Available online at www.sciencedirect.com

ScienceDirect

Comput. Methods Appl. Mech. Engrg. 287 (2015) 69–89

Computer methods in applied mechanics and engineering

www.elsevier.com/locate/cma

Gaussian functional regression for linear partial differential equations

N.C. Nguyen*, J. Peraire

MIT Department of Aeronautics and Astronautics, 77 Massachusetts Ave., Cambridge, MA 02139, USA

Received 13 May 2014; received in revised form 8 January 2015; accepted 9 January 2015
Available online 19 January 2015

Abstract

In this paper, we present a new statistical approach to the problem of incorporating experimental observations into a mathematical model described by linear partial differential equations (PDEs) to improve the prediction of the state of a physical system. We augment the linear PDE with a functional that accounts for the uncertainty in the mathematical model and is modeled as a *Gaussian process*. This gives rise to a stochastic PDE which is characterized by the Gaussian functional. We develop a *Gaussian functional regression* method to determine the posterior mean and covariance of the Gaussian functional, thereby solving the stochastic PDE to obtain the posterior distribution for our prediction of the physical state. Our method has the following features which distinguish itself from other regression methods. First, it incorporates both the mathematical model and the observations into the regression procedure. Second, it can handle the observations given in the form of linear functionals of the field variable. Third, the method is non-parametric in the sense that it provides a systematic way to optimally determine the prior covariance operator of the Gaussian functional based on the observations. Fourth, it provides the posterior distribution quantifying the magnitude of uncertainty in our prediction of the physical state. We present numerical results to illustrate these features of the method and compare its performance to that of the standard Gaussian process regression.

Keywords: Gaussian functionals; Gaussian processes; Regression analysis; Bayesian inference; Data assimilation; Linear PDEs

1. Introduction

Partial differential equations (PDEs) are used to mathematically model a wide variety of physical phenomena such as heat transfer, fluid flows, electromagnetism, and structural deformations. A PDE model of a physical system is typically described by conservation laws, constitutive laws, material properties, boundary conditions, boundary data, and geometry. In practical applications, the mathematical model described by the PDEs is only an approximation to the real physical system due to (i) the deliberate simplification of the mathematical model to keep it tractable (by ignoring certain physics or certain boundary conditions that pose computational difficulties), and (ii) the uncertainty

* Corresponding author.
E-mail addresses: cuongng@mit.edu (N.C. Nguyen), peraire@mit.edu (J. Peraire).

http://dx.doi.org/10.1016/j.cma.2015.01.008

of the available data (by using geometry, material property and boundary data that are not exactly the same as those of the physical system). We refer to the PDE model (available to us) as the *best knowledge PDE model* [1] and to its solution as the *best knowledge state*. To assess the accuracy of the best knowledge model in predicting the physical system, the best knowledge state needs to be compared against experimental data, which typically will have some level of noise.

In cases where the discrepancy between the PDE model and the experimental data is beyond an acceptable level of accuracy, we need to improve the current PDE model. There are several approaches to defining a new improved model. *Parameter estimation* [2,3] involves calibrating some parameters in the model to match the data. An alternative approach to obtain an improved model is *data assimilation* [4–8]. Broadly speaking, data assimilation is a numerical procedure by which we incorporate observations into a mathematical model to reflect the errors inherent in our mathematical modeling of the physical system. Although data assimilation shares the same objective as parameter estimation, it differs from the latter in methodology. More specifically, data assimilation does not assume any parameters to be calibrated; instead, data assimilation defines a new model that matches the observations as well as possible, while being as close as possible to the best knowledge model. Another approach is *data interpolation* [9–14] which involves computing a collection of solutions (snapshots) of a parametrized or time-varying mathematical model and reconstructing the physical state by fitting the experimental data to the snapshots.

A widely used technique for obtaining an improved model in parameter estimation and data assimilation is *least squares regression* [15,5,16]. Least squares is a deterministic regression approach that provides an estimate for the physical state which is optimal in least squares sense. However, it does not provide a means to quantify the prediction uncertainty. A recent work [1] poses the least-square regression as a regularized saddle point Galerkin formulation which admits interpretation from a variational framework and permits its extension to *Petrov–Galerkin formulation*. While the Petrov–Galerkin formulation provides more flexibility than the Galerkin formulation, it does not quantify the uncertainty in the prediction either. A popular statistical approach in parameter estimation and data assimilation is *Bayesian inference* [17–20]. In Bayesian inference an estimate of the physical state is described by random variables and the posterior probability distribution of the estimate is determined by the data according to Bayes' rule [18,19]. Therefore, Bayesian inference provides a powerful framework to quantify the prediction uncertainties.

In this paper, we introduce a new statistical approach to the problem of incorporating observations into the best knowledge model to predict the state of a physical system. Our approach has its root in Gaussian process (GP) regression [21–23]. We augment the linear PDE with a functional that accounts for the uncertainty in the mathematical model and is modeled as a *Gaussian process*.[1] This gives rise to a stochastic PDE whose solutions are characterized by the Gaussian functional. By extending the standard GP regression for *functions of vectors* to *functionals of functions*, we develop a *Gaussian functional regression* method to determine the posterior distribution of the Gaussian functional, thereby solving the stochastic PDE for our prediction of the physical state. Our method is devised as follows. We first derive a *functional regression problem* by making use of the adjoint states and the observations. We next solve the functional regression problem by an application of the principle of Gaussian processes to obtain the posterior mean and covariance of the Gaussian functional. Finally, we compute the posterior distribution for our estimate of the physical state. A crucial ingredient in our method is the *covariance operator* representing the *prior* of the Gaussian functional. The bilinear covariance operators considered incorporate a number of free parameters (the so-called *hyperparameters*) that can be optimally determined from the measured data by maximizing a marginal likelihood.

Our Gaussian functional regression method can be viewed as a generalization of the standard GP regression from a finite dimensional vector (input) space to an infinite dimensional function (input) space. GP regression is a well-established technique to construct maps between inputs and outputs based on a set of sample, or training, input and output pairs, but does not offer a direct method to incorporate prior knowledge, albeit approximate, from an existing model relating the inputs and outputs. By combining the best knowledge model with the data, our method can greatly improve the prediction of the physical system.

Furthermore, we introduce a *nonparametric Bayesian inference* method for linear functional regression with Gaussian noise. It turns out that nonparametric Bayesian inference and Gaussian functional regression represent two different views of the same procedure. Specifically, we can think of Gaussian functional regression as defining a

[1] In the cases considered, the physical system is not stochastic but deterministic. The introduction of the Gaussian functional serves to represent uncertainties in the best knowledge model and in the data, *not* in the physical system *per se*.

distribution over functionals and doing inference in the space of functionals — the *functional-space view*. We can think of nonparametric Bayesian inference as defining a distribution over weights and doing inference in the space of weights — the *weight-space view*. Theoretically, Gaussian functional regression can be interpreted as an application of the *kernel trick* [22] to nonparametric Bayesian inference, thereby avoiding an explicit construction of the feature map.

The paper is organized as follows. In Section 2, we present a description of the problem considered. In Section 3, we give an overview of Gaussian processes for regression problems. In Section 4, we introduce Gaussian functional regression for state estimation problems via linear PDE models. In Sections 5 and 6, we present numerical results to demonstrate our method and compare its performance to that of function Gaussian process regression. In Section 7, we provide some concluding remarks on future research. Finally, in the Appendix, we describe our nonparametric Bayesian inference method.

2. Motivation and problem statement

Let $\Omega^{\text{true}} \in \mathbb{R}^n$ denote a bounded open domain with Lipschitz boundary. Let $V^{\text{true}}(\Omega^{\text{true}})$ be an appropriate real-valued function space in which the true state u^{true} resides. A weak formulation of the true PDE model can be stated as: Find $u^{\text{true}} \in V^{\text{true}}$ and $\boldsymbol{s}^{\text{true}} \in \mathbb{R}^M$ such that

$$a^{\text{true}}(u^{\text{true}}, v) = \ell^{\text{true}}(v), \quad \forall\, v \in V^{\text{true}}, \tag{1a}$$

$$s_i^{\text{true}} = c_i^{\text{true}}(u^{\text{true}}), \quad i = 1, \dots, M, \tag{1b}$$

where $a^{\text{true}} : V^{\text{true}} \times V^{\text{true}} \to \mathbb{R}$ is a bilinear form, $\ell^{\text{true}} : V^{\text{true}} \to \mathbb{R}$ is a linear functional, and $c_i^{\text{true}} : V^{\text{true}} \to \mathbb{R}$, $i = 1, \dots, M$ are observation functionals. We assume that the true PDE model (1) is well defined and accurately describes the physical system of interest.

In actual practice, we do not have access to a^{true}, ℓ^{true}, c_i^{true}, and $V^{\text{true}}(\Omega^{\text{true}})$. Hence, we cannot compute u^{true} and $\boldsymbol{s}^{\text{true}}$. However, we assume that we have access to the “best knowledge” of a^{true}, ℓ^{true}, c_i^{true}, $i = 1, \dots, M$, and $V^{\text{true}}(\Omega^{\text{true}})$, which shall be denoted by a^{bk}, ℓ^{bk}, c_i^{bk}, $i = 1, \dots, M$, and $V^{\text{bk}}(\Omega^{\text{bk}})$, respectively. We then define the best knowledge PDE model: Find $u^{\text{bk}} \in V^{\text{bk}}$ and $\boldsymbol{s}^{\text{bk}} \in \mathbb{R}^M$ such that

$$a^{\text{bk}}(u^{\text{bk}}, v) = \ell^{\text{bk}}(v), \quad \forall\, v \in V^{\text{bk}}, \tag{2a}$$

$$s_i^{\text{bk}} = c_i^{\text{bk}}(u^{\text{bk}}), \quad i = 1, \dots, M. \tag{2b}$$

In the remainder of this paper, we shall drop the superscript “bk” for the quantities associated with the best knowledge model to simplify the notation. (In practice, we replace the continuous function space $V(\Omega)$ with a finite approximation space, which is assumed to be large enough that the numerical solution is indistinguishable from the continuous one.)

We now assume that we are given the observed data $\boldsymbol{d} \in \mathbb{R}^M$, which are the M measurements of the true output vector $\boldsymbol{s}^{\text{true}}$. We further assume that the measurements differ from the true outputs $\boldsymbol{s}^{\text{true}}$ by additive Gaussian noise $\boldsymbol{\varepsilon}$, namely,

$$\boldsymbol{d} = \boldsymbol{s}^{\text{true}} + \boldsymbol{\varepsilon}, \tag{3}$$

where ε_i, $i = 1, \dots, M$ are independent, identically distributed Gaussian distributions with zero mean and variance σ^2. If σ is sufficiently small within the acceptable accuracy then we can use the observed data $\boldsymbol{d}$ to validate the best knowledge model (2). If the best knowledge outputs $\boldsymbol{s}$ are close enough to $\boldsymbol{d}$ within the noise level then we may trust the best knowledge model to predict the behavior of the true model. In many cases, the best knowledge outputs do not match the observed data due to various sources of uncertainty from physical modeling, constitutive laws, boundary conditions, boundary data, material properties, and geometry.

We are interested in improving the best knowledge model when it does not produce a good estimate of the true state. In particular, we propose a method to compute a better estimate for the true state by combining the best knowledge model with the observed data. Our method has its root in Gaussian process regression. Before proceeding to describe the proposed method we review the ideas behind Gaussian processes.

