Mechanical Systems and Signal Processing 168 (2022) 108717

Contents lists available at ScienceDirect

Mechanical Systems and Signal Processing

journal homepage: www.elsevier.com/locate/ymssp

Robust equation discovery considering model discrepancy: A sparse Bayesian and Gaussian process approach

Yi-Chen Zhu [a,*], Paul Gardner [b], David J. Wagg [b], Robert J. Barthorpe [b], Elizabeth J. Cross [b], Ramon Fuentes [b]

[a] Department of Bridge Engineering, School of Transportation, Southeast University, Nanjing, China
[b] Dynamics Research Group, Department of Mechanical Engineering, The University of Sheffield, UK

ARTICLE INFO

Keywords:
Parameter identification
Sparse Bayesian inference
Gaussian process
Model discrepancy
Equation discovery

ABSTRACT

Computational models are widely used to describe engineering systems and to predict their behaviour. However, in many applications, these computational models do not capture the complete physics of the real system, leading to model discrepancy. Model discrepancy causes bias in the inferred model parameters when it is not properly accounted for. This paper proposes a novel approach that seeks to capture the functional form of the model discrepancy and reduce bias in the estimated model parameters through and equation discovery procedure. A sparse Bayesian model is proposed, where sparsity is introduced through a hierarchical prior structure, providing a mechanism for removing erroneous candidate model terms from a series of potential equations as a part of an equation discovery procedure. At the same time, a Gaussian process model is used to account for model discrepancy. These two modelling assumptions are combined in a Bayesian formulation, allowing the system parameters and model discrepancy to be inferred in a probabilistic manner with associated uncertainties quantified based on their posterior distributions. The resulting method is capable of simultaneously providing physical insights into the system behaviour, by selecting the appropriate candidate model components and their respective system parameters, whilst compensating for model discrepancy that may occur due to an incomplete set of candidate terms. In order to efficiently solve the statistical model, an expectation maximisation algorithm is proposed for performing inference, and illustrative examples are presented to validate the proposed method. It is shown that compared to using a conventional sparse Bayesian approach for performing equation discovery, such as the Relevance Vector Machine, the proposed method provides better equation selection and parameter estimation, with less bias in the parameter estimates.

1. Introduction

Physics-based computer models are widely used to describe engineering systems and make predictions of their responses. System identification [1] is concerned with identifying the model forms and extracting associated model parameters based on the measured system response, which can then be used for further prediction and control of the system behaviour. System identification can normally be divided into two parts, *model selection* [2,3] and *model updating* [4,5]. The first part focuses on determining the functional forms or

* Corresponding author.
E-mail address: zhuyichen_uol@hotmail.co.uk (Y.-C. Zhu).

https://doi.org/10.1016/j.ymssp.2021.108717
Received 16 November 2020; Received in revised form 23 September 2021; Accepted 30 November 2021
Available online 13 December 2021

structures of the model while the second part focuses on estimating the parameters associated with the model. Equation discovery has been proposed as a method for recovering system forms and associated system parameters from observed system data [6,7]. Equation discovery as a concept is seen as distinct from parameter estimation or model selection, even though there are some similarities. For example, model selection typically focuses on selecting one model from a set of candidate models (usually probabilistically) that best predicts the observed response data, whilst equation discovery seeks to select model components to construct the overall model from a design matrix (which is a set of candidate model components), that together best predicts the observed system response. In addition, whilst selecting the most appropriate candidate model components, equation discovery approaches also estimate their corresponding system parameter values. This is different from model updating which mainly focuses on estimation of system parameters with a fixed model form.

However, as stated by Box [8], 'all models are wrong, but some are useful', with models being imperfect reflections of reality due to modelling simplifications, missing physics, numerical approximations etc. Formally, the mismatch between the output of a computer model (when the true model parameters are known) and the observed system response is known as model discrepancy, and exists in all computer models to varying degrees [9]. Model calibration based on measured system response data is often required in order to ensure that the model is an accurate representation of the real system [10,11]. Without a mechanism for properly accounting for model discrepancy during calibration, the inferred parameter estimates will be biased, as first considered by Kennedy and O'Hagan [9].

State-of-the-art equation discovery methods [6,7] assume that the 'true' set of model components are included in the design matrix (which is an over-defined set), such that their corresponding 'true' parameter values can be obtained. However, in practice the set of candidate model components is likely to be incomplete, as it is difficult to predict and include all possible physics in the design matrix. The effect of missing some 'true' model components from the design matrix leads to model discrepancy in a similar manner to model calibration methods (as stated previously). When the design matrix does not include all the necessary correct model components that would be found in the real-world system, this model discrepancy leads the equation discovery procedure to select erroneous ones, as well as inferring biased parameter estimates as the method is compensating for the missing physics. This paper focuses on the problem of performing equation discovery in the presence of model discrepancy, overcoming issues faced in equation discovery when a design matrix does not contain *all* the correct physics.

