, y_n at input times t_1, \dots, t_n . To that end, we compute the cross-covariance function between the input process $u(t)$, and the output process $f(t)$, using

$$k_{f,u}(t, t') = \int_{-\infty}^t h(t-s) k(s-t') ds. \quad (19)$$

We assume then that \mathbf{y} and u^* are jointly Gaussian distributed, and their distribution follows

$$\begin{pmatrix} \mathbf{y} \\ u^* \end{pmatrix} \sim \mathcal{N} \left(\begin{pmatrix} \mathbf{0} \\ 0 \end{pmatrix}, \begin{pmatrix} \mathbf{K}_{f,f}(\mathbf{t}, \mathbf{t}) + \sigma^2 \mathbf{I} & \mathbf{k}_{f,u}(\mathbf{t}, t^*) \\ \mathbf{k}_{f,u}^\top(\mathbf{t}, t^*) & k(t^*, t^*) \end{pmatrix} \right),$$

where $\mathbf{K}_{f,f}(\mathbf{t}, \mathbf{t}) \in \mathbb{R}^{n \times n}$ is a matrix with entries given by $\{k_{f,f}(t_i, t_j)\}_{i=1, j=1}^{n, n}$, and $\mathbf{k}_{f,u}(\mathbf{t}, t^*) \in \mathbb{R}^n$ is a vector with entries given by $\{k_{f,u}(t_i, t^*)\}_{i=1}^n$. Notice that due to the assumed independence between $u(t)$ and ϵ_k , the covariance between the process $y(t)$ and $u(t)$ is the same than the covariance between the process $f(t)$, and $u(t)$.

Posterior distribution for u^* given \mathbf{y} follows a Gaussian distribution with mean $\mu_{u^*|\mathbf{y}}$, and variance $\sigma_{u^*|\mathbf{y}}^2$,

$$\begin{aligned}\mu_{u^*|\mathbf{y}} &= \mathbf{k}_{f,u}^\top(t^*, \mathbf{t}) [\mathbf{K}_{f,f}(\mathbf{t}, \mathbf{t}) + \sigma^2 \mathbf{I}]^{-1} \mathbf{y}, \\ \sigma_{u^*|\mathbf{y}}^2 &= k(t^*, t^*) - \mathbf{k}_{f,u}^\top(\mathbf{t}, t^*) [\mathbf{K}_{f,f}(\mathbf{t}, \mathbf{t}) + \sigma^2 \mathbf{I}]^{-1} \mathbf{k}_{f,u}(\mathbf{t}, t^*).\end{aligned}$$

This approach introduced by [2], [4] conveniently reduces the original LFM into an Gaussian process regression problem with a modified covariance function. However, the approach requires that we need to solve the impulse response h and then compute the covariance functions via (16), and (19). If we can obtain closed form expressions for them, then the inference procedure becomes very easy. Otherwise numerical computations are needed.

C. State-space ODE LFM

A convenient way to compute the impulse responses required for the GP inference on LFM is to rewrite the differential equation in state-space form, which allows us to express the impulse response in terms of matrix exponential function. We can rewrite the basic LFM differential equation (14) in state-space form [5], [6] by defining a state vector $\mathbf{f} = (f, df/dt, \dots, d^{m-1}f/dt^{m-1})$ and writing

$$\begin{aligned}\frac{d\mathbf{f}(t)}{dt} &= \underbrace{\begin{pmatrix} 0 & 1 & & \\ & \ddots & \ddots & \\ & & 0 & 1 \\ -a_0 & -a_1 & \dots & -a_{m-1} \end{pmatrix}}_{\mathbf{A}_f} \mathbf{f}(t) + \underbrace{\begin{pmatrix} 0 \\ \vdots \\ 0 \\ \frac{1}{a_m} \end{pmatrix}}_{\mathbf{B}_f} u(t), \\ f(t) &= \underbrace{\begin{pmatrix} 1 & 0 & \dots & 0 \end{pmatrix}}_{\mathbf{C}_f} \mathbf{f}(t).\end{aligned}\tag{20}$$

This gives the following state-space representation for $f(t)$:

$$\begin{aligned}\frac{d\mathbf{f}(t)}{dt} &= \mathbf{A}_f \mathbf{f}(t) + \mathbf{B}_f u(t), \\ f(t) &= \mathbf{C}_f \mathbf{f}(t).\end{aligned}\tag{21}$$

Because this system is of the first order, the impulse response—which is now a matrix valued function—can be written down explicitly in terms of the matrix exponential function $\mathbf{H}(t) = \exp(t \mathbf{A}_f)$ and hence the joint covariance function of \mathbf{f} is

$$\begin{aligned} & \mathbf{K}_f(t - t') \\ &= \int_{-\infty}^t \int_{-\infty}^{t'} \mathbf{H}(t - s) \mathbf{B}_f k(s - s') \mathbf{B}_f^\top \mathbf{H}^\top(t' - s') ds ds', \end{aligned} \quad (22)$$

and we can pick up the covariance function of $f(t)$ from the matrix via

$$k_f(t, t') = \mathbf{C}_f \mathbf{K}_f(t - t') \mathbf{C}_f^\top. \quad (23)$$

The required cross-variance $k_{f,u}$ can be compute by form noting that

$$\mathbf{K}_{f,u}(t - t') = \int_{-\infty}^t \mathbf{H}(t - s) \mathbf{B}_f k(s - t') ds, \quad (24)$$

which further gives

$$k_{f,u}(t, t') = \mathbf{C}_f \mathbf{K}_{f,u}(t - t'). \quad (25)$$

These representations have the advantage that it is easy to numerically approximate the matrix exponential and hence the analytical computation of the impulse response becomes unnecessary. However, we still need to compute the integrals in (22) and (24) somehow. If closed form solution is not available (which surprising often is), then the computations need to be done via numerical integration.

Later in this article we consider state-space methods which also convert the variance function $k(t, t')$ into a (stochastic) differential equation form. In that kind of formulation we are able to avoid the numerical integration altogether and replace it with matrix exponential computations and solutions to Lyapunov equations, which are numerically easier tasks.

D. Partial differential equation based LFMs

Similar to the case of ordinary differential equations, we can associate a Green's function to a linear partial differential operator [10], [11].¹ Let $H(\mathbf{x}, \mathbf{x}')$ be the Green's function for a particular linear partial differential operator, where \mathbf{x} is a general input vector that could include different types of input variables, as we mentioned before. For example, the vector \mathbf{x} could refer

¹The presentation here refers to boundary value problems.

to coordinates x and y for the Poisson equation in two spatial dimensions, or to coordinate x and time t for the Heat equation in one spatial dimension.

For example, the Poisson equation is given as

$$\nabla^2 f(\mathbf{x}) = u(\mathbf{x}).$$

The solution for $f(\mathbf{x})$ has the form

$$f(\mathbf{x}) = \int_{\mathcal{X}} H(\mathbf{x}, \mathbf{z}) u(\mathbf{z}) d\mathbf{z} + g(\mathbf{x}),$$

where \mathcal{X} is the input domain of integration, and $g(\mathbf{x})$ is the solution due to the boundary conditions. For simplicity in the presentation, we assume that the boundary conditions are zero, but the boundary conditions could also be incorporated. In this case the Green's function $H(\mathbf{x}, \mathbf{z})$, with suitable boundary conditions, can be written in closed form. The expression for this Green's function can be found in [11].

For the GP regression view of latent force models, we put a GP prior over $u(\mathbf{x})$,

$$u(\mathbf{x}) \sim \mathcal{GP}(0, k(\mathbf{x} - \mathbf{x}')).$$

Covariance for $f(\mathbf{x})$ then follows a similar expression to (16),

$$k_{f,f}(\mathbf{x}, \mathbf{x}') = \int_{\mathcal{X}} \int_{\mathcal{X}} H(\mathbf{x}, \mathbf{z}) H(\mathbf{x}', \mathbf{z}') k(\mathbf{z} - \mathbf{z}') d\mathbf{z} d\mathbf{z}'.$$

Cross-covariance between the output process $f(\mathbf{x})$, and the input process $u(\mathbf{x})$ can also be computed using a similar expression to (19)

$$k_{f,u}(\mathbf{x}, \mathbf{x}') = \int_{\mathcal{X}} H(\mathbf{x}, \mathbf{z}) k(\mathbf{z} - \mathbf{x}') d\mathbf{z}.$$

Based on a set of noisy training samples, $\mathcal{D} = \{\mathbf{x}_k, y_k\}_{k=1}^n$, we can use the GP regression equations in Sections II-A and II-B, to make predictions for a test input \mathbf{x}^* , both for the output process $f(\mathbf{x})$, and the input process $u(\mathbf{x})$.

As an example of spatio-temporal PDE, the heat equation is given as

$$\frac{\partial f(\mathbf{x}, t)}{\partial t} = D \nabla^2 f(\mathbf{x}, t) - \lambda f(\mathbf{x}, t) + u(\mathbf{x}, t).$$

Let us now assume that we have an initial value problem (i.e., a Cauchy problem) with zero initial condition $f(\mathbf{x}, t_0) = 0$ at a fixed t_0 together with certain boundary conditions on the boundary of a domain \mathcal{X} .

The solution for $f(\mathbf{x}, t)$ can be written as

$$f(\mathbf{x}, t) = \int_{t_0}^t \int_{\mathcal{X}} H(\mathbf{x}, t; \mathbf{z}, s) u(\mathbf{z}, s) d\mathbf{z} ds$$

and again the Green's function H it can be written in closed form. See [11] for the particular expression.

In this case the Green's function is also the kernel of so called operator exponential (exponential semigroup) of the operator:

$$\mathcal{H}(t, s) = \exp \left((t - s) (D \nabla^2 - \lambda) \right). \quad (26)$$

The covariance functions of the process can now be expressed in terms of the Green's function

$$\begin{aligned} k_{f,f}(\mathbf{x}, t; \mathbf{x}', t') &= \int_{t_0}^{t'} \int_{\mathcal{X}} \int_{t_0}^t \int_{\mathcal{X}} H(\mathbf{x}, t; \mathbf{z}, s) H(\mathbf{x}', t'; \mathbf{z}', s') \\ &\quad \times k(\mathbf{z}, s; \mathbf{z}', s') d\mathbf{z} ds d\mathbf{z}' ds', \\ k_{f,u}(\mathbf{x}, t; \mathbf{x}', t') &= \int_{t_0}^t \int_{\mathcal{X}} H(\mathbf{x}, t; \mathbf{z}, s) k(\mathbf{z}, s; \mathbf{x}', t') d\mathbf{z} ds. \end{aligned}$$

The above approach conveniently generalizes to Gaussian process driven partial (or pseudo-differential) equations of the form (cf. [12])

$$\mathcal{D}f(\mathbf{x}, t) = u(\mathbf{x}, t),$$

where \mathcal{D} is a (pseudo) differential operator.

Once we have derived the covariance functions, we can use the Gaussian process regression equations as in Sections II-A and II-B for learning and prediction of functions f and u .

E. State-space PDE LFM

We can also extend the ideas presented in Section II-C to the spatial-temporal case. Assume that that our PDE LFM has the form

$$\frac{\partial \mathbf{f}(\mathbf{x}, t)}{\partial t} = \mathcal{A}_f \mathbf{f}(\mathbf{x}, t) + \mathbf{B}_f u(\mathbf{x}, t), \quad (27)$$

where \mathcal{A}_f is a matrix of linear operators. Let the covariance function of u be $k(\mathbf{x}, \mathbf{x}'; t - t')$. Then, if we define the operator exponential (or the impulse response) as $\mathcal{H}(t) = \exp(t \mathcal{A}_f)$, the covariance function of f can be expressed as

$$k_f(\mathbf{x}, \mathbf{x}'; t - t') = \mathbf{C}_f \mathbf{K}_f(\mathbf{x}, \mathbf{x}'; t - t') \mathbf{C}_f^\top \quad (28)$$

with

$$\begin{aligned} & \mathbf{K}_f(\mathbf{x}, \mathbf{x}'; t - t') \\ &= \int_{-\infty}^t \int_{-\infty}^{t'} \mathcal{H}(t - s) \mathbf{B}_f k(\mathbf{x}, \mathbf{x}'; s - s') \mathbf{B}_f^\top \mathcal{H}^\top(t' - s') ds ds', \end{aligned} \quad (29)$$

where the operator multiplication from right is interpreted as acting onto the variable \mathbf{x}' . The cross-covariance between f and u has the representation

$$k_{f,u}(\mathbf{x}, \mathbf{x}'; t - t') = \mathbf{C}_f \int_{-\infty}^t \mathcal{H}(t - s) \mathbf{B}_f k(\mathbf{x}, \mathbf{x}'; s - t') ds. \quad (30)$$

F. Hyperparameter learning in LFM

Expressions for the covariance functions $k_f(t, t')$, $k_f(\mathbf{x}, \mathbf{x}', t - t')$, and $k_f(\mathbf{x}, \mathbf{x}')$ also involve the parameters of the differential equations, and the parameters of the covariance function of the latent process $u(\cdot)$. Once we have formed the GP model with the modified covariance function, those hyperparameters can be estimated using any of the methods mentioned in the section on GP regression. In the context of latent force models, there have been more recent attempts to estimate these hyperparameters using Markov Chain Monte Carlo, by including priors for the parameters of the differential equations and the covariance function of the latent process [13], [14].