3. Gaussian process regression

We begin by assuming that we are given a training set of M observations

$$S = \{(\boldsymbol{x}_i, y_i), i = 1, \dots, M\}, \tag{4}$$

where $\boldsymbol{x}_i \in \mathbb{R}^N$ denotes an input vector of dimension N and $y_i \in \mathbb{R}$ denotes a scalar real-valued output. The training input vectors $\boldsymbol{x}_i$, $i = 1, \dots, M$ are aggregated in the $N \times M$ real-valued matrix $\boldsymbol{X}$, and the outputs are collected in the real-valued vector $\boldsymbol{y}$, so we can write $S = (\boldsymbol{X}, \boldsymbol{y})$. We assume that

$$y_i = h(\boldsymbol{x}_i) + \mathcal{N}(0, \sigma^2), \tag{5}$$

where $h(\boldsymbol{x})$ is the true but unknown function which we want to infer. The unknown function $h(\boldsymbol{x})$ is modeled as a Gaussian process[2] with zero mean,[3] for simplicity, and covariance function $\kappa(\boldsymbol{x}, \boldsymbol{x}')$, namely,

$$h(\boldsymbol{x}) \sim \mathcal{GP}(0, \kappa(\boldsymbol{x}, \boldsymbol{x}')). \tag{6}$$

From a Bayesian perspective, we encode our belief that instances of $h(\boldsymbol{x})$ are drawn from a Gaussian process with zero mean and covariance function $\kappa(\boldsymbol{x}, \boldsymbol{x}')$ *prior to taking into account observations*.[4] Mathematically speaking, $h(\boldsymbol{x})$ is assumed to reside in a reproducing kernel Hilbert space [24] spanned by the eigenfunctions of the covariance function $\kappa(\boldsymbol{x}, \boldsymbol{x}')$.

Let $\boldsymbol{X}^*$ be the $N \times M^*$ matrix that contains M^* test input vectors $\boldsymbol{x}_j^*$, $j = 1, \dots, M^*$ as its columns. Since $h(\boldsymbol{x}) \sim \mathcal{GP}(0, \kappa(\boldsymbol{x}, \boldsymbol{x}'))$, the joint distribution of the observed outputs and the function values at the test input vectors is

$$\begin{bmatrix} \boldsymbol{y} \\ \boldsymbol{h}^* \end{bmatrix} \sim \mathcal{N}\left(0, \begin{bmatrix} \boldsymbol{\mathcal{K}}(\boldsymbol{X}, \boldsymbol{X}) + \sigma^2 \boldsymbol{I} & \boldsymbol{\mathcal{K}}(\boldsymbol{X}, \boldsymbol{X}^*) \\ \boldsymbol{\mathcal{K}}(\boldsymbol{X}^*, \boldsymbol{X}) & \boldsymbol{\mathcal{K}}(\boldsymbol{X}^*, \boldsymbol{X}^*) \end{bmatrix}\right), \tag{7}$$

where $\boldsymbol{h}^* \in \mathbb{R}^M$, $\boldsymbol{\mathcal{K}}(\boldsymbol{X}, \boldsymbol{X}) \in \mathbb{R}^{M \times M}$, $\boldsymbol{\mathcal{K}}(\boldsymbol{X}, \boldsymbol{X}^*) \in \mathbb{R}^{M \times M^*}$, $\boldsymbol{\mathcal{K}}(\boldsymbol{X}^*, \boldsymbol{X}) \in \mathbb{R}^{M^* \times M}$, and $\boldsymbol{\mathcal{K}}(\boldsymbol{X}^*, \boldsymbol{X}^*) \in \mathbb{R}^{M^* \times M^*}$ have entries

$$\begin{aligned}
&h_i^* = h(\boldsymbol{x}_i^*), \quad i = 1, \dots, M^*, \\
&\mathcal{K}_{ij}(\boldsymbol{X}, \boldsymbol{X}) = \kappa(\boldsymbol{x}_i, \boldsymbol{x}_j), \quad i = 1, \dots, M, j = 1, \dots, M, \\
&\mathcal{K}_{ij}(\boldsymbol{X}, \boldsymbol{X}^*) = \kappa(\boldsymbol{x}_i, \boldsymbol{x}_j^*), \quad i = 1, \dots, M, j = 1, \dots, M^*, \\
&\mathcal{K}_{ij}(\boldsymbol{X}^*, \boldsymbol{X}) = \kappa(\boldsymbol{x}_i^*, \boldsymbol{x}_j), \quad i = 1, \dots, M^*, j = 1, \dots, M, \\
&\mathcal{K}_{ij}(\boldsymbol{X}^*, \boldsymbol{X}^*) = \kappa(\boldsymbol{x}_i^*, \boldsymbol{x}_j^*), \quad i = 1, \dots, M^*, j = 1, \dots, M^*,
\end{aligned} \tag{8}$$

respectively. We next apply the conditional distribution formula (see [22]) to the joint distribution (7) to obtain the *predictive distribution* for $\boldsymbol{h}^*$ as

$$\boldsymbol{h}^* | \boldsymbol{y}, \boldsymbol{X}, \boldsymbol{X}^* \sim \mathcal{N}(\bar{\boldsymbol{h}}^*, \mathrm{cov}(\boldsymbol{h}^*)), \tag{9}$$

where

$$\begin{aligned}
&\bar{\boldsymbol{h}}^* = \boldsymbol{\mathcal{K}}(\boldsymbol{X}^*, \boldsymbol{X}) \boldsymbol{\alpha}, \\
&\mathrm{cov}(\boldsymbol{h}^*) = \boldsymbol{\mathcal{K}}(\boldsymbol{X}^*, \boldsymbol{X}^*) - \boldsymbol{\mathcal{K}}(\boldsymbol{X}^*, \boldsymbol{X}) \boldsymbol{C}^{-1} \boldsymbol{\mathcal{K}}(\boldsymbol{X}, \boldsymbol{X}^*),
\end{aligned} \tag{10}$$

and $\boldsymbol{\alpha} \in \mathbb{R}^M$ and $\boldsymbol{C} \in \mathbb{R}^{M \times M}$ are given by

$$\boldsymbol{C}\boldsymbol{\alpha} = \boldsymbol{y}, \qquad \boldsymbol{C} = \boldsymbol{\mathcal{K}}(\boldsymbol{X}, \boldsymbol{X}) + \sigma^2 \boldsymbol{I}. \tag{11}$$

[2] A Gaussian process is a generalization of the Gaussian probability distribution. A Gaussian process governs the properties of Gaussian random functions, whereas a Gaussian probability distribution describes the properties of Gaussian random variables (scalars or vectors).

[3] The Gaussian process is assumed to have zero mean because we can always subtract the original outputs $\boldsymbol{y}$ from its average $\bar{y} = \frac{1}{M} \sum_{i=1}^M y_i$ to obtain new outputs with zero average. We then work with the new outputs and add the average $\bar{y}$ to our Gaussian process estimator.

[4] At this point, one may ask what if our belief is wrong, that is, what if the covariance function κ is not correctly chosen. Of course, choosing a wrong covariance function will result in very poor prediction. Hence, the covariance function should not be chosen arbitrarily. As discussed later, Gaussian processes provide a framework for optimal selection of a covariance function based on the observed data.

Note that the predictive mean $\bar{\boldsymbol{h}}^*$ is a linear combination of M kernel functions, each one centered on a training input vector. Note also that the predictive covariance $\mathrm{cov}(\boldsymbol{h}^*)$ does not explicitly depend on the observed data $\boldsymbol{y}$, but only on the training input vectors $\boldsymbol{X}$ and the covariance function κ. This is a property of the Gaussian distribution. However, as discussed below, the covariance function κ can be determined by using the observed data. As a result, the predictive covariance implicitly depends on the observed data.

Typically, the covariance function κ has some free parameters $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_Q)$, so that the matrix $\boldsymbol{C}$ depends on $\boldsymbol{\theta}$. These free parameters are called *hyperparameters*. The hyperparameters have a significant impact on the predictive mean and covariance. They are determined by maximizing the log marginal likelihood (see [22]):

$$\log p(\boldsymbol{y}|\boldsymbol{X}, \boldsymbol{\theta}) = -\frac{1}{2}\boldsymbol{y}^T \boldsymbol{C}(\boldsymbol{\theta})^{-1}\boldsymbol{y} - \frac{1}{2}\log(\det(\boldsymbol{C}(\boldsymbol{\theta}))) - \frac{M}{2}\log(2\pi). \tag{12}$$

Once we choose a specific form for κ and determine its hyperparameters, we can compute $\bar{\boldsymbol{h}}^*$ and $\mathrm{cov}(\boldsymbol{h}^*)$ for any given $\boldsymbol{X}^*$. Gaussian processes also provide us a mean to choose an appropriate family among many possible families of covariance functions. Choosing a covariance function for a particular application involves both determining hyperparameters within a family and comparing across different families. This step is termed as *model selection* [22].

We see that the standard GP regression provides us not only a posterior mean, but also a posterior covariance which characterizes uncertainty in our prediction of the true function. Moreover, it allows us to determine the optimal covariance function and thus the optimal reproducing Kernel Hilbert space in which the true function is believed to reside. These features differentiate GP regression from parametric regression methods such as least-squares regression, which typically provides the maximum likelihood estimate only. However, GP regression tends to require larger sample sizes than parametric regression methods because the data must supply enough information to yield a good covariance function by using model selection.

There are a number of obstacles that prevent us from applying the standard GP regression to our problem of interest described in the previous section. First, our outputs are in general not the evaluations of the state at spatial coordinates. Instead, they are linear functionals of the state. Second, the standard GP regression described here does not allow us to make use of the best knowledge model. The best knowledge model plays an important role because it carries crucial prior information about the true model. By taking advantage of the best knowledge model, we may be able to use far less observations to obtain a good prediction and thus address the main disadvantage of the standard GP regression. We propose a new approach to overcome these obstacles.

4. Gaussian functional regression

4.1. A stochastic PDE model

Let $g : V \to \mathbb{R}$ be a linear functional. We introduce a new mathematical model: Find $u^* \in V$ and $\boldsymbol{s}^* \in \mathbb{R}^M$ such that

$$a(u^*, v) + g(v) = \ell(v), \quad \forall v \in V, \tag{13a}$$

$$s_i^* = c_i(u^*), \quad i = 1, \ldots, M. \tag{13b}$$

Notice that the new model (13) differs from the best knowledge model (2) by the functional g. We can determine u^* and $\boldsymbol{s}^*$ only if g is known. The functional g thus characterizes the solution u^* and the output vector $\boldsymbol{s}^*$ of the model (13). We note that if $g(v) = \ell(v) - a(u^{\mathrm{true}}, v)$ then $u^* = u^{\mathrm{true}}$. Unfortunately, this particular choice of g requires the true state u^{true} which we do not know and thus want to infer.

In order to capture various sources of uncertainty in the best knowledge model, we represent g as a *Gaussian functional*[5] with zero mean and covariance operator k, namely,

$$g(v) \sim \mathcal{GF}(0, k(v, v')), \quad \forall\, v, v' \in V. \tag{14}$$

[5] A Gaussian functional can be thought of as a generalization of the Gaussian process from a finite dimensional vector space to an infinite dimensional function space.

Notice that there are three main differences between the Gaussian functional (14) and the Gaussian process (6). First, g is a functional, whereas h is a function. Second, v is a function, whereas $\boldsymbol{x}$ is a vector. And third, $k(v, v')$ is generally a differential and integral operator of v and v', whereas $\kappa(\boldsymbol{x}, \boldsymbol{x}')$ is a function of $\boldsymbol{x}$ and $\boldsymbol{x}'$. We will require that the covariance operator $k : V \times V \to \mathbb{R}$ is symmetric positive-definite. That is,

$$k(v, v') = k(v', v), \quad \text{and} \quad k(v, v) > 0, \quad \forall v, v' \in V. \tag{15}$$

As the covariance operator k characterizes the space of all possible functionals prior to taking into account the observations, it plays an important role in our method. The selection of a covariance operator will be discussed later.

Since g is a Gaussian functional the model (13) becomes a stochastic PDE. In order to solve the stochastic PDE (13), we need to compute the *posterior mean* and *posterior covariance* of g after accounting for the observed data $\boldsymbol{d}$. To this end, we formulate a functional regression problem and describe a procedure for solving it as follows.

4.2. Functional regression problem

We first introduce the adjoint problems: for $i = 1, \dots, M$ we find $\phi_i \in V$ such that

$$a(v, \phi_i) = -c_i(v), \quad \forall v \in V. \tag{16}$$

We note that the adjoint states ϕ_i depend only on the output functionals c_i and the bilinear form a. It follows from (2), (13) and (16) that

$$g(\phi_i) = \ell(\phi_i) - a(u^*, \phi_i) = a(u, \phi_i) + c_i(u^*) = c_i(u^*) - c_i(u) = s_i^* - s_i, \tag{17}$$

for $i = 1, \dots, M$. Moreover, we would like our stochastic PDE to produce the outputs $\boldsymbol{s}^*$ that are consistent with the observed data $\boldsymbol{d}$ in such a way that

$$d_i = s_i^* + \mathcal{N}(0, \sigma^2), \quad i = 1, \dots, M. \tag{18}$$

This equation is analogous to (3) which relates the observed data $\boldsymbol{d}$ to the true outputs $\boldsymbol{s}^{\text{true}}$. We substitute $s_i^* = d_i - \mathcal{N}(0, \sigma^2)$ into (17) to obtain

$$d_i - s_i = g(\phi_i) + \mathcal{N}(0, \sigma^2), \quad i = 1, \dots, M. \tag{19}$$

Notice that this expression characterizes the relationship between $g(\phi_i)$ and $d_i - s_i$ in the same way (5) characterizes the relationship between $h(\boldsymbol{x}_i)$ and y_i.