The problem of model discrepancy within the calibration process has also been considered in the literature [9,12–15]. The idea of modelling discrepancy via a non-parametric regressor was first introduced by Kennedy and O'Hagan [9]. Several other studies have followed this approach [12–15] using Gaussian Process (GP) regression. However, this approach to accounting for model discrepancy, where the system parameters are inferred jointly with a GP model that captures the model discrepancy, has a fundamental problem [16] where the flexibility of the GP model can lead to identifiability issues [17]. Two solutions of this problem has been proposed: using more informative prior distributions for the system parameters and the GP model, or using physical constraints on the GP model [16,18,19]. Alternative approaches to the joint inference procedure have also been proposed, such as Bayesian history matching [20], which seek to overcome the non-identifiability problem by decoupling the parameter and model discrepancy inference. However, none of these techniques seek to identify and select the best set of system model components; capturing any model discrepancy that may occur and inferring any system parameters at the same time.

Motivated by the above concerns, this paper tries to account for model discrepancy during an equation discovery procedure, i.e., removing erroneous model components (that are not physically representative of the real system) whilst accounting for model discrepancy and reducing any parameter bias simultaneously. The proposed method in this paper builds upon work using Relevance Vector Machines (RVM) [21], a form of sparse Bayesian inference, that were first suggested for performing equation discovery in [7]. The novel contribution of the proposed approach in this paper is to extend this method to include a GP model [17] in order to account for model discrepancy. The idea is that the flexible GP model will be able to account for the behaviour of a system that is not covered by the formulaic model components included in the RVM. The RVM and GP model are encapsulated within a Bayesian formulation using a hierarchical prior structure, meaning posterior distributions for the model discrepancy and parameter uncertainty can be obtained.

In order to efficiently solve the inference problem and to constrain the explanatory power of the GP model such that non-identifiability issues are reduced, an iterative expectation maximisation (EM) algorithm [22] is proposed. This is a similar approach to that taken in [21] for performing inference in an RVM, even though the statistical model is formed in a Bayesian manner. The approach is demonstrated to not only reduce the selection of erroneous terms, but also reduce parameter bias compared to using an RVM for equation discovery. Additionally, compared to conventional model discrepancy approaches where fully non-parametric models are used, the method allows physics to be identified by estimating the system parameters.

This paper is organised as follows. Details of the problem context investigated in this work are presented in Section 2. The Bayesian formulation used for parameter inference is proposed in Section 3. The EM-based inference procedure and the properties of the likelihood function are discussed in Sections 4 and 5, respectively. The proposed method is summarised in Section 6. Subsequently, parametric studies involving simulated and experimental data are presented in Section 7 before conclusions are made.

2. Problem context and model specification

Without loss of generality, the system considered in this study is assumed to have a set of measured inputs $\boldsymbol{x}$ ($N \times D$ matrix where N is the number of measured points and D is the dimension of input) and outputs $\boldsymbol{y}$ ($N \times 1$ vector). The governing equation can be represented as

$$\boldsymbol{y} = f(\boldsymbol{x}, \boldsymbol{\theta}) + \delta(\boldsymbol{x}, \boldsymbol{\psi}) + \boldsymbol{\varepsilon} \tag{1}$$

where $f(\boldsymbol{x},\boldsymbol{\theta})$ is the system model (describing the physics of the system) and is a function of the inputs $\boldsymbol{x}$ and a set of model parameters $\boldsymbol{\theta}$; $\delta(\boldsymbol{x},\boldsymbol{\psi})$ is the model discrepancy term that is a function of the inputs $\boldsymbol{x}$ and a set of model hyperparameters $\boldsymbol{\psi}$. The output of $\delta(\boldsymbol{x},\boldsymbol{\psi})$ is denoted as $\boldsymbol{\delta}$ $(N\times 1)$ and $\boldsymbol{\varepsilon}(N\times 1)$ is the measurement error.