III. LATENT FORCE MODELS AS STOCHASTIC STATE-SPACE MODELS

In this section, we discuss state-space methods for Gaussian process regression and especially show how they can be used with latent force models.

A. Temporal GPs in state-space form

As shown in [15]–[17] many covariance function based models of the form

$$u(t) \sim \mathcal{GP}(0, k(t - t')) \quad (31)$$

can be equivalently or approximately expressed in state-space form by defining a state vector such as $\mathbf{u} = (u, du/dt, \dots, d^{s-1}u/dt^{s-1})$ —note that depending on parameterization the state vector might not contain exactly the derivatives, but a linear transformation of them—leading to a state-space model

$$\begin{aligned}\frac{d\mathbf{u}(t)}{dt} &= \mathbf{A}_u \mathbf{u}(t) + \mathbf{B}_u w(t), \\ u(t) &= \mathbf{C}_u \mathbf{u}(t),\end{aligned}\tag{32}$$

where $w(t)$ is a white-noise process. The advantage of this kind of model formulation is that it allows for solving the GP regression problem using Kalman filters and smoothers [18] in $O(n)$ time when the traditional GP takes $O(n^3)$ time (here n denotes the number of measurements).

The key (classical) result is that such a white-noise-driven state-space representation exists if and only if the spectral density (the Fourier transform of $k(\tau)$) is a proper rational function:

$$S(\omega) = \frac{d_0 + d_1 \omega^2 + \dots + d_r \omega^{2r}}{c_0 + c_1 \omega^2 + \dots + c_s \omega^{2s}}.\tag{33}$$

By using the spectral factorization and classical state-space realizations [16] it is then possible to construct the matrices \mathbf{A}_u , \mathbf{B}_u , and \mathbf{C}_u such that $u(t)$ has the desired covariance function.

For example, the Matérn covariance functions $k(\tau) = (\sigma^2 2^{1-\nu}/\Gamma(\nu)) (\sqrt{2\nu} \tau/\ell)^\nu K_\nu(\sqrt{2\nu} \tau/\ell)$ with certain parameter values ($\nu = 1/2, 3/2, \dots$) have exact state-space representations. In particular, with $\nu = 3/2$ we get

$$\begin{aligned}\frac{d\mathbf{u}(t)}{dt} &= \underbrace{\begin{pmatrix} 0 & 1 \\ -\lambda^2 & -2\lambda \end{pmatrix}}_{\mathbf{A}_u} \mathbf{u}(t) + \underbrace{\begin{pmatrix} 0 \\ 1 \end{pmatrix}}_{\mathbf{B}_u} w(t), \\ u(t) &= \underbrace{\begin{pmatrix} 1 & 0 \end{pmatrix}}_{\mathbf{C}_u} \mathbf{u}(t),\end{aligned}\tag{34}$$

where $\lambda = \sqrt{2\nu}/\ell$.

Although the commonly used squared exponential covariance function does not have an exact state-space representation, it can be easily approximated as such using a simple Taylor series expansions or Padé approximants in Fourier domain [17], [19]. State-space representations of periodic covariance functions have been considered in [20], and rational quadratic and general scale mixtures of Gaussians in [21]. A review of different representations and approximations can be found in [22].

To some extent it is also possible to convert non-stationary covariance function into state space form. However, in that case the Fourier domain approach above is not applicable. For example, Wiener processes and n -times integrated Wiener processes are non-stationary Gaussian processes, but they still have easy state-space representation (see, e.g., [22]). There also exists some general procedures (e.g., [23], [24]) for converting non-stationary covariance functions into state-space form.

B. Spatio-temporal GPs in state-space form

It is also possible to extend the above concepts to spatio-temporal covariance functions of the form $k(\mathbf{x} - \mathbf{x}', t - t')$, which in turn gives covariance functions $k(\mathbf{x} - \mathbf{x}')$ as a special case, because we can always reinterpret one of the components of \mathbf{x} as time. The methods to form these infinite-dimensional state-space representations have been considered in [16], [25].

The conversion of a spatio-temporal covariance function into state-space form leads to a system of the form

$$\begin{aligned} \frac{\partial \mathbf{u}(\mathbf{x}, t)}{\partial t} &= \mathcal{A}_u \mathbf{u}(\mathbf{x}, t) + \mathbf{B}_u w(\mathbf{x}, t), \\ u(\mathbf{x}, t) &= \mathbf{C}_u \mathbf{u}(\mathbf{x}, t), \end{aligned} \tag{35}$$

where \mathcal{A}_u is a matrix of linear operators (typically pseudo-differential operators) acting on the \mathbf{x} -variable and $w(\mathbf{x}, t)$ is a time-white spatio-temporal Gaussian process.

The representation can be obtained, for example, by an extension of the spectral factorization based method to the temporal case. We can first form the joint space-time spectral density $S(\boldsymbol{\omega}_x, \omega_t)$ and approximate it as

$$S(\boldsymbol{\omega}_x, \omega_t) \approx \frac{d_0(\boldsymbol{\omega}_x) + d_1(\boldsymbol{\omega}_x) \omega_t^2 + \cdots + d_r(\boldsymbol{\omega}_x) \omega_t^{2r}}{c_0(\boldsymbol{\omega}_x) + c_1(\boldsymbol{\omega}_x) \omega_t^2 + \cdots + c_s(\boldsymbol{\omega}_x) \omega_t^{2s}}. \tag{36}$$

The spectral factorization in variable ω_t then leads to a temporal state space model, where the matrices depend on the spatial frequency $\boldsymbol{\omega}_x$. A spatial inverse Fourier transform then transforms the terms into linear operators leading to a model of the form (35). More details can be found in [16].

For example, the spatio-temporal Matérn covariance function with a suitable parametrization leads to the following representation [16]:

$$\begin{aligned} \frac{\partial \mathbf{u}(\mathbf{x}, t)}{\partial t} &= \underbrace{\begin{pmatrix} 0 & 1 \\ \nabla^2 - \lambda^2 & -2\sqrt{\lambda^2 - \nabla^2} \end{pmatrix}}_{\mathcal{A}_u} \mathbf{u}(\mathbf{x}, t) + \underbrace{\begin{pmatrix} 0 \\ 1 \end{pmatrix}}_{\mathbf{B}_u} w(\mathbf{x}, t), \\ u(\mathbf{x}, t) &= \underbrace{\begin{pmatrix} 1 & 0 \end{pmatrix}}_{\mathbf{C}_u} \mathbf{u}(\mathbf{x}, t), \end{aligned} \quad (37)$$

where $\nabla^2 = \partial^2/\partial x_1^2 + \dots + \partial^2/\partial x_{d-1}^2$ and $\sqrt{\cdot}$ in the matrix \mathcal{A}_u denotes an operator square root.

State-space realizations of spatio-temporal oscillating fields have been discussed in [26]. The implementation of divergence-free covariance functions for magnetic field modeling have been recently presented in [27]. It is worth noting that fixing the GP to be realized as its causal version via the spectral factorization can also cause problems when the GP is combined with a suitable non-causal PDE. This is discussed in Section III-F.

As in the case of temporal processes discussed in Section III-A, it is possible to convert some classes of non-stationary covariance functions in state-space form. In that case the Fourier domain approach is not applicable, but similar methods as in the temporal case can also be used for spatio-temporal models.

C. ODE LFMs in joint state-space form

We can now combine the state-space ODE (21) with the state-space representation of LFMs in the previous section to obtain an augmented state-space representation of the LFM [5], [6]. Thus we obtain a combined model

$$\begin{aligned} \frac{d\mathbf{f}(t)}{dt} &= \mathbf{A}_f \mathbf{f}(t) + \mathbf{B}_f \mathbf{C}_u \mathbf{u}(t), \\ \frac{d\mathbf{u}(t)}{dt} &= \mathbf{A}_u \mathbf{u}(t) + \mathbf{B}_u w(t), \\ f(t) &= \mathbf{C}_f \mathbf{f}(t). \end{aligned} \quad (38)$$

If we now define

$$\begin{aligned} \mathbf{g} &= \begin{pmatrix} \mathbf{f} \\ \mathbf{u} \end{pmatrix}, \quad \mathbf{A} = \begin{pmatrix} \mathbf{A}_f & \mathbf{B}_f \mathbf{C}_u \\ 0 & \mathbf{A}_u \end{pmatrix}, \\ \mathbf{B} &= \begin{pmatrix} 0 & \mathbf{B}_u \end{pmatrix}, \quad \mathbf{C} = \begin{pmatrix} \mathbf{C}_f & 0 \end{pmatrix}, \end{aligned} \quad (39)$$

then the model can be written as a white-noise driven state-space model

$$\begin{aligned}\frac{d\mathbf{g}(t)}{dt} &= \mathbf{A} \mathbf{g}(t) + \mathbf{B} w(t), \\ f(t) &= \mathbf{C} \mathbf{g}(t).\end{aligned}\tag{40}$$

The measurements can be included into the model by replacing the second equation above by

$$y_k = \mathbf{C} \mathbf{g}(t_k) + \epsilon_k.\tag{41}$$

The advantage of this formulation is that we can infer both the process and the latent Gaussian process via Kalman filtering and smoothing in linear time (see [15], [16]). The link to Kalman filtering and smoothing follows from the result that as the model above is linear, we can form an exact discretization of the model as

$$\begin{aligned}\mathbf{g}_k &= \mathbf{U}(\Delta t_k) \mathbf{g}_{k-1} + \mathbf{q}_k, \\ y_k &= \mathbf{C} \mathbf{g}_k + \epsilon_k,\end{aligned}\tag{42}$$

where $\mathbf{g}_k \triangleq \mathbf{g}(t_k)$, $\mathbf{q}_k \sim \mathcal{N}(0, \mathbf{Q}(\Delta t_k))$, $\Delta t_k = t_k - t_{k-1}$, and

$$\begin{aligned}\mathbf{U}(\Delta t_k) &= \exp(\Delta t_k \mathbf{A}), \\ \mathbf{Q}(\Delta t_k) &= \int_0^{\Delta t_k} \exp((\Delta t_k - \tau) \mathbf{A}) \mathbf{B} q_c \mathbf{B}^\top \\ &\quad \times \exp((\Delta t_k - \tau) \mathbf{A})^\top d\tau,\end{aligned}\tag{43}$$

where q_c is the spectral density of the white noise $w(t)$. For numerical computation of the matrix exponential and the above integral there exists a wide range of numerical methods (see, e.g., [28], [29]) with ready-available implementations.

The link to Kalman filtering results from the fact that the model (42) is a standard discrete-time linear Gaussian model and the Kalman filter and (Rauch–Tung–Striebel, RTS) smoother provide the equations for exact Bayesian inference on it (see, e.g., [18]).