We now introduce a training set of M observations

$$T = \{(\phi_i, d_i - s_i), \ i = 1, \dots, M\}, \tag{20}$$

and use this training set to learn about g. More specifically, we wish to determine $g(\phi^*)$ for any given $\phi^* \in V$ based on the training set T. This problem is similar to the regression problem described in the previous section and is named the *functional regression problem* to emphasize that the object of interest g is a functional. We next describe the solution of the functional regression problem.

4.3. Regression procedure

Let $\Phi = [\phi_1, \dots, \phi_M]$ be a collection of M adjoint states as determined by (16). Let $\Phi^* = [\phi_1^* \in V, \dots, \phi_M^* \in V]$ be a collection of M^* test functions. The joint distribution of the observed outputs and the functional values for the test functions according to the prior (14) is given by

$$\begin{bmatrix} \boldsymbol{d} - \boldsymbol{s} \\ \boldsymbol{g}^* \end{bmatrix} \sim \mathcal{N}\left(0, \begin{bmatrix} \boldsymbol{K}(\Phi, \Phi) + \sigma^2 \boldsymbol{I} & \boldsymbol{K}(\Phi, \Phi^*) \\ \boldsymbol{K}(\Phi^*, \Phi) & \boldsymbol{K}(\Phi^*, \Phi^*) \end{bmatrix}\right), \tag{21}$$

where $\boldsymbol{g}^* \in \mathbb{R}^{M^*}$, $\boldsymbol{K}(\Phi, \Phi) \in \mathbb{R}^{M \times M}$, $\boldsymbol{K}(\Phi, \Phi^*) \in \mathbb{R}^{M \times M^*}$, $\boldsymbol{K}(\Phi^*, \Phi) \in \mathbb{R}^{M^* \times M}$, and $\boldsymbol{K}(\Phi^*, \Phi^*) \in \mathbb{R}^{M^* \times M^*}$ have entries

Table 1
Comparison between Gaussian process regression and Gaussian functional regression. Note that the best knowledge model enters in the Gaussian functional regression through the adjoint states Φ and the best knowledge outputs $\boldsymbol{s}$.

Quantities	Gaussian process regression	Gaussian functional regression
Input	vector $\boldsymbol{x} \in \mathbb{R}^N$	function $v \in V$
Output	function $h(\boldsymbol{x})$	functional $g(v)$
Prior	$h \sim \mathcal{N}(0, \kappa(\cdot,\cdot))$	$g \sim \mathcal{N}(0, k(\cdot,\cdot))$
Kernel	function $\kappa : \mathbb{R}^N \times \mathbb{R}^N \to \mathbb{R}$	operator $k : V \times V \to \mathbb{R}$
Training inputs	$\boldsymbol{X} \in \mathbb{R}^{N\times M}$	$\Phi \in V^M$ adjoint states
Observations	$\boldsymbol{y} = h(\boldsymbol{X}) + \mathcal{N}(0,\sigma)$	$\boldsymbol{d} - \boldsymbol{s} = g(\Phi) + \mathcal{N}(0,\sigma)$
Coefficients	$\boldsymbol{C\alpha} = \boldsymbol{y}, \boldsymbol{C} = \left[\boldsymbol{\mathcal{K}}(\boldsymbol{X},\boldsymbol{X}) + \sigma^2\boldsymbol{I}\right]$	$\boldsymbol{D\beta} = \boldsymbol{d} - \boldsymbol{s}, \boldsymbol{D} = \left[\boldsymbol{K}(\Phi,\Phi) + \sigma^2\boldsymbol{I}\right]$
Test inputs	$\boldsymbol{X}^* \in \mathbb{R}^{N\times M^*}$	$\Phi^* \in V^{M^*}$
Mean	$\boldsymbol{\mathcal{K}}(\boldsymbol{X}^*,\boldsymbol{X})\boldsymbol{\alpha}$	$\boldsymbol{K}(\Phi^*,\Phi)\boldsymbol{\beta}$
Covariance	$\boldsymbol{\mathcal{K}}(\boldsymbol{X}^*,\boldsymbol{X}^*) - \boldsymbol{\mathcal{K}}(\boldsymbol{X}^*,\boldsymbol{X})\boldsymbol{C}^{-1}\boldsymbol{\mathcal{K}}(\boldsymbol{X},\boldsymbol{X}^*)$	$\boldsymbol{K}(\Phi^*,\Phi^*) - \boldsymbol{K}(\Phi^*,\Phi)\boldsymbol{D}^{-1}\boldsymbol{K}(\Phi,\Phi^*)$

$$\begin{aligned}
&g_i^* = g(\phi_i^*), \quad i = 1,\ldots,M^*,\\
&K_{ij}(\Phi,\Phi) = k(\phi_i,\phi_j), \quad i = 1,\ldots,M,\ j = 1,\ldots,M,\\
&K_{ij}(\Phi,\Phi^*) = k(\phi_i,\phi_j^*), \quad i = 1,\ldots,M,\ j = 1,\ldots,M^*,\\
&K_{ij}(\Phi^*,\Phi) = k(\phi_i^*,\phi_j), \quad i = 1,\ldots,M^*,\ j = 1,\ldots,M,\\
&K_{ij}(\Phi^*,\Phi^*) = k(\phi_i^*,\phi_j^*), \quad i = 1,\ldots,M^*,\ j = 1,\ldots,M^*,
\end{aligned} \tag{22}$$

respectively. It thus follows that the *predictive distribution* for $\boldsymbol{g}^*$ is

$$\boldsymbol{g}^*|(\boldsymbol{d}-\boldsymbol{s}),\ \Phi,\ \Phi^* \sim \mathcal{N}(\bar{\boldsymbol{g}}^*, \mathrm{cov}(\boldsymbol{g}^*)), \tag{23}$$

where

$$\bar{\boldsymbol{g}}^* = \boldsymbol{K}(\Phi^*,\Phi)\boldsymbol{\beta}, \qquad \mathrm{cov}(\boldsymbol{g}^*) = \boldsymbol{K}(\Phi^*,\Phi^*) - \boldsymbol{K}(\Phi^*,\Phi)\boldsymbol{D}^{-1}\boldsymbol{K}(\Phi,\Phi^*), \tag{24}$$

and $\boldsymbol{\beta} \in \mathbb{R}^M$ and $\boldsymbol{D} \in \mathbb{R}^{M\times M}$ are given by

$$\boldsymbol{D\beta} = \boldsymbol{d} - \boldsymbol{s}, \qquad \boldsymbol{D} = \boldsymbol{K}(\Phi,\Phi) + \sigma^2\boldsymbol{I}. \tag{25}$$

Notice that we have correspondence with function Gaussian process regression described in the previous section, when identifying $(\Phi, \boldsymbol{d}-\boldsymbol{s})$ with $(\boldsymbol{X},\boldsymbol{y})$, $(\Phi^*,\boldsymbol{g}^*)$ with $(\boldsymbol{X}^*,\boldsymbol{h}^*)$, and $k(\cdot,\cdot)$ with $\kappa(\cdot,\cdot)$.

While our approach share similarities with Gaussian process regression, it differs from the latter in many important ways. We summarize in Table 1 the differences between Gaussian process regression and Gaussian functional regression.

In the Appendix, we introduce a nonparametric Bayesian framework for linear functional regression with Gaussian noise. It turns out that this nonparametric Bayesian framework is equivalent to the Gaussian functional regression described here. In fact, Gaussian functional regression can be viewed as an application of the *kernel trick* to nonparametric Bayesian inference for linear functional regression, thereby avoiding the computation of the eigenfunctions of the covariance operator k. We next introduce a family of bilinear covariance operators and then describe a method for determining the hyperparameters.

4.4. Covariance operators

The covariance operator k is a crucial ingredient in our approach. Here, we consider a class of bilinear covariance operators parametrized by $\boldsymbol{\theta} = (\theta_1,\theta_2)$ of the form:

$$k(v,v';\boldsymbol{\theta}) = \theta_1 \int_\Omega vv'd\boldsymbol{x} + \theta_2 \int_\Omega \nabla v\cdot\nabla v'd\boldsymbol{x}. \tag{26}$$

More general forms of the covariance operator are possible provided that they are symmetric and positive definite.

In order for a covariance operator to be used in our method, we need to specify its hyperparameters $\boldsymbol{\theta}$. Fortunately, Gaussian processes allow us to determine the hyperparameters by using the observed data. In order to do this, we first calculate the probability of the observed data given the hyperparameters, or *marginal likelihood* and choose $\boldsymbol{\theta}$ so that this likelihood is maximized. We note from (21) that

$$p(\boldsymbol{d}-\boldsymbol{s}|\Phi,\boldsymbol{\theta}) = \mathcal{N}(0,\boldsymbol{D}(\boldsymbol{\theta})), \tag{27}$$

where the matrix $\boldsymbol{D}(\boldsymbol{\theta})$ as defined in (25) depends on $k(\cdot,\cdot;\boldsymbol{\theta})$ and thus on $\boldsymbol{\theta}$ as well. Rather than maximizing (27), it is more convenient to maximize the log marginal likelihood which is given by,

$$\log p(\boldsymbol{d}-\boldsymbol{s}|\Phi,\boldsymbol{\theta}) = -\frac{1}{2}(\boldsymbol{d}-\boldsymbol{s})^T\boldsymbol{D}(\boldsymbol{\theta})^{-1}(\boldsymbol{d}-\boldsymbol{s}) - \frac{1}{2}\log(\det(\boldsymbol{D}(\boldsymbol{\theta}))) - \frac{M}{2}\log(2\pi). \tag{28}$$

Thus, we find $\boldsymbol{\theta}$ by solving the maximization problem

$$\boldsymbol{\theta} = \arg\max_{\boldsymbol{\theta}'\in\mathbb{R}^2} \log p(\boldsymbol{d}-\boldsymbol{s}|\Phi,\boldsymbol{\theta}'). \tag{29}$$

Hence, the hyperparameters $\boldsymbol{\theta}$ are chosen as the maximizer of the log marginal likelihood.

Once we determine the covariance operator, we can compute $(\bar{\boldsymbol{g}}^*, \mathrm{cov}(\boldsymbol{g}^*))$ for any given set Φ^* of test functions as described in Section 4.3. Therefore, Gaussian functional regression is *non-parametric* in the sense that the hyperparameters are chosen in light of the observed data. In order words, the data are used to define both the prior covariance and the posterior covariance. In contrast, parametric regression methods use a number of parameters to define the prior and combine this prior with the data to determine the posterior prediction. It remains to describe how to compute the posterior mean and covariance of the solution u^* of the stochastic PDE.