Another benefit of the representation (38) is that we can compute the covariance functions corresponding to the ODE LFM without resorting to numerical integration as in (16) and (19), or (22) and (24). The joint stationary covariance function of $\mathbf{g}(t) = (\mathbf{f}(t), \mathbf{u}(t))$ can be expressed as

$$\mathbf{K}_{\mathbf{g}}(t, t') = \begin{cases} \mathbf{P}_\infty \exp((t - t') \mathbf{A})^\top, & \text{if } t \geq t', \\ \exp((t' - t) \mathbf{A}) \mathbf{P}_\infty, & \text{if } t < t', \end{cases}\tag{44}$$

where \mathbf{P}_∞ is the solution to the Lyapunov equation

$$\mathbf{A} \mathbf{P}_\infty + \mathbf{P}_\infty \mathbf{A}^\top + \mathbf{B} q_c \mathbf{B}^\top = 0. \quad (45)$$

We can then pick up the covariance function of f and the cross-covariance of f and u as

$$\begin{aligned} k_{f,f}(t, t') &= \mathbf{C} \mathbf{K}_g(t, t') \mathbf{C}^\top, \\ k_{f,u}(t, t') &= \mathbf{C} \mathbf{K}_g(t, t') \mathbf{D}^\top, \end{aligned} \quad (46)$$

where we have defined $\mathbf{D} = \begin{pmatrix} 0 & \mathbf{C}_u \end{pmatrix}$.

D. PDE LFMs in joint state-space form

Partial differential equation models driven by Gaussian processes can also be often represented in a similar state-space form as the temporal models considered in the previous section. Consider for example the heat equation

$$\frac{\partial f(\mathbf{x}, t)}{\partial t} = D \nabla^2 f(\mathbf{x}, t) - \lambda f(\mathbf{x}, t) + u(\mathbf{x}, t). \quad (47)$$

Let us now assume that we have a state-space representation of the unknown function $u(\mathbf{x}, t)$ in form (35). This thus leads to the system

$$\begin{aligned} \frac{\partial f(\mathbf{x}, t)}{\partial t} &= \underbrace{(D \nabla^2 - \lambda)}_{\mathcal{A}_f} f(\mathbf{x}, t) + \mathbf{C}_u \mathbf{u}(\mathbf{x}, t), \\ \frac{\partial \mathbf{u}(\mathbf{x}, t)}{\partial t} &= \mathcal{A}_u \mathbf{u}(\mathbf{x}, t) + \mathbf{B}_u w(\mathbf{x}, t), \end{aligned} \quad (48)$$

where the analogy to the model (38) is apparent. For higher order derivatives on the left hand side we should define a vector of derivatives $(f, \partial f / \partial t, \dots)$, but for the present first order case this is unnecessary.

In order to obtain a single augmented model, analogously to the previous section we can define (with $\mathbf{B}_f = 1$ in this particular case)

$$\begin{aligned} \mathbf{g} &= \begin{pmatrix} f \\ \mathbf{u} \end{pmatrix}, \quad \mathcal{A} = \begin{pmatrix} \mathcal{A}_f & \mathbf{B}_f \mathbf{C}_u \\ 0 & \mathcal{A}_u \end{pmatrix}, \\ \mathbf{B} &= \begin{pmatrix} 0 & \mathbf{B}_u \end{pmatrix}, \quad \mathbf{C} = \begin{pmatrix} 1 & 0 \end{pmatrix}, \end{aligned} \quad (49)$$

which leads to a model of the form

$$\begin{aligned} \frac{\partial \mathbf{g}(\mathbf{x}, t)}{\partial t} &= \mathcal{A} \mathbf{g}(\mathbf{x}, t) + \mathbf{B} w(\mathbf{x}, t), \\ y_k &= \mathbf{C} \mathbf{g}(t_k) + \epsilon_k. \end{aligned} \quad (50)$$

The discretization now proceeds analogously to the previous section, except that the matrix exponential is replaced with an operator exponential (exponential semigroup) and its time-integral:

$$\begin{aligned}\mathcal{U}(\Delta t_k) &= \exp(\Delta t_k \mathcal{A}), \\ \mathbf{Q}(\mathbf{x}, \mathbf{x}'; \Delta t_k) &= \int_0^{\Delta t_k} \exp((\Delta t_k - \tau) \mathcal{A}) \mathbf{B} q_c(\mathbf{x}, \mathbf{x}') \mathbf{B}^\top \\ &\quad \times \exp((\Delta t_k - \tau) \mathcal{A})^\top d\tau.\end{aligned}\tag{51}$$

Furthermore, both the spectral density $q_c(\mathbf{x}, \mathbf{x}')$ and the covariance $\mathbf{Q}(\mathbf{x}, \mathbf{x}'; \Delta t_k)$ become kernels in \mathbf{x}, \mathbf{x}' . Note that above, we need to interpret the operator application from the right as operating onto the variable \mathbf{x}' (cf. [25]).

The inference in the above model can be done with infinite-dimensional Kalman filters and RTS smoothers (see, [25]). However, in practice we often use some kind of basis function expansion methods to project the equations onto a finite-dimensional sub-space which reduces the filters and smoothers to their finite-dimensional versions.

The covariance functions for explicit GP inference could also be computed with the operator analogs of Equations (44), (45), and (46).

E. Parameter estimation in state-space LFMs

In the conversion of the LFM into a state-space form, the parameters of the differential equations and covariance functions transform into parameters of the state-space model. In the resulting model we can estimate the parameters using parameter estimation methods for state-space models. See, for example, [18] for a review of parameter estimation methods for state-space models.

F. Counter-example: the Poisson equation

The state-space representation has one hard limitation in the physical model: it needs to be causal in the time-direction. Although statistics of the LFM GP do not depend on the causality of the process and thus we can always replace the LFM GP with its causal version, this is not true for a physical process as we demonstrate in this section.

Sometimes we are interested in partial differential equations without time-dependence, with a prototypical example being the Poisson equation

$$\nabla^2 f(\mathbf{x}) = u(\mathbf{x}).\tag{52}$$

Although there is no time-variable, we can redefine one of the spatial variables as time $\mathbf{x} \triangleq (\mathbf{x}, t)$ which thus leads to the equation

$$\frac{\partial^2 f(\mathbf{x}, t)}{\partial t^2} + \nabla^2 f(\mathbf{x}, t) = u(\mathbf{x}, t). \quad (53)$$

However, this kind of equations without builtin physical time-dependence need to more care than physically time dependent equations as will be illustrated in the following.

It would now be tempting to move the term $\nabla^2 f(\mathbf{x}, t)$ to the right hand side and form a state-space form via defining the state as $\mathbf{f} = (f, \partial f / \partial t)$, which formally gives:

$$\frac{\partial \mathbf{f}(\mathbf{x}, t)}{\partial t} = \begin{pmatrix} 0 & 1 \\ -\nabla^2 & 0 \end{pmatrix} \mathbf{f}(\mathbf{x}, t) + \begin{pmatrix} 0 \\ 1 \end{pmatrix} u(\mathbf{x}, t). \quad (54)$$

However, this is not a valid state-space model, because it is not causal. This can be easiest seen by taking the spatial Fourier transform of (54) which gives

$$\frac{\partial \tilde{\mathbf{f}}(i\boldsymbol{\omega}_x, t)}{\partial t} = \begin{pmatrix} 0 & 1 \\ \|\boldsymbol{\omega}_x\|^2 & 0 \end{pmatrix} \tilde{\mathbf{f}}(i\boldsymbol{\omega}_x, t) + \begin{pmatrix} 0 \\ 1 \end{pmatrix} \tilde{u}(i\boldsymbol{\omega}_x, t). \quad (55)$$

which can be seen to correspond to a unstable system, because the transition matrix has the positive eigenvalue $\|\boldsymbol{\omega}_x\|$, which implies a diverging solution.

In order to form a valid state-space model, let us take time-space Fourier transform of (53) which gives

$$(i\omega_t)^2 F(i\boldsymbol{\omega}_x, i\omega_t) + \|\boldsymbol{\omega}_x\|^2 F(i\boldsymbol{\omega}_x, i\omega_t) = U(i\boldsymbol{\omega}_x, i\omega_t), \quad (56)$$

implying that the spectral density of f has the form

$$S_f(\boldsymbol{\omega}_x, \omega_t) = \frac{S_u(\boldsymbol{\omega}_x, \omega_t)}{((\omega_t)^2 + \|\boldsymbol{\omega}_x\|^2)^2} \quad (57)$$

which can be refactored with respect to ω_t to give the stable, statistically equivalent system

$$\frac{\partial^2 f(\mathbf{x}, t)}{\partial t^2} + 2\sqrt{-\nabla^2} \frac{\partial f(\mathbf{x}, t)}{\partial t} - \nabla^2 f(\mathbf{x}, t) = u(\mathbf{x}, t) \quad (58)$$

which has the state-space representation

$$\begin{aligned} \frac{\partial \mathbf{f}(\mathbf{x}, t)}{\partial t} &= \underbrace{\begin{pmatrix} 0 & 1 \\ \nabla^2 & -2\sqrt{-\nabla^2} \end{pmatrix}}_{\mathbf{A}_f} \mathbf{f}(\mathbf{x}, t) + \underbrace{\begin{pmatrix} 0 \\ 1 \end{pmatrix}}_{\mathbf{B}_f} u(\mathbf{x}, t), \\ f(\mathbf{x}, t) &= \underbrace{\begin{pmatrix} 1 & 0 \end{pmatrix}}_{\mathbf{C}_f} \mathbf{f}(\mathbf{x}, t). \end{aligned} \quad (59)$$

This model is now time-stable and can be augmented with a state-space model for u in order to yield a state-space representation for the full LFM model.

However, it turns out that although we can indeed estimate the process f easily using the representation, the estimation of u becomes non-trivial. This is because we have implicitly made a unitary transformation for the input process. Although this does not affect the covariance structure of the input, it does transform the posterior mean of it using the same transformation. In order to determine the original u from the estimate of u obtained from this equation, we need to solve another pseudo-differential equation, which can be hard. Thus the benefit of using state-space methods for the Poisson equation is quite limited.

G. Detectability and observability of latent force models

In this section, our aim is to discuss the detectability and observability of the latent force models. For the theorems, we need to assume that the joint force system has a state-space representation, because otherwise the classical observability and detectability results [30]–[32] are not applicable. We only consider finite-dimensional models which is enough for our purposes because infinite-dimensional (PDE) models anyway need to be discretized and in order to ensure the detectability and observability of the resulting models, the finite-dimensional results are sufficient. However, the reader is referred to the book of Curtain [33] for the corresponding infinite-dimensional results.

That is, we assume that we have a latent force model which has the following state space representation

$$\begin{aligned}\frac{d\mathbf{g}(t)}{dt} &= \mathbf{A} \mathbf{g}(t) + \mathbf{B} w(t), \\ y_k &= \mathbf{C} \mathbf{g}(t_k) + \epsilon_k,\end{aligned}\tag{60}$$

where the matrices above are defined by (cf. (39))

$$\mathbf{A} = \begin{pmatrix} \mathbf{A}_f & \mathbf{B}_f \mathbf{C}_u \\ 0 & \mathbf{A}_u \end{pmatrix}, \mathbf{B} = \begin{pmatrix} 0 & \mathbf{B}_u \end{pmatrix}, \mathbf{C} = \begin{pmatrix} \mathbf{C}_f & 0 \end{pmatrix},\tag{61}$$

with $\mathbf{g} = (\mathbf{f}, \mathbf{u})$. In this representation we have dropped the control signal, because it does not affect the detectability and observability.

It is also reasonable to assume that the state-space representation of the latent force model is stable and hence detectable. However, the physical system part itself usually is not stable. We

need to assume though that it is at least detectable and preferably it should be observable. The most useful case occurs when the whole joint system is observable. The sampling procedure also affects the observability and we need to ensure that we do not get 'aliasing' kind of phenomenon analogously to sampling a signal with a sampling frequency that is below the Nyquist frequency. Let us start with the following result for detectability.

Lemma III.1. *Assume that we have a latent force model which has the state space representation given in (60). Assume that $(\exp(\mathbf{A}_f \Delta t_k), \mathbf{C}_f)$ is detectable, and that the latent force $u(t)$ has a exponentially stable state space representation. Then the full system is detectable and the Kalman filter for the model is exponentially stable.*

Proof. We first discretize the system at arbitrary time points. The discretized system is (cf. (42))

$$\begin{aligned} \mathbf{g}_k &= \mathbf{U}(\Delta t_k) \mathbf{g}_{k-1} + \mathbf{q}_k, \\ y_k &= \mathbf{C} \mathbf{g}_k + \epsilon_k, \end{aligned} \tag{62}$$

which will be detectable provided that there exists a bounded gain sequence \mathbf{K}_k such that $\tilde{\mathbf{g}}_k = (\mathbf{U}(\Delta t_k) - \mathbf{K}_k \mathbf{C}) \tilde{\mathbf{g}}_{k-1}$ is exponentially stable [32]. More explicitly, the following system needs to be exponentially stable with some choice of sequence \mathbf{K}_k :

$$\begin{aligned} \tilde{\mathbf{f}}_k &= \exp(\mathbf{A}_f \Delta t_k) \tilde{\mathbf{f}}_{k-1} + \Gamma_k \tilde{\mathbf{u}}_{k-1} - \mathbf{K}_k \mathbf{C}_f \tilde{\mathbf{f}}_{k-1}, \\ \tilde{\mathbf{u}}_k &= \exp(\mathbf{A}_u \Delta t_k) \tilde{\mathbf{u}}_{k-1}. \end{aligned} \tag{63}$$

As the process \mathbf{u}_k is exponentially stable, the sequence $\tilde{\mathbf{u}}_k$ is exponentially decreasing and bounded. Hence it does not affect the stability of the first equation. Therefore, the full system will be detectable provided that there exists a gain sequence \mathbf{K}_k such that $\tilde{\mathbf{f}}_k = (\exp(\mathbf{A}_f \Delta t_k) - \mathbf{K}_k \mathbf{C}_f) \tilde{\mathbf{f}}_{k-1}$ is exponentially stable. The gain sequence exists, because $[\exp(\mathbf{A}_f \Delta t_k), \mathbf{C}_f]$ is detectable by assumption. \square

Above, in Lemma III.1 we had to assume the detectability of the discretized system. There are many ways to assure that, but one way is to demand that the continuous physical model is observable and that we are not sampling critically [34], that is, in a way that would lead to aliasing of frequencies as in the Shannon-Nyquist theory. Although observability is a quite strong condition compared to detectability, it assures that we have the chance to reconstruct the physical system with an arbitrary precision by improving the measurement protocol, which would not be true for mere detectability. Actually, sole detectability of the system is quite useless in practice, because our estimates might still have error offsets that would be impossible to observe.

If we assume that the physical system part is observable and the sampling is not critical, we get the following detectability theorem. Note that we do not yet assume that the latent force model part would be observable although its stability already implies that it is detectable.

Theorem III.1. *Assume that $(\mathbf{A}_f, \mathbf{C}_f)$ is observable, the physical system is not critically sampled, and that the latent force model part is stable. Then the full system is detectable and the Kalman filter for the model is exponentially stable.*

Proof. According to [34], the observability of the continuous time system together with the non-critical sampling ensures that the discrete-time system is also observable. As discrete-time observability implies discrete-time detectability the result follows from Lemma III.1. \square

Let us now consider the conditions for the observability of the full system. It turns out that in general, the best way to determine the observability of the joint system is not to attempt to think of the physical system and the latent force model separately, but explicitly consider the joint state-space model. There are numerous attempts to map the properties of this kind cascaded systems to the properties of the joint system (e.g. [35]–[37]), but still the best way to go seems to be simply to use a standard observability tests on the joint system. The properties of the sub-systems of this kind of cascade do not alone determine the observability, because we can have phenomena like zero-pole cancellation which leads to a non-observable system even when all the subsystems are observable (see, e.g., [35]). When we also account for the effect of sampling to observability, we get the following theorem.

Theorem III.2. *Assume that the continuous-time joint system (\mathbf{A}, \mathbf{C}) is observable, and the observations are not critically sampled, then the discrete-time full system is observable.*

Proof. See [34]. \square

In practical terms it is thus easiest to use, for example, the classical rank-condition (see, e.g., [38]) which says that the (joint) system (\mathbf{A}, \mathbf{C}) is observable, which in time-invariant case is ensured provided that the following matrix has full rank:

$$\mathcal{O} = \begin{pmatrix} \mathbf{C} \\ \mathbf{C} \mathbf{A} \\ \vdots \\ \mathbf{C} \mathbf{A}^{m+s-1} \end{pmatrix}, \quad (64)$$

and then ensure that sampling is non-critical [34].

IV. STOCHASTIC CONTROL OF LATENT FORCE MODELS

In this section, we discuss the stochastic control problems related to latent force models. In particular, we provide and analyze the solutions for the linear quadratic regulation (LQR) problem for them.

A. Controlled basic LFM

Let us consider a controlled version of the state-space model with a Gaussian process input (21):

$$\frac{d\mathbf{f}(t)}{dt} = \mathbf{A}_f \mathbf{f}(t) + \mathbf{B}_f u(t) + \mathbf{M}_f \mathbf{c}(t). \quad (65)$$

Also recall that any basic LFM of the form (14) with a control signal can be written in a state-space form as above.

We will specifically aim to consider optimal control problems which minimize the quadratic cost functional

$$\begin{aligned} \mathcal{J}[\mathbf{c}] = & \frac{1}{2} \mathbb{E} \left[\mathbf{f}^\top(T) \mathbf{\Phi} \mathbf{f}(T) \right. \\ & \left. + \int_0^T (\mathbf{f}^\top(t) \mathbf{X}(t) \mathbf{f}(t) + \mathbf{c}^\top(t) \mathbf{U}(t) \mathbf{c}(t)) dt \right], \end{aligned} \quad (66)$$

because they lead to computationally tractable control laws. However, the principle outlined here can also be extended to more general cost functionals although the numerical methods become order of magnitude more complicated.

A straightforward approach to optimal control with the quadratic cost (66) is to use the separation principle of linear estimation and control which amounts to designing the optimal controller for the case $u(t) = 0$ and use it in cascade with a Kalman filter. This indeed is the optimal solution in the case of white $u(t)$, but not in our case.

The correct approach in this case, which also utilizes the learning outcome of the Gaussian process regression is to use the augmented state space model with the control signal. In this case it is given as (see (40))

$$\frac{d\mathbf{g}(t)}{dt} = \mathbf{A} \mathbf{g}(t) + \mathbf{B} w(t) + \mathbf{M} \mathbf{c}(t), \quad (67)$$

with the measurement model (41), where $\mathbf{M} = \begin{pmatrix} \mathbf{M}_f \\ 0 \end{pmatrix}$ and the matrices \mathbf{A} and \mathbf{B} are defined in (39). What we now should do is to design a controller for the above model with $w(t) = 0$ and run it in cascade with a Kalman filter processing the measurements in the model (41). This yields to a controller which jointly learns the functions f and u and optimizes the control with respect to the cost criterion. In this case the control cost function can be rewritten in form

$$\begin{aligned} \mathcal{J}[\mathbf{c}] = & \frac{1}{2} \mathbb{E} \left[\mathbf{g}^\top(T) \boldsymbol{\Phi}_g \mathbf{g}(T) \right. \\ & \left. + \int_0^T (\mathbf{g}^\top(t) \mathbf{X}_g(t) \mathbf{g}(t) + \mathbf{c}^\top(t) \mathbf{U}(t) \mathbf{c}(t)) dt \right], \end{aligned} \quad (68)$$

where

$$\boldsymbol{\Phi}_g = \begin{pmatrix} \boldsymbol{\Phi} & 0 \\ 0 & 0 \end{pmatrix}, \quad \mathbf{X}_g(t) = \begin{pmatrix} \mathbf{X}(t) & 0 \\ 0 & 0 \end{pmatrix}. \quad (69)$$

The design of the optimal linear quadratic controller for the resulting model can be done by using the classical Riccati-equation-based approaches [30], [39]. Namely, the optimal control takes the form

$$\mathbf{c}(t) = -\mathbf{U}^{-1}(t) \mathbf{M}^\top \mathbf{K}(t) \hat{\mathbf{g}}(t), \quad (70)$$

where $\hat{\mathbf{g}}(t)$ is the Kalman filter estimate of $\mathbf{g}(t)$ and the matrix $\mathbf{K}(t)$ solves the backward Riccati differential equation

$$\begin{aligned} \frac{d\mathbf{K}(t)}{dt} = & -\mathbf{A}^\top \mathbf{K}(t) - \mathbf{K}(t) \mathbf{A} \\ & + \mathbf{K}(t) \mathbf{M} \mathbf{U}^{-1}(t) \mathbf{M}^\top \mathbf{K}(t) - \mathbf{X}_g(t) \end{aligned} \quad (71)$$

with the boundary condition $\mathbf{K}(T) = \boldsymbol{\Phi}_g$. However, it turns out that we write this solution for the LFM model in more explicit form which reveals its structure better. That is summarized in the following theorem which can be obtained by partitioning $\mathbf{K} = \begin{pmatrix} \mathbf{K}_{11} & \mathbf{K}_{12} \\ \mathbf{K}_{12}^\top & \mathbf{K}_{22} \end{pmatrix}$.

Theorem IV.1. *The control law in (70) can be written as*

$$\mathbf{c}(t) = - \left(\mathbf{U}^{-1} \mathbf{M}_f^\top \mathbf{K}_f(t) \quad \mathbf{U}^{-1} \mathbf{M}_f^\top \mathbf{K}_{12}(t) \right) \hat{\mathbf{g}}(t), \quad (72)$$

where $\mathbf{K}_f(t) \triangleq \mathbf{K}_{11}(t)$ is the Riccati equation solution for the non-forced physical model. The full set of equations is

$$\begin{aligned}
\frac{d\mathbf{K}_{11}(t)}{dt} &= -\mathbf{A}_f^\top \mathbf{K}_{11} - \mathbf{K}_{11} \mathbf{A}_f \\
&\quad + \mathbf{K}_{11} \mathbf{M}_f \mathbf{U}^{-1} \mathbf{M}_f^\top \mathbf{K}_{11} - \mathbf{X}(t), \\
\frac{d\mathbf{K}_{12}(t)}{dt} &= -\mathbf{A}_f^\top \mathbf{K}_{12} - \mathbf{K}_{11} \mathbf{B}_f \mathbf{C}_u - \mathbf{K}_{12} \mathbf{A}_u \\
&\quad + \mathbf{K}_{11} \mathbf{M}_f \mathbf{U}^{-1} \mathbf{M}_f^\top \mathbf{K}_{12}, \\
\frac{d\mathbf{K}_{22}(t)}{dt} &= -\mathbf{C}_u^\top \mathbf{B}_f^\top \mathbf{K}_{12} - \mathbf{A}_u^\top \mathbf{K}_{22} - \mathbf{K}_{21} \mathbf{B}_f \mathbf{C}_u \\
&\quad - \mathbf{K}_{22} \mathbf{A}_u + \mathbf{K}_{12}^\top \mathbf{M}_f \mathbf{U}^{-1} \mathbf{M}_f^\top \mathbf{K}_{12}.
\end{aligned} \tag{73}$$

In the above theorem the gain for the physical system (i.e. f) portion of the state is exactly the same as in the optimal controller without a latent force. However, the second part of gain is non-zero and uses the latent force model state for control feedback as well.

It would also be possible to consider more complicated cost functions of the form

$$\mathcal{J}[\mathbf{c}] = \mathbb{E} \left[\psi(\mathbf{f}(T)) + \int_0^T L(\mathbf{f}(t), \mathbf{c}(t), t) dt \right], \tag{74}$$

which would then lead to interesting control laws that utilize the latent force in a non-linear feedback control. However, the use of more general cost functions than the quadratic cost is considerably more complicated than using a quadratic cost, because the estimation and control no longer separate. A typical practical approach is to assume that this still approximately holds. Also in this case we argue that it is desirable to design the (deterministic) controller for the augmented system instead of the original system with $u(t) = 0$, because this design is able to fully use the learning outcome of the Gaussian process regression of the input.

In the next section we will simplify the control problem even more, and consider the limit $T \rightarrow \infty$, because it leads to a particularly convenient class of linear controllers which are computationally tractable while still being able to use the learning outcome of the Gaussian process inference.