4.5. *Computation of the mean state and covariance*

We recall that our stochastic PDE model consists of finding $u^* \in V$ such that

$$a(u^*, v) = \ell(v) - g(v), \quad \forall v \in V. \tag{30}$$

Let $\{v_j(\boldsymbol{x})\}_{j=1}^J$ be a "suitable" basis set of the function space $V(\Omega)$, where J is the dimension of $V(\Omega)$. Since the functional g is Gaussian and the best knowledge model is linear, we can express the solution of the stochastic PDE (30) as

$$u^*(\boldsymbol{x}) = \sum_{j=1}^J \gamma_j^* v_j(\boldsymbol{x}), \qquad \boldsymbol{\gamma}^* \sim \mathcal{N}(\bar{\boldsymbol{\gamma}}^*, \mathrm{cov}(\boldsymbol{\gamma}^*)). \tag{31}$$

In order to determine $\bar{\boldsymbol{\gamma}}^*$ and $\mathrm{cov}(\boldsymbol{\gamma}^*)$, we choose $v = v_i, i = 1,\dots,J$ in (30) to arrive at the stochastic linear system:

$$\boldsymbol{A}\boldsymbol{\gamma}^* = \boldsymbol{l} - \boldsymbol{g}^*, \tag{32}$$

where $A_{ij} = a(v_i, v_j), l_i = \ell(v_i)$ for $i, j = 1,\dots,J$, and $\boldsymbol{g}^* \sim \mathcal{N}(\bar{\boldsymbol{g}}^*, \mathrm{cov}(\boldsymbol{g}^*))$ with

$$\bar{\boldsymbol{g}}^* = \boldsymbol{K}(\Phi^*, \Phi)\boldsymbol{\beta}, \qquad \mathrm{cov}(\boldsymbol{g}^*) = \boldsymbol{K}(\Phi^*, \Phi^*) - \boldsymbol{K}(\Phi^*, \Phi)\boldsymbol{D}^{-1}\boldsymbol{K}(\Phi, \Phi^*), \tag{33}$$

for $\Phi^* \equiv [v_1, v_2, \dots, v_J]$. It thus follows from (32) that

$$\bar{\boldsymbol{\gamma}}^* = \boldsymbol{A}^{-1}(\boldsymbol{l} - \bar{\boldsymbol{g}}^*), \qquad \mathrm{cov}(\boldsymbol{\gamma}^*) = \boldsymbol{A}^{-1}\mathrm{cov}(\boldsymbol{g}^*)\boldsymbol{A}^{-T}, \tag{34}$$

as $\boldsymbol{g}^*$ is Gaussian and $\boldsymbol{A}$ is invertible.

Now let $\boldsymbol{x}_i \in \Omega, i = 1,\dots,N$ be spatial points at which we would like to evaluate the predictive mean and covariance of u^*. Let $\boldsymbol{V} \in \mathbb{R}^{N\times J}$ be a matrix with entries $V_{ij} = v_j(\boldsymbol{x}_i), i = 1,\dots,N, j = 1,\dots,J$. It then follows from (31) that

$$\boldsymbol{u}^* = \boldsymbol{V}\boldsymbol{\gamma}^*, \tag{35}$$

where $u_i^* = u^*(\boldsymbol{x}_i), i = 1,\dots,N$. It follows from (34) and (35) that

$$\boldsymbol{u}^* \sim \mathcal{N}(\bar{\boldsymbol{u}}^*, \mathrm{cov}(\boldsymbol{u}^*)), \tag{36}$$

```
Input:  $a, \ell, \{c_i\}_{i=1}^M$ (best knowledge), $\{v_j\}_{j=1}^J$(basis functions), $\boldsymbol{d}$ (observed data),
        $\sigma$ (noise level), $k$ (covariance operator), $\{\boldsymbol{x}_i \in \Omega\}_{i=1}^N$ (spatial coordinates)
1.  Compute $\boldsymbol{s}$ by solving the best knowledge model (2)
2.  Compute the adjoint states $\{\phi_i\}_{i=1}^M$ by solving (16)
3.  Compute $(\bar{\boldsymbol{g}}^*, \mathrm{cov}(\boldsymbol{g}^*))$ in (33) by using Gaussian functional regression
4.  Form $\boldsymbol{A}, \boldsymbol{l}, \boldsymbol{V}$ and solve $\boldsymbol{A}^T\boldsymbol{U}^T = \boldsymbol{V}^T$ to obtain $\boldsymbol{U}$
5.  Compute $\bar{\boldsymbol{u}}^* = \boldsymbol{U}(\boldsymbol{l} - \bar{\boldsymbol{g}}^*)$, $\mathrm{cov}(\boldsymbol{u}^*) = \boldsymbol{U}\mathrm{cov}(\boldsymbol{g}^*)\boldsymbol{U}^T$
Return:  $\bar{\boldsymbol{u}}^*$ (posterior mean), $\mathrm{cov}(\boldsymbol{u}^*)$ (posterior covariance)
```

Fig. 1. Main algorithm for computing the posterior distribution of $\boldsymbol{u}^*$.

where

$$\bar{\boldsymbol{u}}^* = \boldsymbol{U}(\boldsymbol{l} - \bar{\boldsymbol{g}}^*), \qquad \mathrm{cov}(\boldsymbol{u}^*) = \boldsymbol{U}\mathrm{cov}(\boldsymbol{g}^*)\boldsymbol{U}^T, \tag{37}$$

with $\boldsymbol{U} = \boldsymbol{V}\boldsymbol{A}^{-1}$. We examine the posterior distribution as given by (37). Note first that the posterior mean $\bar{\boldsymbol{u}}^*$ is the difference between two terms: the first term $\boldsymbol{u} = \boldsymbol{U}\boldsymbol{l}$ is simply the best knowledge state u evaluated at $\boldsymbol{x}_i$, $i = 1, \ldots, N$; the second term $\boldsymbol{U}\bar{\boldsymbol{g}}^*$ is a correction term to the best knowledge state and is obtained by using our Gaussian functional regression. Note also that the posterior covariance is a quadratic form of $\boldsymbol{U}$ with the posterior covariance matrix $\mathrm{cov}(\boldsymbol{g}^*)$, showing that the predictive uncertainty grows with the magnitude of $\boldsymbol{U}$. Hence, the predictive uncertainty depends on the inverse matrix $\boldsymbol{A}^{-1}$. The implementation of our method for computing the posterior distribution (37) is shown in Fig. 1.

4.6. *Relationship with least-squares regression*

Here, we show an alternative approach to computing the posterior mean in (37) by solving a deterministic least-squares problem. We note that the posterior mean $\bar{u}^* \in V$ satisfies

$$a(\bar{u}^*, v) = \ell(v) - \bar{g}^*(v), \quad \forall v \in V. \tag{38}$$

Here, for any $v \in V$, $\bar{g}^*(v)$ is the posterior mean of $g(v)$ and given by

$$\bar{g}^*(v) = \sum_{j=1}^{M} \beta_j k(\phi_j, v), \quad \forall v \in V, \tag{39}$$

where the adjoint states ϕ_j, $j = 1, \ldots, M$ satisfy (16) and the coefficient vector $\boldsymbol{\beta}$ is the solution of (25). It thus follows that the mean state $\bar{u}^* \in V$ satisfies

$$a(\bar{u}^*, v) = \ell(v) - k(\bar{q}^*, v), \quad \forall v \in V, \tag{40}$$

where $\bar{q}^* = \sum_{i=1}^{M} \beta_i \phi_i$ is the weighted sum of the adjoint states. We then evaluate the mean outputs of the stochastic PDE model as

$$\bar{s}_i^* = c_i(\bar{u}^*), \quad i = 1, \ldots, M. \tag{41}$$

The following lemma sheds light on the relationship between the mean outputs and the observed data.

Lemma 1. *Assume that the covariance operator $k(\cdot,\cdot)$ is a bilinear form. We have that $\bar{\boldsymbol{s}}^* = \boldsymbol{d} - \sigma\boldsymbol{\beta}$.*

Proof. We first note from the adjoint equation (16) and (40) that

$$\bar{s}_i^* = c_i(\bar{u}^*) = -a(\bar{u}^*, \phi_i) = \sum_{j=1}^{M} k(\phi_j, \phi_i)\beta_j - \ell(\phi_i), \quad i = 1, \ldots, M. \tag{42}$$

We next recall that $\boldsymbol{\beta}$ satisfies

$$\sum_{j=1}^{M}(k(\phi_j,\phi_i)+\sigma\delta_{ij})\beta_j = d_i - s_i, \quad i=1,\dots,M, \tag{43}$$

where δ_{ij} is the Kronecker delta. Moreover, we obtain from the best knowledge model (2) and the adjoint equation (16) that

$$\ell(\phi_i)=a(u,\phi_i)=-c_i(u)=-s_i, \quad i=1,\dots,M. \tag{44}$$

The desired result immediately follows from the above three equations. This completes the proof. □

This lemma shows that the mean outputs differ from the observed data by the product of the noise level σ and the coefficient vector $\boldsymbol{\beta}$. When the observed data is noise-free (namely, $\sigma=0$) we have that the mean output vector is exactly equal to the observed data. Henceforth, whenever M is sufficiently large and σ is relatively small, we expect that our method will yield a much better estimate of the true state than the best knowledge model. The following theorem shows the optimality of the mean state.

Theorem 2. *Assume that the covariance operator $k(\cdot,\cdot)$ is a bilinear form. Then we have $(\bar{u}^*,\bar{q}^*,\boldsymbol{\beta})=(u^{\mathrm{o}},q^{\mathrm{o}},\boldsymbol{\gamma}^{\mathrm{o}})$, where*

$$\begin{aligned}(u^{\mathrm{o}},q^{\mathrm{o}},\boldsymbol{\gamma}^{\mathrm{o}}) = \arg \min_{z\in V,q\in V,\boldsymbol{\gamma}\in\mathbb{R}^M} & \frac{1}{2}k(q,q)+\frac{1}{2}\sigma^2\boldsymbol{\gamma}^T\boldsymbol{\gamma}\\ \text{s.t. } & a(z,v)+k(q,v)=\ell(v), \quad \forall v\in V,\\ & c_i(z)+\sigma^2\gamma_i=d_i, \quad i=1,\dots,M.\end{aligned} \tag{45}$$

Proof. We introduce the Lagrangian

$$\mathcal{L}(q,z,\boldsymbol{\gamma},p,\boldsymbol{\varrho}) = \frac{1}{2}k(q,q)+\frac{1}{2}\sigma^2\boldsymbol{\gamma}^T\boldsymbol{\gamma}-a(z,p)-k(q,p)+\ell(p)-\sum_{i=1}^{M}\varrho_i(c_i(z)+\sigma^2\gamma_i-d_i), \tag{46}$$

where $p\in V$ and $\boldsymbol{\varrho}\in\mathbb{R}^M$ are the Lagrange multipliers of the constraints. The optimal solution $(q^{\mathrm{o}},u^{\mathrm{o}},\boldsymbol{\gamma}^{\mathrm{o}},p^{\mathrm{o}},\boldsymbol{\varrho}^{\mathrm{o}})$ satisfies

$$\frac{\partial\mathcal{L}}{\partial q}=\frac{\partial\mathcal{L}}{\partial z}=\frac{\partial\mathcal{L}}{\partial\boldsymbol{\gamma}}=\frac{\partial\mathcal{L}}{\partial p}=\frac{\partial\mathcal{L}}{\partial\boldsymbol{\varrho}}=0, \tag{47}$$

which yields

$$k(q^{\mathrm{o}},v)-k(p^{\mathrm{o}},v)=0, \quad \forall v\in V, \tag{48a}$$

$$a(v,p^{\mathrm{o}})+\sum_{i=1}^{M}\varrho_i^{\mathrm{o}}c_i(v)=0, \quad \forall v\in V, \tag{48b}$$

$$\gamma_i^{\mathrm{o}}-\varrho_i^{\mathrm{o}}=0, \quad i=1,\dots,M, \tag{48c}$$

$$a(u^{\mathrm{o}},v)+k(q^{\mathrm{o}},v)-\ell(v)=0, \quad \forall v\in V, \tag{48d}$$

$$c_i(u^{\mathrm{o}})+\sigma^2\gamma_i^{\mathrm{o}}=d_i, \quad i=1,\dots,M. \tag{48e}$$

Note that when taking the partial derivatives we have used the assumption that k is bilinear. The first two equations (48a) and (48b) yield that

$$q^{\mathrm{o}}=p^{\mathrm{o}}=\sum_{i=1}^{M}\varrho_i^{\mathrm{o}}\phi_i, \tag{49}$$

where $\phi_i, i=1,\dots,M$ are the adjoint states. Therefore, if we can show that $\boldsymbol{\varrho}^{\mathrm{o}}=\boldsymbol{\beta}$ then (48c), (48d) and (49) imply that $(\bar{u}^*,\bar{q}^*,\boldsymbol{\beta})=(u^{\mathrm{o}},q^{\mathrm{o}},\boldsymbol{\gamma}^{\mathrm{o}})$. To this end, we note from (48d) and the adjoint equation (16) that

$$c_i(u^{\mathrm{o}})=-a(u^{\mathrm{o}},\phi_i)=k(q^{\mathrm{o}},\phi_i)-\ell(\phi_i), \quad i=1,\dots,M. \tag{50}$$

Moreover, we obtain from the best knowledge model (2) and the adjoint equation (16) that

$$\ell(\phi_i) = a(u, \phi_i) = -c_i(u) = -s_i, \quad i = 1, \dots, M. \tag{51}$$

Finally, it follows from (48c), (48e), (49), (50), and (51) that

$$\sum_{j=1}^{m} \left(k(\phi_i, \phi_j) + \sigma^2 \delta_{ij} \right) \varrho_j^{\mathrm{o}} = d_i - s_i, \quad i = 1, \dots, M, \tag{52}$$

which implies that $\boldsymbol{\varrho}^{\mathrm{o}} = \boldsymbol{\beta}$. This completes the proof. □

This theorem establishes a connection between Gaussian functional regression and traditional least-squares regression when the covariance operator is bilinear. In particular, the posterior mean state $\bar{u}^*$ is the optimal solution of a least-squares minimization. This is hardly a surprise as the posterior mean state is also the maximum a posteriori (MAP) estimate of linear functional regression model in the Bayesian framework discussed in the Appendix. It is well known that the MAP estimate coincides with the least-squares solution. The main advantage of our approach over least-squares regression is that we can compute not only the posterior mean state but also the posterior covariance. Another advantage of our approach is that it allows us to choose a covariance operator based on the observed data by exploiting the marginal likelihood function, whereas least-squares regression does not provide a mechanism to optimally set the prior covariance operator.

5. A simple heat conduction example

5.1. Problem description

For the true PDE model we consider a one-dimensional heat equation:

$$-\frac{\partial^2 u^{\mathrm{true}}}{\partial x^2} = f^{\mathrm{true}}, \quad \text{in } \Omega^{\mathrm{true}} \equiv (-1, 1), \tag{53}$$

with Dirichlet boundary conditions $u^{\mathrm{true}}(-1) = u^{\mathrm{true}}(1) = 0$. The function space $V^{\mathrm{true}}(\Omega^{\mathrm{true}})$ is then given by

$$V^{\mathrm{true}}(\Omega^{\mathrm{true}}) = \left\{ v \; : \; \int_{\Omega^{\mathrm{true}}} \left(v^2 + \frac{\partial v}{\partial x} \frac{\partial v}{\partial x} \right) dx < \infty \text{ and } v(-1) = v(1) = 0 \right\}. \tag{54}$$

The true state $u^{\mathrm{true}} \in V^{\mathrm{true}}(\Omega^{\mathrm{true}})$ satisfies

$$a^{\mathrm{true}}(u^{\mathrm{true}}, v) = \ell^{\mathrm{true}}(v), \quad \forall v \in V^{\mathrm{true}}(\Omega^{\mathrm{true}}), \tag{55}$$

where

$$a^{\mathrm{true}}(w, v) = \int_{\Omega^{\mathrm{true}}} \frac{\partial w}{\partial x} \frac{\partial v}{\partial x} dx, \qquad \ell^{\mathrm{true}}(v) = \int_{\Omega^{\mathrm{true}}} f^{\mathrm{true}} v dx, \quad \forall w, v \in V^{\mathrm{true}}(\Omega^{\mathrm{true}}). \tag{56}$$

We prescribe a *synthetic source term* as $f^{\mathrm{true}} = \sin(\pi x) + 4 \sin(4\pi x)$. It is easy to see that

$$u^{\mathrm{true}} = \frac{\sin(\pi x)}{\pi^2} + \frac{\sin(4\pi x)}{4\pi^2}. \tag{57}$$

The true state is not known to us and will serve to assess the performance of our method.

We next assume that we know almost everything about the true model except for the source term f^{true} and the boundary data. We introduce a function space $V(\Omega)$ with $\Omega = (-1, 1)$ as

$$V(\Omega) = \left\{ v : \int_{\Omega} \left(v^2 + \frac{\partial v}{\partial x} \frac{\partial v}{\partial x} \right) dx < \infty \text{ and } v(-1) = b_1, v(1) = b_2 \right\}, \tag{58}$$

where the boundary data b_1 and b_2 will be determined from the observed data. We then define our best knowledge model: find $u \in V(\Omega)$ such that

$$a(u, v) = \ell(v), \quad \forall v \in V(\Omega), \tag{59}$$

where

$$a(w, v) = \int_{\Omega} \frac{\partial w}{\partial x} \frac{\partial v}{\partial x} dx, \qquad \ell(v) = \int_{\Omega} f v dx, \quad \forall w, v \in V(\Omega). \tag{60}$$

In practice, we replace the continuous space $V(\Omega)$ with a discrete counterpart, for this problem, a 2000-element linear finite element space.

5.2. Model specifications

We now specify the observation functionals $c_i(v) = \int_{\Omega} \delta(x_i) v dx = v(x_i), i = 1, \ldots, M$, where $\delta(x)$ is the Dirac delta function and the x_i are the extended Chebyshev nodes [25] in the interval $[-1, 1]$:

$$x_i = -\frac{\cos((2i-1)\pi/(2M))}{\cos(\pi/(2M))}, \quad i = 1, \ldots, M. \tag{61}$$

These functionals correspond to pointwise observations taken at the points x_i. Note that the set of measurement points $\{x_i\}_{i=1}^M$ varies with M. Note also that the finite element mesh is designed to include $\{x_i\}_{i=1}^M$ in its grid points.

We shall assume that the observations are noise-free, that is we have $\sigma = 0$ and $d_i = u^{\text{true}}(x_i), i = 1, \ldots, M$. Since the observations at x_1 and x_M are used to define the function space $V(\Omega)$ in (58) (that is we set $b_1 = d_1 = 0$ and $b_2 = d_M = 0$), we can only use the remaining $(M-2)$ observations to construct the training set as

$$T = \{(\phi_i, d_i - s_i), \ i = 2, \ldots, M-1\}, \tag{62}$$

where $\phi_i \in V(\Omega), i = 2, \ldots, M-1$ satisfies

$$a(v, \phi_i) = -c_i(v), \quad \forall v \in V(\Omega), \tag{63}$$

and $s_i = u(x_i), i = 2, \ldots, M-1$ are the outputs of the best knowledge model. Hence, the training set has only $(M-2)$ samples. Furthermore, we use a bilinear covariance operator of the form

$$k(w, v; \boldsymbol{\theta}) = \int_{\Omega} \left(\theta_1 w v + \theta_2 \frac{\partial w}{\partial x} \frac{\partial v}{\partial x} \right) dx. \tag{64}$$

The parameters $\boldsymbol{\theta} = (\theta_1, \theta_2)$ are determined by maximizing the log marginal likelihood (28).

We will compare our method to the standard Gaussian process regression described in Section 3 which ignores the best knowledge model and utilizes only the data. To this end, we employ a squared-exponential covariance function of the form

$$\kappa(x, x'; \boldsymbol{\zeta}) = \zeta_1^2 \exp\left(-\frac{1}{2\zeta_2^2} (x - x')^2 \right), \tag{65}$$

where ζ_1 represents the signal variance, while ζ_2 represents the length scale. These parameters are set by maximizing the log marginal likelihood (12).

5.3. Results and discussions

We consider $f = 4\sin(4\pi x)$ for the best knowledge model (59)–(60). This yields the best knowledge state $u = \sin(4\pi x)/(4\pi^2)$. Fig. 2 shows the true state u^{true} and the best knowledge state u. We observe that u is considerably different from u^{true}. Therefore, the best knowledge model does not produce a good prediction of the heat equation (53). We now apply Gaussian functional regression to this example and present numerical results to demonstrate the performance of our method relative to the standard GP regression.

We present in Table 2 the optimal hyperparameters, the $L^2(\Omega)$ norm of the prediction error, and the $L^2(\Omega)$ norm of the posterior standard deviation (the square root of the posterior variance) for our method and the standard GP regression. Here the $L^2(\Omega)$ norm of a function v is defined as $\|v\|_{\Omega} = (\int_{\Omega} v^2 dx)^{1/2}$. We observe that while θ_2 is always zero, θ_1 decreases as M increases, indicating that the prediction uncertainty is reduced as the number of observations increases. We also note that the length scale ζ_2 of the squared-exponential covariance function is

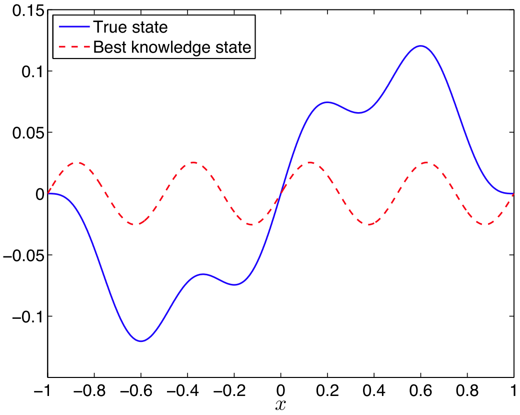


Fig. 2. Plots of the true state and the best knowledge state.

Table 2

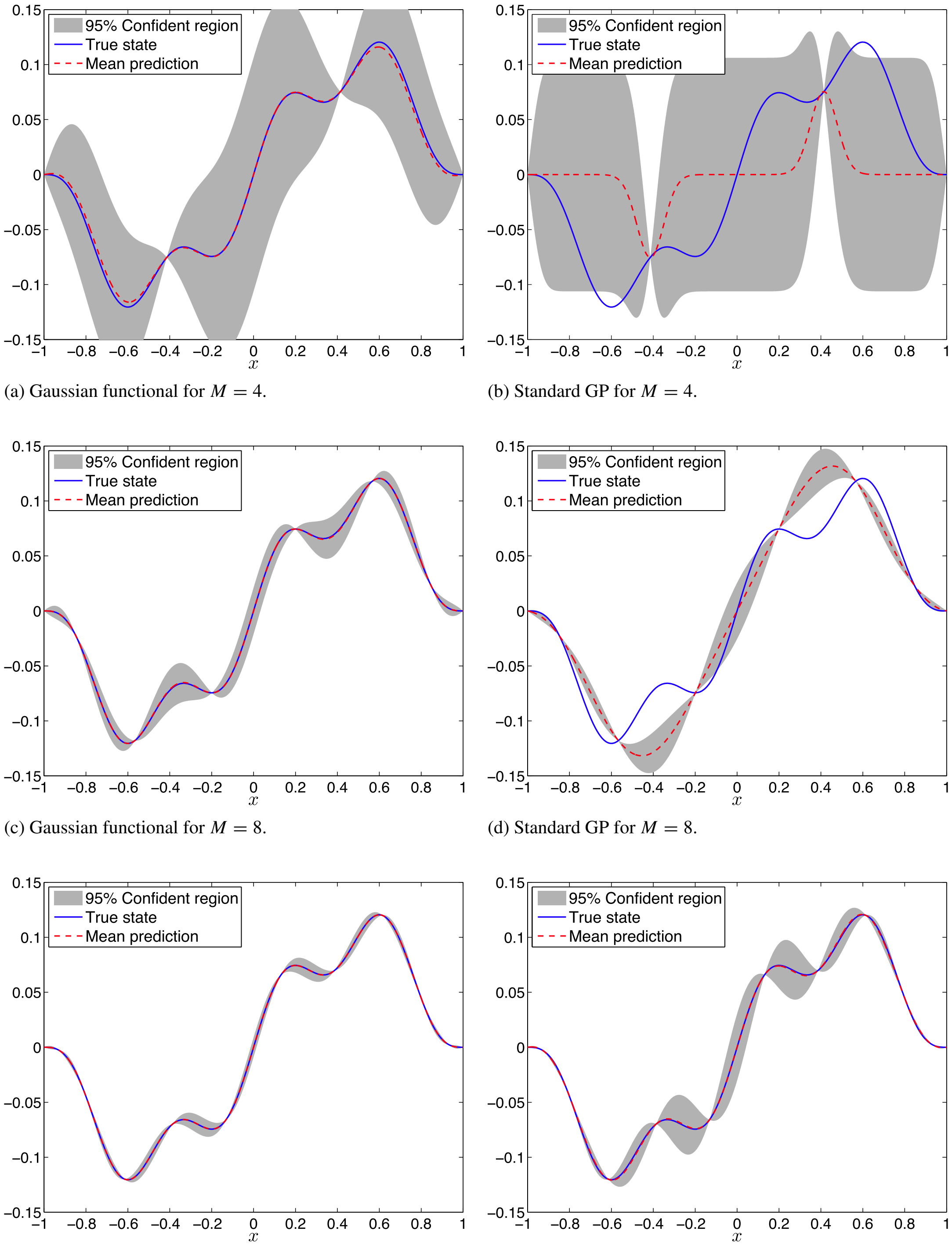
The optimal hyperparameters, the $L^2(\Omega)$ norm of the prediction error ($u^{\rm true}(x) - \bar{u}^*(x)$ in our method and $u^{\rm true}(x) - \bar{h}^*(x)$ in the standard GP regression), and the $L^2(\Omega)$ norm of the standard deviation function ($\sqrt{\mathrm{var}(u^*(x))}$ in our method and $\sqrt{\mathrm{var}(h^*(x))}$ in the standard GP regression) as a function of M for both Gaussian functional regression and standard GP regression. Here $\bar{h}^*(x)$ and $\mathrm{var}(h^*(x))$ are the mean prediction and the posterior variance of $u^{\rm true}(x)$ for the standard GP regression.