B. Linear quadratic regulation of basic LFM

In the LFM case, namely because we have restricted our consideration to time-invariant models, a very convenient type of control problem is the infinite-time linear regulation problem

which corresponds to the cost function

$$\mathcal{J}[\mathbf{c}] = \int_0^\infty (\mathbf{f}^\top(t) \mathbf{X} \mathbf{f}(t) + \mathbf{c}^\top(t) \mathbf{U} \mathbf{c}(t)) dt. \quad (75)$$

By rewriting the model as an augmented state-space model as we did in the previous section and by following the classical results, the controller becomes

$$\mathbf{c}(t) = -\mathbf{U}^{-1} \mathbf{M}^\top \mathbf{K} \hat{\mathbf{g}}(t), \quad (76)$$

where the matrix \mathbf{K} is the solution to the algebraic Riccati equation (ARE)

$$0 = -\mathbf{A}^\top \mathbf{K} - \mathbf{K} \mathbf{A} + \mathbf{K} \mathbf{M} \mathbf{U}^{-1} \mathbf{M}^\top \mathbf{K} - \mathbf{X}_g, \quad (77)$$

where $\mathbf{X}_g = \begin{pmatrix} \mathbf{X} & 0 \\ 0 & 0 \end{pmatrix}$.

By solving the control law from these equations, we get a controller which is function of both the estimate of the function \mathbf{f} as well the latent force \mathbf{u} , and is hence able to utilize both the estimate of the function as well the learned Gaussian latent force model.

It is also possible to express the solution to the LFM control problem above in terms of the corresponding control solution to the non-forced problem similarly to the time-varying case considered in the previous section. This result is summarized in the following theorem which can be derived by setting the time derivatives in Theorem IV.1 to zero.

Theorem IV.2. *The control law in (76) can now be written as*

$$\mathbf{c}(t) = - \left(\mathbf{U}^{-1} \mathbf{M}_f^\top \mathbf{K}_f \quad \mathbf{U}^{-1} \mathbf{M}_f^\top \mathbf{K}_{12} \right) \hat{\mathbf{g}}(t), \quad (78)$$

where $\mathbf{U}^{-1} \mathbf{M}_f^\top \mathbf{K}_f$ is just the non-forced-case gain and \mathbf{K}_{12} can be solved from the Sylvester equation

$$(\mathbf{K}_f \mathbf{M}_f \mathbf{U}^{-1} \mathbf{M}_f^\top - \mathbf{A}_f^\top) \mathbf{K}_{12} - \mathbf{K}_{12} \mathbf{A}_u = \mathbf{K}_f \mathbf{B}_f \mathbf{C}_u. \quad (79)$$

Note that although the system is stabilizable also by setting the second term to zero, that is, using the non-forced gain (cf. Theorem IV.5), a better solution is obtained by using the control in Theorem IV.2 which depends on the latent force as well.

C. Stabilizability and non-controllability of controlled LFM

The aim is now to discuss the controllability and stabilizability of state-space latent force models. We assume that the model has the form

$$\frac{d\mathbf{g}(t)}{dt} = \mathbf{A} \mathbf{g}(t) + \mathbf{B} w(t) + \mathbf{M} \mathbf{c}(t), \quad (80)$$

where (cf. (39))

$$\mathbf{A} = \begin{pmatrix} \mathbf{A}_f & \mathbf{B}_f \mathbf{C}_u \\ 0 & \mathbf{A}_u \end{pmatrix}, \mathbf{B} = \begin{pmatrix} 0 & \mathbf{B}_u \end{pmatrix}, \mathbf{M} = \begin{pmatrix} \mathbf{M}_f \\ 0 \end{pmatrix}, \quad (81)$$

and $\mathbf{g} = (\mathbf{f}, \mathbf{u})$. We will not explicitly consider infinite-dimensional (PDE) models, because such models need to be discretized anyway, and it is best to analyze the stabilizability and the controllability in the resulting discretized representation. For infinite-dimensional results the reader is referred to the book of Curtain [33].

First of all, the stabilizability of the system is guaranteed solely by ensuring that the physical model part is stabilizable, because state-space representations of stationary GPs—provided that they are properly formed—are stable. Thus we have the following theorem.

Theorem IV.3. *Assume that $(\mathbf{A}_f, \mathbf{M}_f)$ is stabilizable and the latent force has an exponentially stable state space representation. Then the full system is stabilizable.*

Proof. The system is stabilizable if there exist a finite gain \mathbf{K}_c such that the system $d\tilde{\mathbf{g}}/dt = (\mathbf{A} + \mathbf{M} \mathbf{K}_c) \tilde{\mathbf{g}}$ is exponentially stable [40]. More explicitly we should have

$$\begin{aligned} \frac{d\tilde{\mathbf{f}}}{dt} &= (\mathbf{A}_f + \mathbf{M}_f \mathbf{K}_f) \tilde{\mathbf{f}} + (\mathbf{B}_f \mathbf{C}_u + \mathbf{M}_f \mathbf{K}_u) \tilde{\mathbf{u}}, \\ \frac{d\tilde{\mathbf{u}}}{dt} &= \mathbf{A}_u \tilde{\mathbf{u}}, \end{aligned} \quad (82)$$

where we have written $\mathbf{K}_c = \begin{pmatrix} \mathbf{K}_f & \mathbf{K}_u \end{pmatrix}$. Because $\tilde{\mathbf{u}}$ is exponentially decreasing and bounded, we can safely set $\mathbf{K}_u = 0$. The remainder of the system will be stabilizable if there exists a gain \mathbf{K}_f such that $d\tilde{\mathbf{f}}/dt = (\mathbf{A}_f + \mathbf{M}_f \mathbf{K}_f) \tilde{\mathbf{f}}$ is exponentially stable. By our assumption on the stabilizability of $(\mathbf{A}_f, \mathbf{M}_f)$, this is true and hence the result follows. \square

The stabilizability also implies that the corresponding LQ controller is uniquely determined [39].

However, the sole stabilizability is not very useful in practice, because sole stabilizability says that we might have randomly wandering subprocesses in the joint system which practically

prevent us from controlling the process exactly where we wish it to go. A much stronger requirement is to require that the full system be controllable. Unfortunately, it turns out that latent force models are never fully controllable in the present formulation, because we cannot control the subsystem corresponding to the GP force.

Theorem IV.4. *Latent force models are not controllable.*

Proof. The model is in Kalman's canonical form [31], where the non-controllable part is the latent force. \square

In practice, the non-controllability of the force part is not a problem, as we are actually interested in controlling the physical system part of the model, not the force per se. It turns out that the physical system can be controllable even though the full system is not controllable. This result can be obtained as a corollary of so called output controllability (see, e.g., [38]) as follows.

Theorem IV.5. *Assume that $(\mathbf{A}_f, \mathbf{M}_f)$ is controllable. Then the full system is output controllable with respect to the physical system part.*

Proof. This can be derived by writing down the output controllability condition [38] and noticing that it reduces to controllability of the physical system part. \square

The above result is useful when the system is fully observable as well. Then it ensures that we can successfully control the physical system part although the latent force model remains uncontrollable. However, if the latent force model is not fully observable, then the latent force model inherently causes disturbance to the physical system and although we can keep the system stable, the state cannot be forced to follow a given trajectory.

As a conclusion, for all practical purposes a (time-invariant) latent force model is controllable, if it is observable and the following matrix has a full rank:

$$\mathcal{C} = \begin{pmatrix} \mathbf{M}_f & \mathbf{A}_f \mathbf{M}_f & \dots & \mathbf{A}_f^{m-1} \mathbf{M}_f \end{pmatrix}. \quad (83)$$

D. Controlled PDE based LFMs

In the case of PDE LFMs we get models of the form

$$\begin{aligned} \frac{\partial \mathbf{g}(\mathbf{x}, t)}{\partial t} &= \mathcal{A} \mathbf{g}(\mathbf{x}, t) + \mathbf{B} w(\mathbf{x}, t) + \mathbf{M}_f c(\mathbf{x}, t), \\ y_k &= \mathbf{C} \mathbf{g}(t_k) + \epsilon_k, \end{aligned} \quad (84)$$

where the control problem corresponds to designing the control function $c(\mathbf{x}, t)$ minimizing, for example, a linear quadratic cost functional. In principle, it is possible to directly analyze such infinite-dimensional control problems which leads to, for example, generalizations of the controllability concepts [33]. However, in practice, as the first step after setting up the model, we replace the infinite-dimensional model with a finite-dimensional approximation. Therefore it is actually more fruitful to directly analyze the finite-dimensional approximation rather than the original infinite-dimensional model—this way we can also easily account for the effect of discretization. For the analysis of the finite-dimensional approximate model the results in the previous sections apply as such.

V. EXTENSIONS AND DISCUSSION

A. Non-linear models

It is also possible to consider non-linear differential equation (physical) models which are driven by Gaussian process. In that case it is necessary to resort to approximations in order to compute the posterior distribution for u^* given \mathbf{y} , or the predictive distribution for f^* given \mathbf{y} . Approximation methods for the covariance function formulation of LFMs have been presented in [14], [41], [42]. In the state-representation, the non-linearity transform the linear state-space model into a non-linear state-space model, where we can use non-linear Kalman filtering and smoothing methods [6], [18]. The stochastic control problems could also be extended to the non-linear case (see, e.g., [43]).

B. Switching LFMs

In the basic LFM, the input process $u(t)$ is a continuous process. An important extension in different applied scenarios is to allow the input force $u(t)$ to be discontinuous. Equally important is to allow the coefficients $\{a_i\}_{i=0}^m$ that define the differential operator L , to take discrete values as function of t . The setup now includes a sequential series of basic LFMs $\{L_j f(t) = u_j(t)\}_{j=1}^J$, each of them applied in a particular time interval $\{[T_{j-1}, T_j]\}_{j=1}^J$, where J is the maximum number of time intervals considered, L_j is a differential operator with coefficients $\{a_i^j\}_{i=0}^{m,J}$, and $\{u_j(t)\}_{j=1}^J$ are GPs with covariance functions $k_j(t, t')$. Continuity in the output $f(t)$ is ensured by forcing the initial conditions for the j th ODE to take the same value than the output $f(t)$ and its derivatives, at the last input point for the $(j-1)$ th time interval. This kind of models have been considered in [5], [44].

C. Operators and non-linearities in measurement model

As shown in [12], [16], [25] Gaussian process regression models and hence LFM's easily extend to having linear operators or functionals \mathcal{H}_k in the measurement model:

$$y_k = \mathcal{H}_k f(\cdot, t) + \epsilon_k. \quad (85)$$

This kind of formulations are especially useful in inverse problems [45] arising, for example, in computer tomography and other imaging methods where the operator \mathcal{H}_k encodes the physics of the system into the model. This kind of formulation can also be used to enforce the field to have zero curl, which is useful, for example, when the unknown field corresponds to a magnetic field [27].

We can also consider non-linear transformations in the measurement model which allows us to consider, for example, classification problems. There exists efficient methods for approximate inference in such non-linear models [1], [18].

D. Numerical methods for PDEs

One commonly used method for approximate solution of partial differential equations (PDEs) is the Galerkin's method which approximates the solution on a finite-dimensional space spanned by sets of the basis functions $\{\phi_i(\mathbf{x}) \mid i = 1, 2, \dots, N\}$. With a suitable selection of basis functions it (in its weak formulation) leads to widely-used methods such as finite element method (FEM). We can then approximate \mathbf{g} via the series expansion

$$\mathbf{g}(\mathbf{x}, t) = \sum_{i=1}^N \mathbf{g}_i(t) \phi_i(\mathbf{x}), \quad (86)$$

where $\mathbf{g}_i(t)$ are jointly Gaussian random variables.

By substituting this into the LFM and approximating the Gaussian process prior using a similar series expansion, we can re-express the model in terms of the Gaussian state variables $\mathbf{g}_i(t)$ [25]. When we use the eigenfunctions of the Laplacian operator as the basis and when the covariance function is stationary, we can approximate the covariances via point-wise evaluations of the spectral density of u [46]. With this basis the case of translation invariant operators also becomes particularly easy, because the matrices become diagonal. Finite-differences can also be used to approximate the operators in the equations [47].