M	Gaussian functional regression			Standard GP regression		
	(θ_1, θ_2)	$\Vert u^{\rm true} - \bar{u}^*\Vert_\Omega$	$\Vert\sqrt{\mathrm{var}(u^*)}\Vert_\Omega$	(ζ_1, ζ_2)	$\Vert u^{\rm true} - \bar{h}^*\Vert_\Omega$	$\Vert\sqrt{\mathrm{var}(h^*)}\Vert_\Omega$
4	(0.485, 0)	4.43E−3	4.64E−2	(0.053, 0.064)	9.14E−2	6.84E−2
5	(0.319, 0)	6.02E−3	2.47E−2	(0.076, 0.005)	8.12E−2	9.73E−2
6	(0.247, 0)	1.41E−3	1.60E−2	(0.062, 0.046)	8.68E−2	7.84E−2
7	(0.199, 0)	9.06E−4	1.12E−2	(0.086, 0.621)	3.13E−2	4.40E−4
8	(0.167, 0)	4.16E−4	8.31E−3	(0.074, 0.306)	3.64E−2	8.36E−3
9	(0.143, 0)	2.38E−4	6.42E−3	(0.067, 0.203)	3.25E−2	2.05E−2
10	(0.125, 0)	1.42E−4	5.12E−3	(0.064, 0.186)	9.39E−3	1.85E−2
11	(0.111, 0)	9.10E−5	4.18E−3	(0.062, 0.223)	1.90E−2	7.00E−3
12	(0.100, 0)	6.10E−5	3.48E−3	(0.065, 0.193)	9.24E−4	9.43E−3
13	(0.091, 0)	4.25E−5	2.94E−3	(0.061, 0.196)	4.57E−3	5.96E−3
14	(0.084, 0)	3.06E−5	2.52E−3	(0.064, 0.200)	4.24E−4	3.87E−3
15	(0.077, 0)	2.26E−5	2.18E−3	(0.065, 0.205)	1.58E−3	2.22E−3

relatively small for $M \leq 6$, indicating that the training set may be inadequate for the standard GP regression to produce a good prediction.

We see from Table 2 that the prediction error in our method converges significantly faster than that in the standard GP regression as M increases. Therefore, our method requires fewer observations to achieve the same accuracy. In particular, our method with 4 observations has slightly smaller error than the standard GP regression with 13 observations. This is made possible because our method uses both the best knowledge model and the observations to do regression on the space of functionals, whereas the standard GP progression uses the observations only to do regression on the space of functions. We also observe that the posterior standard deviation (measured in $L^2(\Omega)$ norm) shrinks with increasing M albeit at a slower rate than the prediction error, indicating that our posterior variance of the prediction error is rigorous. In contrast, the standard GP regression has the posterior standard deviation even smaller than the prediction error for small values of M, indicating that the posterior variance of the standard GP regression may not be rigorous when the training set is inadequate. This can be attributed to the fact the standard GP regression requires a large enough set of observations to provide accurate prediction and rigorous error estimation.

Finally, we show in Fig. 3 the true state, the mean prediction, and the 95% confidence region (shaded area) for our method (left panels) and the standard GP regression (right panels). Here the 95% confidence region is an area bounded by the mean prediction plus and minus two times the standard deviation function. Note that the prediction error is zero at the measurement points, which is consistent with the theoretical result stated in Lemma 1. Moreover, the standard



(a) Gaussian functional for $M = 4$.

(b) Standard GP for $M = 4$.

(c) Gaussian functional for $M = 8$.

(d) Standard GP for $M = 8$.

(e) Gaussian functional for $M = 12$.

(f) Standard GP for $M = 12$.

Fig. 3. Panels show the true state and the mean prediction obtained using $M = 4, 8, 12$ observations for both Gaussian functional regression (left) and the standard GP regression (right). In these plots the shaded area represents the mean prediction plus and minus two times the standard deviation function (corresponding to the 95% confidence region).

deviation function is also zero at the measurement points — a consequence of the fact that the prediction error is zero at those points. We see that our method does remarkably well even with just 4 observations when it is compared to the

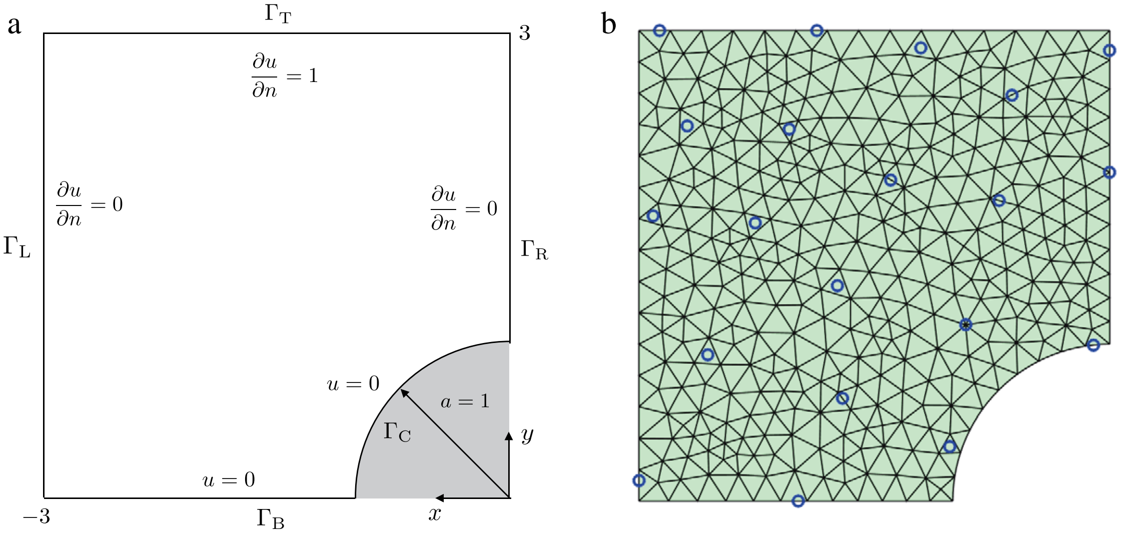


Fig. 4. Potential flow around a circular cylinder: (a) Geometry and boundary conditions and (b) FE mesh and measurement points.

standard GP regression. For our method the true state u^{true} resides in the 95% confidence region which shrinks rapidly as M increases, whereas for the standard GP regression u^{true} does not always reside in the 95% confidence region. Indeed, as seen in Fig. 3(d), the standard GP regression gives poor prediction and erroneous 95% confidence region for $M = 8$. Although the standard GP regression provides more accurate prediction and rigorous 95% confidence region for $M = 12$, it is still not as good as our method. In summary, the numerical results obtained for this simple example show that our method outperforms the standard GP regression.

6. Potential flow around a circular cylinder

6.1. Problem description

We consider a potential flow around a circular cylinder of radius $a = 1$ in an infinite domain. The stream function for this flow is given by

$$u^{\text{true}} = y\left(1 - \frac{1}{x^2 + y^2}\right). \tag{66}$$

This solution is shown in Fig. 5(a). Note that only the upper left quadrant of the domain is considered because of the symmetry.

Since the circular cylinder is a simple geometry the stream function of the potential flow is known analytically. For more complicated geometries, however, the problem is usually solved by using a numerical method. In the case of a finite element method, the infinite domain needs to be truncated to a finite domain. This gives rise to the following PDE model

$$\Delta u = 0, \quad \text{in } \Omega, \tag{67}$$

where the truncated domain Ω and the boundary conditions are shown in Fig. 4(a). The weak formulation is given by

$$a(u, v) = \ell(v), \quad \forall v \in V, \tag{68}$$

where, for all $w, v \in V$,

$$a(w, v) = \int_{\Omega} \nabla w \cdot \nabla v dx dy, \qquad \ell(v) = \int_{\Gamma_{\mathrm{T}}} v dx dy, \tag{69}$$

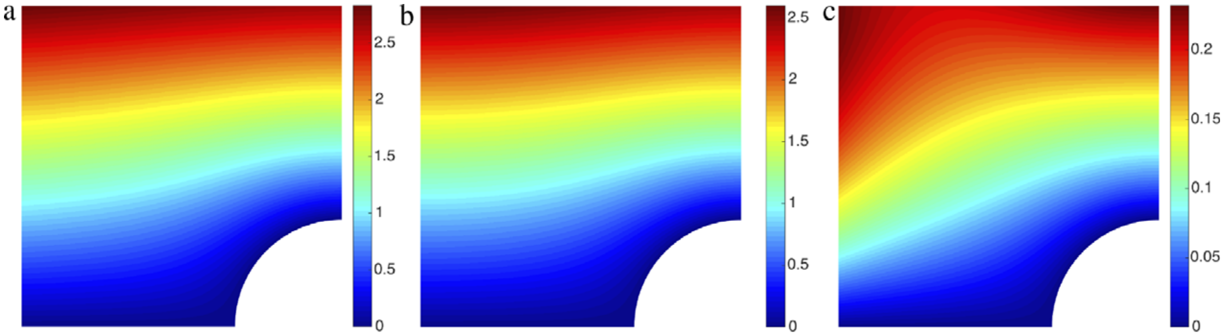


Fig. 5. Plot of the true state u^{true}, the best knowledge state u, and their difference $u^{\text{true}} - u$.

and $V \equiv \{v \in H^1(\Omega) : v = 0 \text{ on } \Gamma_B \cup \Gamma_C\}$. (In practice, we replace V with a FE approximation space of piecewise polynomials of degree $p = 3$ defined on a mesh of 690 elements. This FE discretization is fine enough such that the numerical error is insignificant.)

This example serves to demonstrate our approach when the uncertainty source comes from the domain truncation and the boundary condition. Due to this uncertainty, the best knowledge state u is significantly different from the true state u^{true} as shown in Fig. 5(c). We aim to improve the prediction of the true state u^{true} by combining the best knowledge model with observations of the true state. The observation functionals are specified as

$$c_i(v) = \int_\Omega \exp\left(-\frac{(x - x_i)^2 + (y - y_i)^2}{0.25^2}\right) v dx dy, \quad i = 1, \ldots, M, \tag{70}$$

where $(x_i, y_i), i = 1, \ldots, M$ are selected by using the empirical interpolation method [9,11] and are shown in Fig. 4(b). We shall assume that the observations are noise-free, so that $d_i = c_i(u^{\text{true}}), i = 1, \ldots, M$.

6.2. Results and discussions

We consider a bilinear covariance operator of the form

$$k(w, v; \boldsymbol{\theta}) = \int_\Omega (\theta_1 w v + \theta_2 \nabla w \cdot \nabla v) \, dx dy. \tag{71}$$

The parameters $\boldsymbol{\theta} = (\theta_1, \theta_2)$ are determined by maximizing the log marginal likelihood (28). Fig. 6(a) shows the hyperparameter θ_1 as a function of the number of measurements. It is interesting to see that our hyperparameter selection method yields $\theta_2 = 0$ for any M. Note further that θ_1 tends to decrease as M increases.