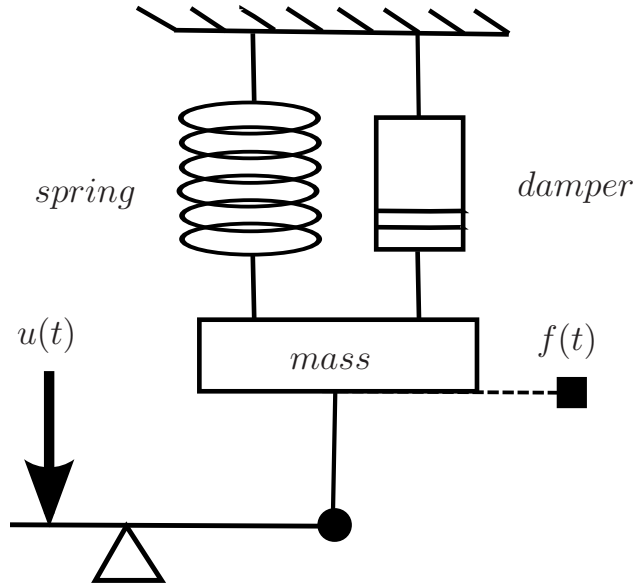


Fig. 2. A mass-spring-damper system to illustrate a latent force model with ordinary differential equations.

VI. EXPERIMENTAL RESULTS

In this section, we illustrate the latent force model framework in three different problems: a second order ordinary differential equation modeling a spring with and without control, a Poisson equation in two spatial dimensions, and an controlled heat source in two dimensions.

A. Inference on GP driven spring

Our first illustrative example corresponds to the second order differential equation model described in equation (1) with no control $c(t) = 0$. Figure 2 shows a mass-spring-damper system being forced by a latent function $u(t)$, which is an example of a physical system with such a model. The output $f(t)$ is also shown in the figure and represents the position of the body with a particular reference. For this model, we can compute the covariance function for $k_{f,f}(t, t')$ in equation (16) using the following Green's function

$$h(t - s) = \frac{1}{\omega} \exp(-\alpha(t - s)) \sin(\omega(t - s)), \quad (87)$$

where $\omega = \sqrt{4\gamma - \lambda^2}/2$, and $\alpha = \lambda/2$. If the covariance function $k(t - t')$ for the latent process $u(t)$ is a squared exponential (SE) covariance of the form

$$k(t - t') = \sigma_t^2 \exp \left[-\frac{(t - t')^2}{\ell_t^2} \right],$$

it is possible to obtain closed-form solutions for $k_{f,f'}(t, t')$ and $k_{f,u}$ (see Eq. (19)) [2].

We assume that the input source $u(t)$ is given as

$$u(t) = \sin(0.73t - 0.64) + \sin(0.29t - 0.64),$$

where $t \in [0, 50]$. The dashed line in Figure 4 shows the input $u(t)$ computed for $N = 5000$ points equally distributed in the input domain.

Parameters for the ODE are $\lambda = 0.1$, and $\gamma = 1$. The output function $f(t)$ obtained by solving the differential equation is shown in Figure 3 as a dashed line. We uniformly choose $n = 500$ time points $\{t_k\}_{k=1}^n$ within the interval $t \in [0, 50]$, together with their corresponding function evaluations $\{f_k\}_{k=1}^n$, as part of the training data. For the training points, we add Gaussian noise ϵ_k with zero mean and variance equal to 0.09, to get measurements $\{y_k\}_{k=1}^n$. We assume that the parameters λ , γ , and the variance for ϵ_k are known, and only estimate the parameters σ_t^2 , and ℓ_t^2 of the kernel function $k(t - t')$. The estimates were obtained by maximizing the log marginal likelihood by using the conjugate gradient method [1].

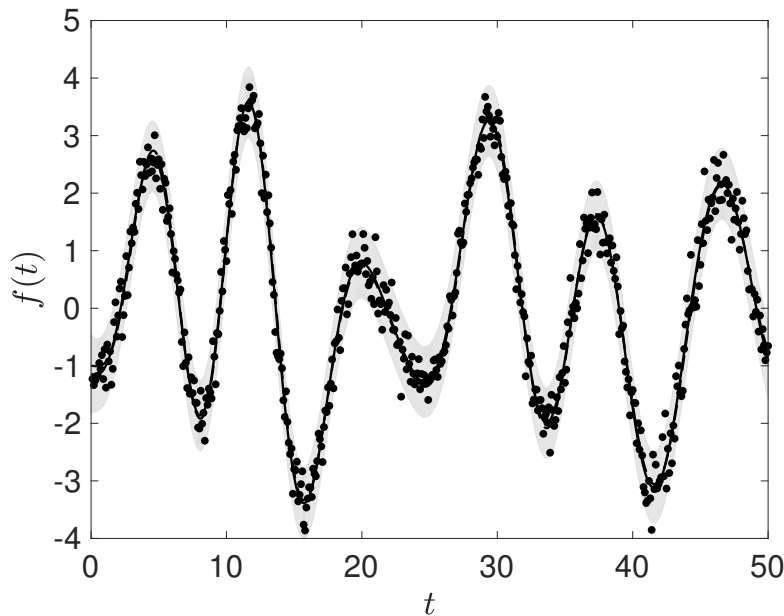


Fig. 3. ODE output $f(t)$ and measurements $\{y_k\}_{k=1}^n$. The dashed line is the ground truth output $f(t)$ obtained by solving the ODE, and the dots represent the noisy measurements $\{y_k\}_{k=1}^n$. The solid line is the predictive mean $f(t_*)$, and the grey region represents the predictive uncertainty, two standard deviations away from the mean prediction.

Results of inference for the latent function $u(t)$ are shown in Figure 4. The mean squared error (MSE) between the ground truth source $u(t)$, and the inferred mean source $u^*(t)$ is equal

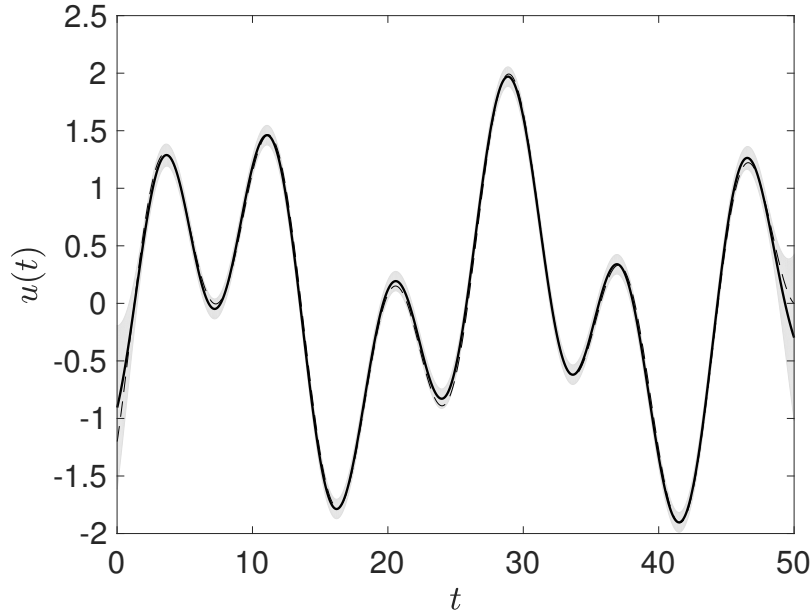


Fig. 4. Ground truth and predicted source $u(t)$. The dashed line is the ground truth source $u(t)$. The solid line is the mean posterior for u^* , while the shaded region covers two standard deviation away from the mean.

to 0.0036. The MSE between the ground truth output $f(t)$ and the mean predicted output $f^*(t)$ evaluated at the test input points is 0.0040. As can be seen in the figure, the estimate is very close to the truth and the error quantiles are realistic.

The results above were obtained by explicitly computing the modified covariance function corresponding to the LFM. However, the result of using the equivalent state-space representation is practically the same, except for the minor differences due to the Padé approximation required for the SE covariance function. Hence we do not show the state-space result here (it would be visually indistinguishable).

B. Controlled ODE Model

To demonstrate the benefit of the state-space representation and the modeling of the force as GP we consider the model (1) with linear closed loop optimal control design for $c(t)$. In the experiment the mass-spring-damper system was started at static position $f(0) = -1$ and the aim was to keep it at the origin while the unknown latent force is affecting it. In the experiment we use the latent force hyper-parameters that were estimated from the open loop system considered in the previous section. We assumed that we only get measurements of the position, but with a

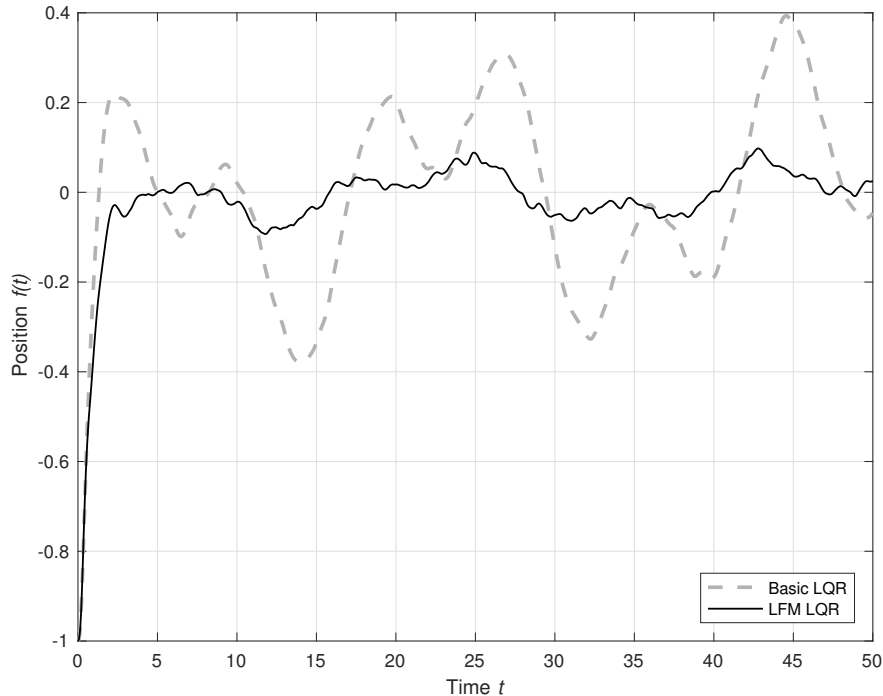


Fig. 5. Result of controlling the ODE model with Basic LQR and LFM LQR. It can be seen that the control designed for the full LFM outperforms the basic LQR significantly. The average position tracking error for the Basic LFM was approximately 0.21 units whereas in the case of LFM LQR it was approximately 0.11 units.

relatively small variance $\sigma^2 = 0.01^2$ —this was assumed to emphasize the differences between the control designs.

We consider two ways of designing the controller which were discussed in Section IV-A: using the assumed separability design based on putting $u(t) = 0$ and a controller which is designed by taking into account the existence of the latent force as described in the same section. The results of using the basic linear quadratic regulator (LQR), that is, the certainty equivalent design, and the result of using the joint LFM control are shown in Figure 5. It can be seen that the LFM controller is able to maintain the system much better near the origin than the basic controller.

C. Inference on Poisson equation

In this example, we model the 2D Poisson equation where our aim is to infer the unknown input field $u(x, y)$ from noisy observations y_k , as in Eq. (4). We assume that the source follows a GP prior with an SE covariance. The input space consists of two space variables that we refer

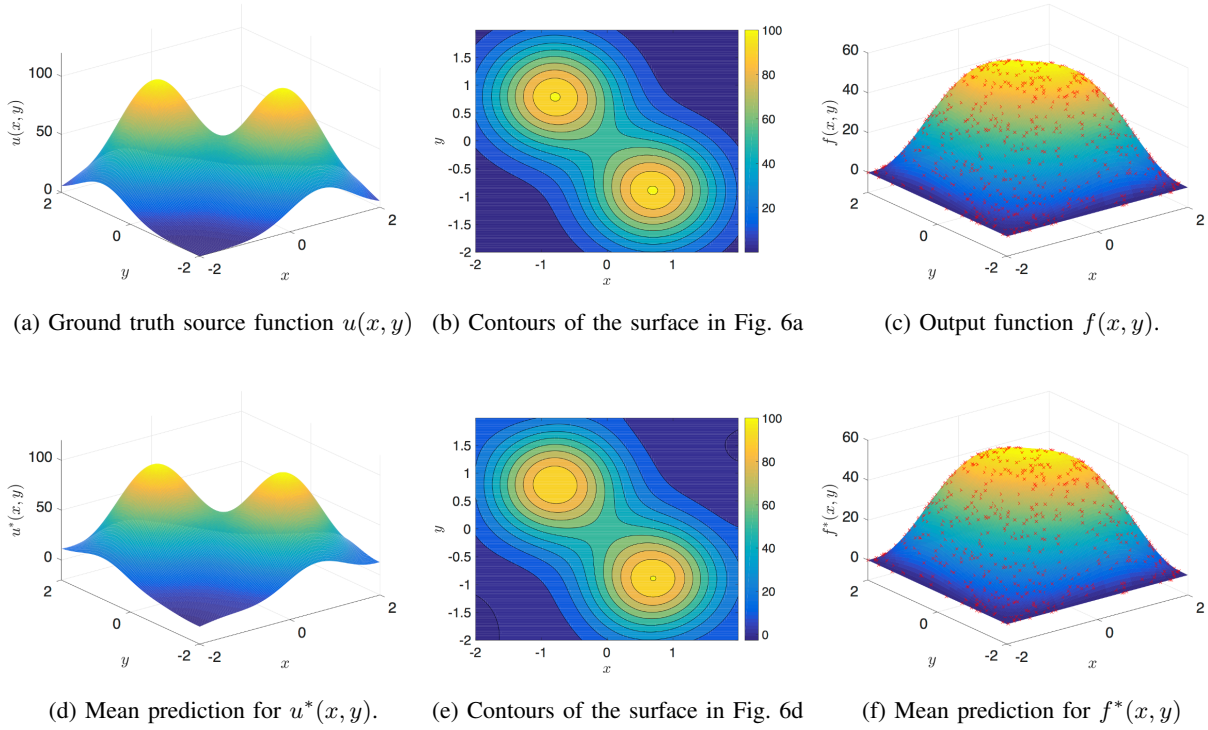


Fig. 6. The upper row shows the ground truth data for the 2D PDE Poisson example: the input source (Fig. 6a); contours of the ground truth input source 6b; ground truth for the output function (Fig. 6c). Noisy observations are shown as red crosses in Fig. 6c. The lower row shows the predicted mean function $\mathbb{E}[u^*(x, y)|\mathbf{y}]$ (Fig. 6d); contours of the predicted mean function (Fig. 6e); the predicted mean output $\mathbb{E}[f^*(x, y)|\mathbf{y}]$ (Fig. 6f). Noisy observations are again shown as red crosses in Fig. 6f.

to as x and y . The covariance function for the source is given as

$$k(x - x', y - y') = \exp \left[-\frac{(x - x')^2}{\sigma_x^2} \right] \exp \left[-\frac{(y - y')^2}{\sigma_y^2} \right],$$

where σ_x , and σ_y are the length-scales along the spatial dimensions.

Assuming boundary conditions equal to zero, we can use the corresponding Green's function (see [11] for mathematical details) to compute covariance functions for the output $k_{f,f}(x, y, x', y')$, and between the output and the latent source $k_{f,u}(x, y, x', y')$. We can then use GP regression expressions to infer the posterior distribution $p(u^* | \mathbf{y})$.

In this example, the input field $u(x, y)$ is given by

$$u(x, y) = 100 \exp \left\{ -\frac{[(x + 0.8)^2 + (y - 0.8)^2]}{s^2} \right\} + 100 \exp \left\{ -\frac{[(x - 0.7)^2 + (y + 0.9)^2]}{s^2} \right\},$$

where $s^2 = 1$. Figure 6a shows a 3D description of the input source in the domain $[-2, 2] \times [-2, 2]$. Figure 6b shows contours of the same input source. Figure 6c corresponds to the output function $f(t)$ obtained by solving the Poisson equation with an input source given by the function in Figure 6a.

The output function is made of 17,956 observation points distributed in a grid of 134×134 inputs uniformly organized in the input domain. From those observations, we randomly choose $n = 1000$ spatial points with their corresponding function evaluations $\{f_k\}_{k=1}^n$ as part of the data used to estimate the parameters σ_x^2 and σ_y^2 . Furthermore, we add Gaussian noise with variance 0.01 to the function evaluations, and obtain the noisy observations $\{y_k\}_{k=1}^n$. Figure 6c shows the noisy observations as red crosses. For obtaining estimates of σ_x^2 and σ_y^2 , we maximize the marginal log-likelihood function with respect to the parameters.

Figure 6d shows the mean prediction of the distribution $p(u^* | \mathbf{y})$. Figure 6e shows contours for the estimated mean function. We notice that the mean predictive function $\mathbb{E}[u^*(x, y) | \mathbf{y}]$ is qualitatively similar to the ground truth function $u(x, y)$. Figure 6f shows the mean of the predictive distribution $p(f^*(x, y) | \mathbf{y})$. We note the similarity between the ground truth in Figure 6c and the mean prediction in Figure 6f. Furthermore, the MSE between the ground truth source $u(x, y)$, and the inferred mean source $u^*(x, y)$ is equal to 0.5153. The MSE between the ground truth output $f(x, y)$ and the mean predicted output $f^*(x, y)$ evaluated at the test input points is 8.3354×10^{-4} .

D. Controlled heat equation

In this experiment we consider a controlled heat equation of the form

$$\frac{\partial f(\mathbf{x}, t)}{\partial t} = D \nabla^2 f(\mathbf{x}, t) - \lambda f(\mathbf{x}, t) + u(\mathbf{x}, t) + c(\mathbf{x}, t), \quad (88)$$

where $u(\mathbf{x}, t)$ is an unknown latent force, $c(\mathbf{x}, t)$ is the control signal, $\mathbf{x} \in \mathbb{R}^2$, and $\lambda, D > 0$ are positive constants. Figure 7 is a cartoon representation of the simulated scenario which is a heat source moving across a 2D spatial field. The field is measured at a discrete grid and the measurements are corrupted by Gaussian noise. In the simulation, the latent force $u(\mathbf{x}, t)$ is the heat generated by the moving source and the aim is to reconstruct f and u from noisy observations as well as design an optimal control signal $c(\mathbf{x}, t)$, which aims to regulate the temperature $f(\mathbf{x}, t)$ to zero.

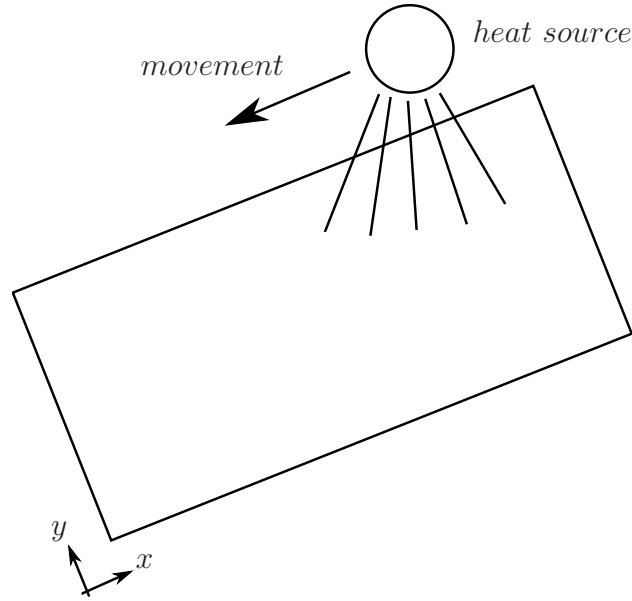


Fig. 7. A cartoon representation of a heat source moving across a 2D spatial field.

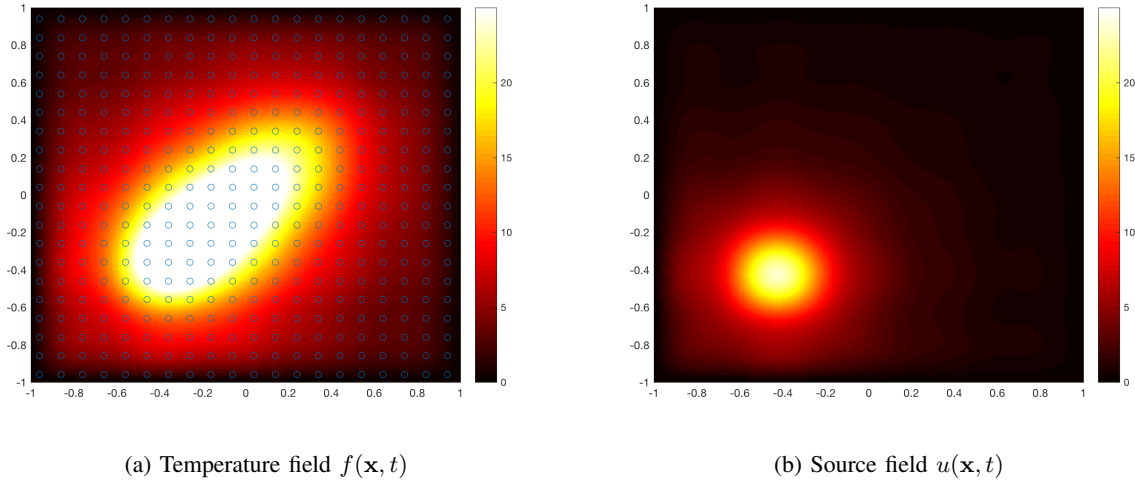


Fig. 8. The temperature function $f(\mathbf{x}, t)$ and the source function $u(\mathbf{x}, t)$ at time $t = 6.9$. The small circles mark the positions of the measurements.

In the simulation, we used the parameters $\lambda = 0.2$ and $D = 0.001$ and the heat source was moving for 10 seconds from top-right to bottom-left direction and then it was turned off. The temperature then increases at the application point and when the heat source moves away, the position starts cooling down. Figures 8a and 8b show the temperature field and the heat source at time $t = 6.9$ when no control is applied.

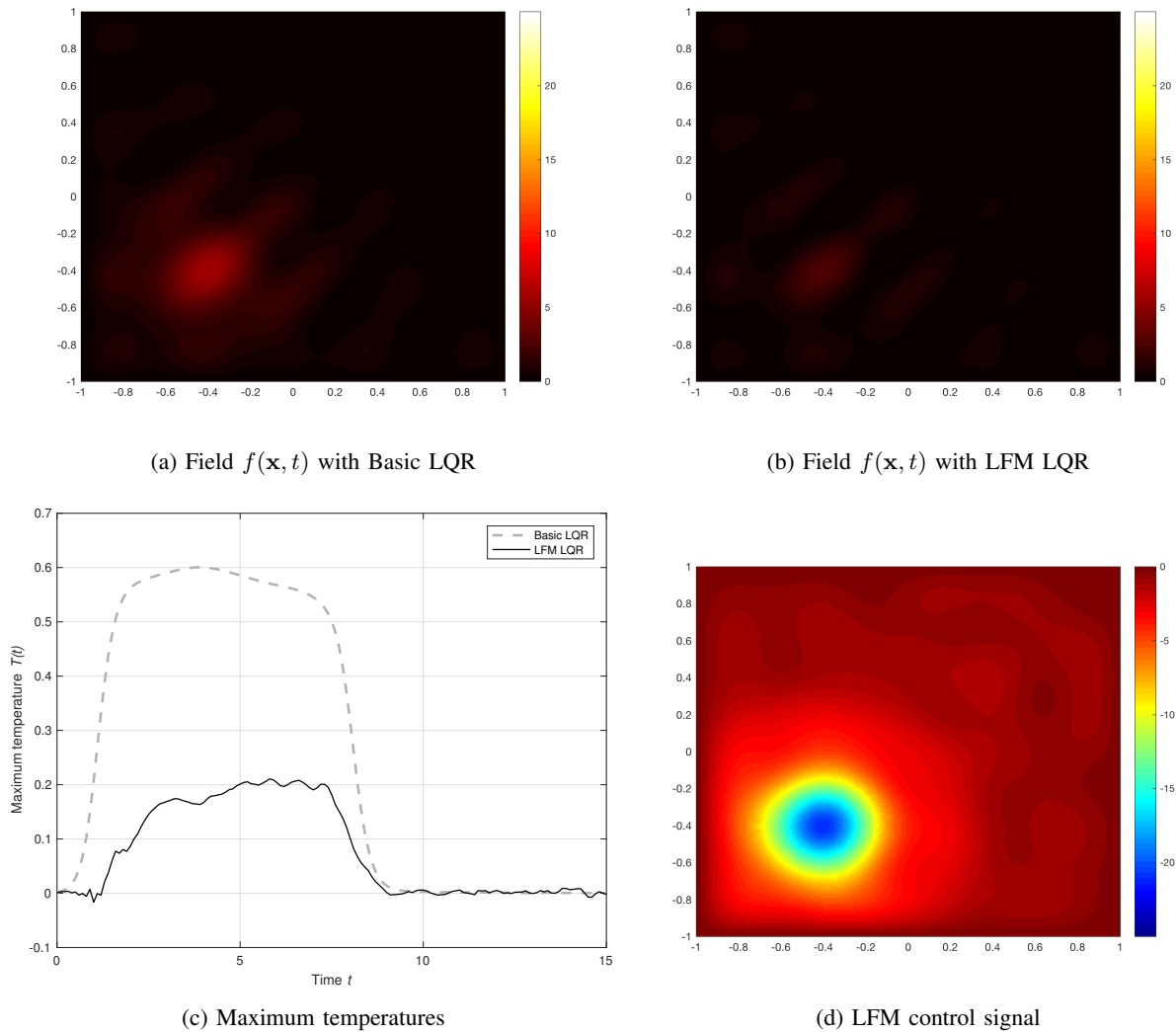


Fig. 9. The results of using Basic LQR and LFM LQR controllers to regulate the temperature field to zero. It can be seen from Figures 9a, 9b, and 9c that LFM LQR is able to keep the temperature closer to zero than Basic LQR. Figure 9d shows an example control signal which can be seen to effectively cancel out the latent force model part as one would expect.