We next show in Fig. 6(b) the norm of the standard deviation function $\|\sqrt{\text{var}(u^*)}\|_\Omega$ and the error norm $\|u^{\text{true}} - \bar{u}\|_\Omega$ as a function of the number of measurements. As expected, both the error norm and the standard deviation norm decrease as M increases. Hence, increasing the number of measurements reduces the uncertainty in our prediction of the true state. This can be seen more clearly in Fig. 7 which shows the error function and the standard deviation function for three different values of $M = 5$, 10, and 15. Moreover, we observe that both the error function and the standard deviation function near the top boundary Γ_T and the left boundary Γ_L of the domain are larger than those in the other region of the domain. This can be attributed to the fact that the uncertainty source comes from the domain truncation and the boundary condition on Γ_T and Γ_L.

7. Conclusions

In this paper, we have presented a new statistical approach to the problem of combining the observed data with a mathematical model to improve our prediction of a physical system. A new *Gaussian functional regression* method is presented, which has its root in Gaussian processes for functions. Our approach has the following unique properties. First, it allows for the incorporation of the best knowledge model into the regression procedure. Second, it can handle observations given in the form of linear functionals of the field variable. Third, the method is non-parametric in the sense that it provides a systematic way to optimally determine the prior covariance based on the data. Fourth, our

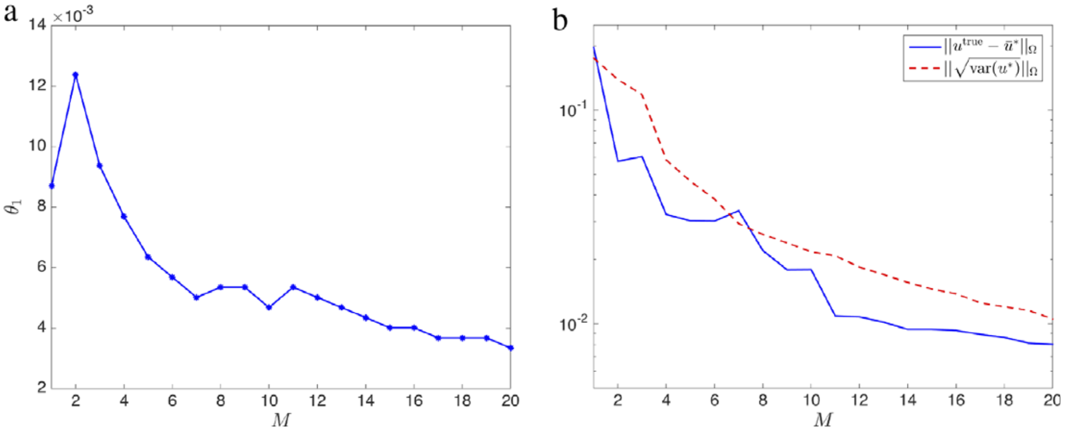


Fig. 6. (a) The hyperparameter θ_1 and (b) the error norm and the standard deviation norm as a function of M. Note that $\theta_2 = 0$ for any M.

method can compute not only the mean prediction but also the posterior covariance, characterizing the uncertainty in our prediction of the physical state. These features distinguish our method from other regression methods. Numerical results were presented to highlight these features and demonstrate the superior performance of the proposed method relative to the standard GP regression.

We conclude the paper by pointing out several possible extensions and directions for further research. We would like to extend the proposed approach to nonlinear PDE models, which will broaden the application domain of our method. Nonlinear PDEs represent some significant challenges because the adjoint problems will depend on the true state which we do not know and because our stochastic PDE does not preserve the Gaussian property of the functional g due to nonlinearity. We would also like to extend the method to goal-oriented statistical estimation in which we would like to infer new outputs rather the state of the physical system.

Acknowledgments

We would like to thank Professor Robert M. Freund of Sloan School of Management at MIT for fruitful discussions. This work was supported by AFOSR Grant No. FA9550-11-1-0141, AFOSR Grant No. FA9550-12-0357, and the Singapore-MIT Alliance.

Appendix. Bayesian linear functional regression

In this section, we develop a nonparametric Bayesian framework for linear functional regression with Gaussian noise. We then show that this framework is equivalent to our Gaussian functional regression method described in Section 3.

We begin by introducing an orthonormal basis set $\{\psi_i \in V\}_{i=1}^J$ such that $\int_\Omega \psi_i \psi_j d\boldsymbol{x} = \delta_{ij}$, where δ_{ij} is the Kronecker delta. We define associated linear functionals $\ell_i : V \to \mathbb{R}$ as

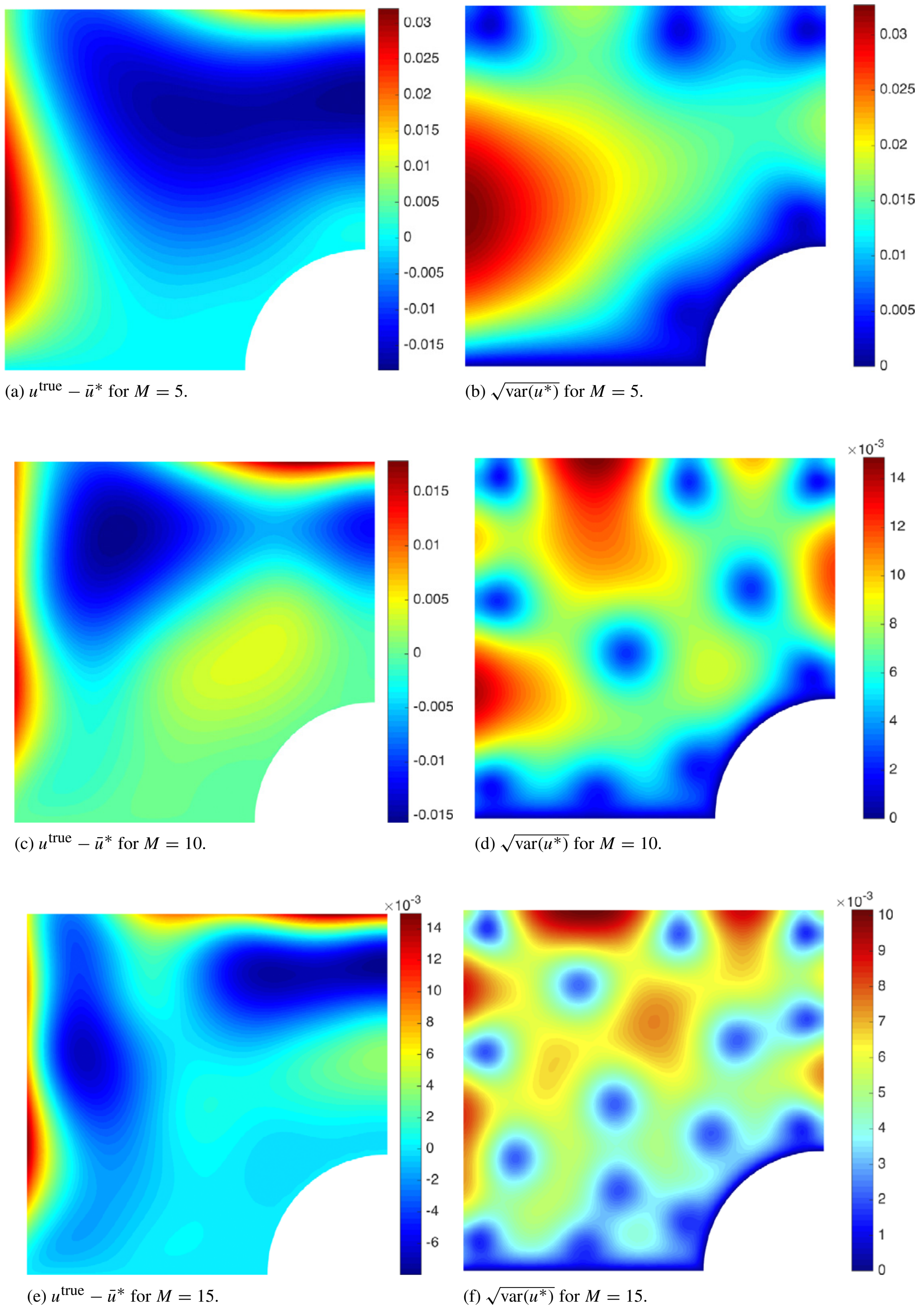
$$\ell_i(v) = \int_\Omega \psi_i v d\boldsymbol{x} \equiv m(\psi_i, v), \quad i = 1, \ldots, J. \tag{72}$$

The linear functional regression model is defined by

$$g(v) = \sum_{i=1}^J w_i \ell_i(v), \quad y = g(v) + \varepsilon, \tag{73}$$

where $v \in V$ is the input function, $\boldsymbol{w} = [w_1, \ldots, w_J]^T$ is a vector of weights (parameters) of the linear model, g is the functional and $y \in \mathbb{R}$ is the observed target value. We assume that the observed value y differs from the functional value $g(v)$ by additive noise

$$\varepsilon \sim \mathcal{N}(0, \sigma^2). \tag{74}$$



(a) $u^{\text{true}} - \bar{u}^*$ for $M = 5$.

(b) $\sqrt{\text{var}(u^*)}$ for $M = 5$.

(c) $u^{\text{true}} - \bar{u}^*$ for $M = 10$.

(d) $\sqrt{\text{var}(u^*)}$ for $M = 10$.

(e) $u^{\text{true}} - \bar{u}^*$ for $M = 15$.

(f) $\sqrt{\text{var}(u^*)}$ for $M = 15$.

Fig. 7. Panels show the error function (left) and the standard deviation function (right) obtained using $M = 5, 10, 15$ observations.

This noise assumption together with the model directly gives rise to the following likelihood

$$p(\boldsymbol{y}|\Phi, \boldsymbol{w}) = \prod_{j=1}^{M} \frac{1}{\sqrt{2\pi}\sigma} \exp\left(\frac{-(y_j - g(\phi_j))^2}{2\sigma^2}\right) = \mathcal{N}(\boldsymbol{L}^T \boldsymbol{w}, \sigma^2), \tag{75}$$

where $\boldsymbol{L}$ has entries $L_{ij} = \ell_i(\phi_j), i = 1, \dots, J, j = 1, \dots, M$. Recall that $\phi_i, i = 1, \dots, M$ are the adjoint states and that $y_i = d_i - s_i, i = 1, \dots, M$ are the observed outputs which are the differences between the observed data and the best knowledge outputs.

In the Bayesian formalism we need to specify *a prior distribution* for $\boldsymbol{w}$, which encodes our belief about the weights prior to taking into account the observations. We consider a zero-mean Gaussian prior distribution with covariance matrix $\boldsymbol{\Lambda}$ for the weights

$$p(\boldsymbol{w}) = \mathcal{N}(0, \boldsymbol{\Lambda}). \tag{76}$$

Without loss of generality we assume that $\boldsymbol{\Lambda}$ is a diagonal matrix, that is, $\Lambda_{ij} = 0$ if $i \neq j$. If $\boldsymbol{\Lambda}$ is not a diagonal matrix then we can always introduce a new weight vector $\boldsymbol{w}' = \boldsymbol{\Lambda}^{-1/2}\boldsymbol{w}$, so that the covariance of $\boldsymbol{w}'$ is a diagonal matrix. Then we choose to work with $\boldsymbol{w}'$ instead of $\boldsymbol{w}$. The posterior distribution over the weights is then obtained by using Bayes' rule as

$$p(\boldsymbol{w}|\boldsymbol{y}, \Phi) = \frac{p(\boldsymbol{y}|\Phi, \boldsymbol{w})p(\boldsymbol{w})}{p(\boldsymbol{y}|\Phi)}, \tag{77}$$

where the normalizing constant is given by

$$p(\boldsymbol{y}|\Phi) = \int p(\boldsymbol{y}|\Phi, \boldsymbol{w})p(\boldsymbol{w})d\boldsymbol{w}. \tag{78}$$

Substituting (75) and (76) into (77) and working through some algebraic manipulations, we obtain

$$p(\boldsymbol{w}|\boldsymbol{y}, \Phi) \sim \mathcal{N}\left(\frac{1}{\sigma^2}\boldsymbol{B}^{-1}\boldsymbol{L}\boldsymbol{y}, \boldsymbol{B}^{-1}\right), \tag{79}$$

where $\boldsymbol{B} = \sigma^{-2}\boldsymbol{L}\boldsymbol{L}^T + \boldsymbol{\Lambda}^{-1}$. The posterior distribution (79) combines the likelihood and the prior distribution, and captures all information about the weights.