We then formed a Fourier-basis approximation to the PDE (with 100 basis functions) and designed two controllers for it—one using an assumed separability design (“Basic LQR”) and one by taking the latent force into account (“LFM LQR”). We used SE covariance functions for the latent force model in both time and space directions. A Kalman filter was used to estimate the physical system and latent force states from temperature measurements with low variance ($\sigma^2 = 0.01^2$) and the controller was applied using the estimate.

Figures 9a – 9d show the results when the controllers were used. It can be seen that the LFM LQR provides a significantly smaller tracking error.

VII. CONCLUSIONS

In this paper we have presented a latent force model (LFM) framework for learning and control in hybrid systems which are combinations of first-principles (physical) models and non-parametric Gaussian process models as their inputs. We have presented both covariance function based inference methods and state-space based inference methods for the models as well as analyzed the detectability and observability properties of LFMs. Additionally, we have extended the methodology to allow for stochastic control of the (state-space) LFMs. Although the full models are uncontrollable, they are output controllable with respect to the physical model part which is enough for practical implementation of the controllers.

REFERENCES

- [1] C. E. Rasmussen and C. K. Williams, *Gaussian Processes for Machine Learning*. The MIT Press, 2006.
- [2] M. Álvarez, D. Luengo, and N. D. Lawrence, “Latent force models,” in *JMLR Workshop and Conference Proceedings Volume 5 (AISTATS 2009)*, 2009, pp. 9–16.
- [3] M. Álvarez, J. R. Peters, N. D. Lawrence, and B. Schölkopf, “Switched latent force models for movement segmentation,” in *Advances in neural information processing systems*, 2010, pp. 55–63.
- [4] M. A. Álvarez, D. Luengo, and N. D. Lawrence, “Linear latent force models using Gaussian processes,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 11, pp. 2693–2705, 2013.
- [5] J. Hartikainen and S. Särkkä, “Sequential inference for latent force models,” in *Proceedings of The 27th Conference on Uncertainty in Artificial Intelligence (UAI 2011)*, 2011.
- [6] J. Hartikainen, M. Seppänen, and S. Särkkä, “State-space inference for non-linear latent force models with application to satellite orbit prediction,” in *Proceedings of the 29th International Conference on Machine Learning (ICML)*, 2012.
- [7] Y. Bar-Shalom, X.-R. Li, and T. Kirubarajan, *Estimation with Applications to Tracking and Navigation*. Wiley, 2001.
- [8] K. P. Murphy, *Machine Learning: A Probabilistic Perspective*. The MIT Press, 2012.
- [9] M. Filippone, M. Zhong, and M. Girolami, “A comparative evaluation of stochastic-based inference methods for Gaussian process models,” *Machine Learning*, vol. 93, no. 1, pp. 93–114, 2013. [Online]. Available: <http://dx.doi.org/10.1007/s10994-013-5388-x>
- [10] T. Myint-U and L. Debnath, *Linear Partial Differential Equations for Scientists and Engineers*, 4th ed. Birkhäuser, 2006.
- [11] A. D. Polyanin, *Handbook of Linear Partial Differential Equations for Engineers and Scientists*. Chapman & Hall/CRC Press, 2002.
- [12] S. Särkkä, “Linear operators and stochastic partial differential equations in Gaussian process regression,” in *Proceedings of ICANN*, 2011.
- [13] M. Titsias, N. D. Lawrence, and M. Rattray, “Efficient sampling for Gaussian process inference using control variables,” in *NIPS*, D. Koller, D. Schuurmans, Y. Bengio, and L. Bottou, Eds. Cambridge, MA: MIT Press, 2009, vol. 21, pp. 1681–1688.
- [14] M. Titsias, A. Honkela, N. Lawrence, and M. Rattray, “Identifying targets of multiple co-regulating transcription factors from expression time-series by Bayesian model comparison,” *BMC Systems Biology*, vol. 6, no. 1, p. 53, 2012. [Online]. Available: <http://www.biomedcentral.com/1752-0509/6/53>

- [15] J. Hartikainen and S. Särkkä, “Kalman filtering and smoothing solutions to temporal Gaussian process regression models,” in *Proceedings of IEEE International Workshop on Machine Learning for Signal Processing (MLSP)*, 2010.
- [16] S. Särkkä, A. Solin, and J. Hartikainen, “Spatiotemporal learning via infinite-dimensional Bayesian filtering and smoothing,” *IEEE Signal Processing Magazine*, vol. 30, no. 4, pp. 51–61, 2013.
- [17] S. Särkkä and R. Piché, “On convergence and accuracy of state-space approximations of squared exponential covariance functions,” in *Proceedings of 2014 IEEE International Workshop on Machine Learning for Signal Processing (MLSP)*, 2014, pp. 1–6.
- [18] S. Särkkä, *Bayesian Filtering and Smoothing*. Cambridge University Press, 2013.
- [19] T. Karvonen and S. Särkkä, “Approximate state-space Gaussian processes via spectral transformation,” in *Proceedings of IEEE International Workshop on Machine Learning for Signal Processing (MLSP)*, 2016.
- [20] A. Solin and S. Särkkä, “Explicit link between periodic covariance functions and state space models,” in *Proceedings of the Seventeenth International Conference on Artificial Intelligence and Statistics (AISTATS 2014)*, ser. JMLR Workshop and Conference Proceedings, vol. 33, 2014, pp. 904–912.
- [21] A. Solin and S. Särkkä, “Gaussian quadratures for state space approximation of scale mixtures of squared exponential covariance functions,” in *Proceedings of the IEEE International Workshop on Machine Learning for Signal Processing (MLSP)*, 2014.
- [22] A. Solin, “Stochastic differential equation methods for spatio-temporal Gaussian process regression,” Doctoral dissertation, Aalto University, 2016.
- [23] B. Anderson, J. Moore, and S. Loo, “Spectral factorization of time-varying covariance functions,” *IEEE Trans. on Inf. Theory*, vol. 15, no. 5, pp. 550–557, 1969.
- [24] H. L. Van Trees, *Detection, Estimation, and Modulation Theory Part II*. John Wiley & Sons, New York, 1971.
- [25] S. Särkkä and J. Hartikainen, “Infinite-dimensional Kalman filtering approach to spatio-temporal Gaussian process regression,” in *JMLR Workshop and Conference Proceedings Volume 22 (AISTATS 2012)*, 2012, pp. 993–1001.
- [26] A. Solin and S. Särkkä, “Infinite-dimensional Bayesian filtering for detection of quasiperiodic phenomena in spatiotemporal data,” *Physical Review E*, vol. 88, p. 052909, Nov 2013. [Online]. Available: <http://link.aps.org/doi/10.1103/PhysRevE.88.052909>
- [27] A. Solin, M. Kok, N. Wahlström, T. B. Schön, and S. Särkkä, “Modeling and interpolation of the ambient magnetic field by Gaussian processes,” *IEEE Transactions on Robotics*, 2017, conditionally accepted.
- [28] M. S. Grewal and A. P. Andrews, *Kalman Filtering, Theory and Practice Using MATLAB*. Wiley, 2001.
- [29] S. Särkkä, “Recursive Bayesian inference on stochastic differential equations,” Doctoral dissertation, Helsinki University of Technology, 2006.
- [30] R. E. Kalman, “Contributions to the theory of optimal control,” *Boletín de la Sociedad Matemática Mexicana*, vol. 5, no. 1, pp. 102–119, 1960.
- [31] —, “Mathematical description of linear dynamical systems,” *Journal of the Society for Industrial and Applied Mathematics, Series A: Control*, vol. 1, no. 2, pp. 152–192, 1963.
- [32] B. Anderson and J. B. Moore, “Detectability and stabilizability of time-varying discrete-time linear systems,” *SIAM Journal on Control and Optimization*, vol. 19, no. 1, pp. 20–32, 1981.
- [33] R. F. Curtain and H. Zwart, *An introduction to infinite-dimensional linear systems theory*. Springer Science & Business Media, 2012, vol. 21.
- [34] F. Ding, L. Qiu, and T. Chen, “Reconstruction of continuous-time systems from their non-uniformly sampled discrete-time systems,” *Automatica*, vol. 45, no. 2, pp. 324–332, 2009.

- [35] E. G. Gilbert, “Controllability and observability in multivariable control systems,” *Journal of the Society for Industrial and Applied Mathematics, Series A: Control*, vol. 1, no. 2, pp. 128–151, 1963.
- [36] C. T. Chen and C. Desoer, “Controllability and observability of composite systems,” *IEEE Transactions on Automatic Control*, vol. 12, no. 4, 1967.
- [37] E. Davison and S. Wang, “New results on the controllability and observability of general composite systems,” *IEEE Transactions on Automatic Control*, vol. 20, no. 1, pp. 123–128, 1975.
- [38] K. Ogata, *Modern control engineering*, 3rd ed. Prentice Hall, 1997.
- [39] B. D. O. Anderson and J. B. Moore, *Optimal Control: Linear Quadratic Methods*. Dover, 2007.
- [40] W. M. Wonham, *Linear Multivariable Control: A Geometric Approach*. Springer-Verlag, 1985.
- [41] N. D. Lawrence, G. Sanguinetti, and M. Rattray, “Modelling transcriptional regulation using Gaussian processes,” in *Advances in Neural Information Processing Systems*, 2007, pp. 785–792.
- [42] P. Gao, A. Honkela, M. Rattray, and N. D. Lawrence, “Gaussian process modelling of latent chemical species: Applications to inferring transcription factor activities,” *Bioinformatics*, vol. 24, no. 16, pp. i70–i75, 2008.
- [43] P. Maybeck, *Stochastic Models, Estimation and Control, Volume 3*. Academic Press, 1982.
- [44] M. A. Álvarez, J. Peters, B. Schölkopf, and N. D. Lawrence, “Switched latent force models for movement segmentation,” in *NIPS*. Cambridge, MA: MIT Press, 2011, vol. 24, pp. 55–63.
- [45] J. Kaipio and E. Somersalo, *Statistical and Computational Inverse Problems*. Springer, 2005.
- [46] A. Solin and S. Särkkä, “Hilbert space methods for reduced-rank Gaussian process regression,” *ArXiv preprint arXiv:1401.5508*, 2014.
- [47] F. Lindgren, H. Rue, and J. Lindström, “An explicit link between Gaussian fields and Gaussian Markov random fields: the stochastic partial differential equation approach,” *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 73, no. 4, pp. 423–498, 2011.



Michael Shell Biography text here.

John Doe Biography text here.

Jane Doe Biography text here.