To make predictions for a test function $\phi^* \in V$ we average the linear functional regression model (73) over all possible values of the weights under their posterior distribution (79). Hence, the predictive distribution for $g^* = g(\phi^*)$ at ϕ^* is given by

$$g^*|\phi^*, \Phi, \boldsymbol{y} \sim \mathcal{N}\left(\frac{1}{\sigma^2}\boldsymbol{l}(\phi^*)^T\boldsymbol{B}^{-1}\boldsymbol{L}\boldsymbol{y}, \boldsymbol{l}(\phi^*)^T\boldsymbol{B}^{-1}\boldsymbol{l}(\phi^*)\right), \tag{80}$$

where $\boldsymbol{l}(\phi^*)$ has entries $l_i(\phi^*) = \ell_i(\phi^*), i = 1, \dots, J$. The predictive distribution is a Gaussian distribution as expected. However, the predictive distribution (80) requires the matrix inversion $\boldsymbol{B}^{-1}$ which may be expensive. Fortunately, we can derive an equivalent distribution which is more efficient to compute than the original one (80).

We recall the Woodbury, Sherman and Morrison (WSM) formula for the matrix inversion

$$(\boldsymbol{Z} + \boldsymbol{U}\boldsymbol{W}\boldsymbol{V}^T)^{-1} = \boldsymbol{Z}^{-1} - \boldsymbol{Z}^{-1}\boldsymbol{U}(\boldsymbol{W}^{-1} + \boldsymbol{V}^T\boldsymbol{Z}^{-1}\boldsymbol{U})^{-1}\boldsymbol{V}^T\boldsymbol{Z}^{-1}. \tag{81}$$

Using the WSM formula with $\boldsymbol{Z} = \boldsymbol{\Lambda}$, $\boldsymbol{W} = \sigma^{-2}\boldsymbol{I}$, $\boldsymbol{U} = \boldsymbol{V} = \boldsymbol{L}$ we obtain

$$\boldsymbol{B}^{-1} = (\sigma^{-2}\boldsymbol{L}\boldsymbol{L}^T + \boldsymbol{\Lambda}^{-1})^{-1} = \boldsymbol{\Lambda} - \boldsymbol{\Lambda}\boldsymbol{L}(\sigma^2\boldsymbol{I} + \boldsymbol{L}^T\boldsymbol{\Lambda}\boldsymbol{L})^{-1}\boldsymbol{L}^T\boldsymbol{\Lambda}. \tag{82}$$

It thus follows that

$$\begin{aligned} \boldsymbol{B}^{-1}\boldsymbol{L} &= \boldsymbol{\Lambda}\boldsymbol{L} - \boldsymbol{\Lambda}\boldsymbol{L}(\sigma^2\boldsymbol{I} + \boldsymbol{L}^T\boldsymbol{\Lambda}\boldsymbol{L})^{-1}\boldsymbol{L}^T\boldsymbol{\Lambda}\boldsymbol{L} \\ &= \boldsymbol{\Lambda}\boldsymbol{L}(\sigma^2\boldsymbol{I} + \boldsymbol{L}^T\boldsymbol{\Lambda}\boldsymbol{L})^{-1}((\sigma^2\boldsymbol{I} + \boldsymbol{L}^T\boldsymbol{\Lambda}\boldsymbol{L}) - \boldsymbol{L}^T\boldsymbol{\Lambda}\boldsymbol{L}) \\ &= \sigma^2\boldsymbol{\Lambda}\boldsymbol{L}(\sigma^2\boldsymbol{I} + \boldsymbol{L}^T\boldsymbol{\Lambda}\boldsymbol{L})^{-1}. \end{aligned} \tag{83}$$

Inserting (82) and (83) into (80) we get

$$g^*|\phi^*, \Phi, \boldsymbol{y} \sim \mathcal{N}(\bar{g}^*, \mathrm{var}(g^*)), \tag{84}$$

where

$$\begin{aligned} \bar{g}^* &= \boldsymbol{l}(\phi^*)^T \boldsymbol{\Lambda} \boldsymbol{L}(\sigma^2 \boldsymbol{I} + \boldsymbol{L}^T \boldsymbol{\Lambda} \boldsymbol{L})^{-1} \boldsymbol{y}, \\ \mathrm{var}(g^*) &= \boldsymbol{l}(\phi^*)^T \boldsymbol{\Lambda} \boldsymbol{l}(\phi^*) - \boldsymbol{l}(\phi^*)^T \boldsymbol{\Lambda} \boldsymbol{L}(\sigma^2 \boldsymbol{I} + \boldsymbol{L}^T \boldsymbol{\Lambda} \boldsymbol{L})^{-1} \boldsymbol{L}^T \boldsymbol{\Lambda} \boldsymbol{l}(\phi^*). \end{aligned} \tag{85}$$

In the predictive distribution (84), we need to invert the matrix $(\sigma^2 \boldsymbol{I} + \boldsymbol{L}^T \boldsymbol{\Lambda} \boldsymbol{L})$ of size $M \times M$. Hence, it is more attractive than the original distribution (80).

Thus far, we have not mentioned how we construct the basis functions $\{\psi_i\}_{i=1}^J$ and the covariance matrix $\boldsymbol{\Lambda}$. We observe that in the predictive distribution (84) we need to compute $\boldsymbol{L}^T \boldsymbol{\Lambda} \boldsymbol{L}, \boldsymbol{l}(\phi^*)^T \boldsymbol{\Lambda} \boldsymbol{L}, \boldsymbol{l}(\phi^*)^T \boldsymbol{\Lambda} \boldsymbol{l}(\phi^*)$, where both $\boldsymbol{L}$ and $\boldsymbol{l}(\phi^*)$ depend on the basis set $\{\psi_i\}_{i=1}^J$ through the definition of the linear functionals $\ell_i(\cdot), i = 1, \dots, J$. The entries of these quantities have the form $\boldsymbol{l}(v)^T \boldsymbol{\Lambda} \boldsymbol{l}(v')$ for $v, v' \in V$. We can thus define

$$k(v', v) = \boldsymbol{l}(v')^T \boldsymbol{\Lambda} \boldsymbol{l}(v), \quad \text{for all } v, v' \in V. \tag{86}$$

It follows that

$$k(v', v) = \sum_{i=1}^J \sum_{j=1}^J \ell_i(v') \Lambda_{ij} \ell_j(v) = \sum_{i=1}^J m(\psi_i, v') \Lambda_{ii} m(\psi_i, v), \quad \forall v', v \in V, \tag{87}$$

since $\ell_i(v) = m(\psi_i, v)$ and $\boldsymbol{\Lambda}$ is a diagonal matrix. We next choose $v' = \psi_n$ and invoke $m(\psi_i, \psi_n) = \delta_{in}$ to arrive at the following eigenvalue problem:

$$k(\psi_n, v) = \Lambda_{nn} m(\psi_n, v), \quad \forall v \in V. \tag{88}$$

This equation shows that $\{\psi_n\}_{n=1}^J$ and $\{\Lambda_{nn}\}_{n=1}^J$ are the eigenfunctions and eigenvalues of the covariance operator $k(\cdot, \cdot)$. Once the covariance operator $k(\cdot, \cdot)$ is specified, we can construct the basis set $\{\psi_n\}_{n=1}^J$ and the covariance matrix $\boldsymbol{\Lambda}$ by solving the eigenvalue problem (88).[6] We can then make predictions using Eqs. (84) and (85). The predictive distribution (84) is nothing but the same as that of Gaussian functional regression described in Section 3. However, unlike nonparametric Bayesian inference, Gaussian functional regression does not require the computation of eigenfunctions and eigenvalues. This is because Gaussian functional regression needs only to compute the inner products $k(v, v')$ for some pairs $v, v' \in V$, which is known as the *kernel trick* [22].

References

[1] M. Yano, J.D. Penn, A.T. Patera, A model-data weak formulation for simultaneous estimation of state and model bias, C. R. Math. 351 (23–24) (2013) 937–941.

[2] Albert Tarantola, Inverse Problem Theory and Methods for Model Parameter Estimation, SIAM, 2005.

[3] Z. Zhang, Parameter estimation techniques: a tutorial with application to conic fitting, Image Vis. Comput. 15 (1) (1997) 59–76.

[4] R.E. Kalman, A new approach to linear filtering and prediction problems, Trans. ASME D 82 (1) (1960) 35–45.

[5] J.M. Lewis, S. Lakshmivarahan, S. Dhall, Dynamic Data Assimilation: A Least Squares Approach, Cambridge University Press, 2006.

[6] Z. Li, I.M. Navon, Optimality of variational data assimilation and its relationship with the Kalman filter and smoother, Q. J. R. Meteorol. Soc. 127 (572) (2001) 661–683.

[7] A.C. Lorenc, A global three-dimensional multivariate statistical interpolation scheme, Mon. Weather Rev. 109 (4) (1981) 701–721.

[8] T. Sondergaard, P.F.J. Lermusiaux, Data assimilation with Gaussian mixture models using the dynamically orthogonal field equations. Part I: theory and scheme, Mon. Weather Rev. 141 (6) (2013) 1737–1760.

[9] M. Barrault, Y. Maday, N.C. Nguyen, A.T. Patera, An 'empirical interpolation' method: application to efficient reduced-basis discretization of partial differential equations, C. R. Math. 339 (9) (2004) 667–672.

[10] R. Everson, L. Sirovich, Karhunen–Loeve procedure for gappy data, Opt. Soc. Amer. A 12 (8) (1995) 1657–1664.

[6] Of course, we can also define the covariance operator $k(v', v)$ in terms of the basis set $\{\psi_n\}_{n=1}^J$ and the covariance matrix $\boldsymbol{\Lambda}$ by using the expression (87). However, this series expansion of the covariance operator render the inner products too computationally expensive and should be avoided. Instead, one should work directly with an explicitly analytical form of the covariance operator.

[11] Y. Maday, N.C. Nguyen, A.T. Patera, G.S.H. Pau, A general multipurpose interpolation procedure: the magic points, Comm. Pure Appl. Math. 8 (1) (2008) 383–404.

[12] N.C. Nguyen, A.T. Patera, J. Peraire, A best points interpolation method for efficient approximation of parametrized functions, Internat. J. Numer. Methods Engrg. 73 (4) (2008) 521–543.

[13] A.T. Patera, E.M. Ronquist, Regression on parametric manifolds: Estimation of spatial fields, functional outputs, and parameters from noisy data, C. R. Math. 350 (9–10) (2012) 543–547.

[14] K. Willcox, Unsteady flow sensing and estimation via the gappy proper orthogonal decomposition, Comput. & Fluids 35 (2006) 208–226.

[15] A. Cohen, M.A. Davenport, D. Leviatan, On the stability and accuracy of least squares approximations, Found. Comput. Math. 13 (5) (2013) 819–834.

[16] J. Wolberg, Data Analysis Using the Method of Least Squares: Extracting the Most Information from Experiments, 2006 ed., Springer, 2005.

[17] T.A. El Moselhy, Y.M. Marzouk, Bayesian inference with optimal maps, J. Comput. Phys. 231 (23) (2012) 7815–7850.

[18] A. Gelman, J. Carlin, H. Stern, D. Dunson, A. Vehtari, D. Rubin, Bayesian Data Analysis, third ed., Chapman and Hall/CRC, 2013.

[19] Simo Särkkä, Bayesian Filtering and Smoothing, Cambridge University Press, 2013.

[20] C.K. Wikle, L.M. Berliner, A Bayesian tutorial for data assimilation, Physica D 230 (1–2) (2007) 1–16.

[21] A. O'Hagan, J.F.C. Kingman, Curve fitting and optimal design for prediction, J. R. Stat. Soc. Ser. B Stat. Methodol. 40 (1) (1978) 1–42.

[22] C.E. Rasmussen, C.K.I. Williams, Gaussian Processes for Machine Learning, MIT Press, 2006.

[23] J. Sacks, W.J. Welch, T.J. Mitchell, H.P. Wynn, Design and analysis of computer experiments, Statist. Sci. 4 (4) (1989) 409–423.

[24] N. Aronszajn, Theory of reproducing kernels, 1950.

[25] L. Brutman, On the Lebesgue function for polynomial interpolation, SIAM J. Numer. Anal. 15 (4) (1978) 694–704.