First consider the system model $f(\boldsymbol{x},\boldsymbol{\theta})$ as a linear superposition over a set of basis functions given by

$$f(\boldsymbol{x},\boldsymbol{\theta}) = \boldsymbol{\Phi\theta} \tag{2}$$

where $\boldsymbol{\Phi} = [\,\Phi_1(\boldsymbol{x}) \quad \cdots \quad \Phi_M(\boldsymbol{x})\,]$ denotes the $N\times M$ 'design' matrix with its columns representing M basis functions with respect to the inputs $\boldsymbol{x}$. Conventionally, $\boldsymbol{\theta}$ is an $M\times 1$ weighting vector. In this work, $\boldsymbol{\theta}$ is treated as a set of associated system parameters to be identified when components of the physics-based model are used as basis functions.

A Gaussian Process (GP) model is adopted for the model discrepancy $\delta(.,.)$ in order to capture the missing physics of the system model. Specifically,

$$\delta(\boldsymbol{x},\boldsymbol{\psi}) \sim \mathscr{GP}(m(\boldsymbol{x}|\boldsymbol{\psi}), k(\boldsymbol{x},\boldsymbol{x}'|\boldsymbol{\psi})) \tag{3}$$

where $m(.)$ and $k(.,.)$ are the mean and covariance function (also known as a kernel function), with outputs of these functions denoted as $\boldsymbol{m}(N\times 1)$ and $\boldsymbol{K}$ $(N\times N)$ respectively, which characterize the GP model and also depend on a set of hyperparameters $\boldsymbol{\psi}$. More details on choices of mean and covariance functions are given in [17].

Finally, the measurement error $\boldsymbol{\varepsilon}$ is modelled as an independent and identically distributed (i.i.d.) zero mean Gaussian with variance σ^2.

Here an example in structural dynamics is presented to illustrate the proposed model. For structural dynamic systems, the governing equation is often represented in a state-space form, which can be expressed in the form of Eq. (2). Consider a Single-Degree-of-Freedom (SDOF) non-linear dynamic system as an example:

$$m\ddot{q} + c\dot{q} + kq + g(q,\dot{q}) = u \tag{4}$$

where m, c, k are the mass, damping and stiffness, $g(.,.)$ is an arbitrary function of displacement q and velocity $\dot{q}$, $\ddot{q}$ is the acceleration and u is the input forces. Assuming $x_1 = q$ and $x_2 = \dot{q}$, the state-space form of this system can be written as

$$\dot{x}_1 = x_2 \tag{5}$$

$$\dot{x}_2 = \frac{1}{m}\left(u - kx_1 - cx_2 - g\left(x_1,x_2\right)\right) \tag{6}$$

Consider $g(q,\dot{q}) = k_3q^3$ (a duffing oscillator) for example, the measured system response can be written in a compact matrix–vector notation as

$$\boldsymbol{y} = \boldsymbol{\Phi\theta} + \delta(\boldsymbol{x},\boldsymbol{\psi}) + \boldsymbol{\varepsilon} \tag{7}$$

Here, $\boldsymbol{y} = \dot{\boldsymbol{x}}_2$, $\boldsymbol{\Phi} = \left[\,\boldsymbol{u} \quad \boldsymbol{x}_2 \quad \boldsymbol{x}_1 \quad \boldsymbol{x}_1^3\,\right]$ and $\boldsymbol{\theta} = [\,1/m \quad -c/m \quad -k/m \quad -k_3/m\,]^T$ (note that the input force can also be merged into the design matrix). In this example, the input $\boldsymbol{x}$ includes the input force $\boldsymbol{u}$, the displacement $\boldsymbol{x}_1$ and the velocity $\boldsymbol{x}_2$. The output $\boldsymbol{y}$ is the acceleration response of the system. The system parameters $\boldsymbol{\theta}$, the hyperparameters of the GP model $\boldsymbol{\psi}$, and the noise variance σ^2 are the parameters to be inferred.

The premise of this work is that physics-based models, and more specifically the design matrix used in equation discovery, typically will have some level of missing physics due to simplifications or lack of knowledge (i.e., the design matrix, to some degree, is incomplete). It is therefore a typical scenario that the complete set of 'true' model components are not included in the design matrix and some of the candidate model components are likely to be incorrect, leading to model discrepancy and bias in the estimated system parameters. In order to address this problem, sparse Bayesian inference is used to infer $f(.,.)$, which allows some of the model components that do not provide a significant contribution to be removed during inference. The Least Absolute Shrinkage and Selector Operator (Lasso) method is the classic approach of introducing sparsity [23]. However, the most suitable sparsity level is determined in a non-probabilistic manner. In this work, following [7] the RVM [21] is used, which allows the posterior distribution of the system parameters to be obtained and for terms inside the design matrix to be removed in a probabilistic manner. It is also capable of being embedded in the Bayesian formulation that will be proposed in this work.

The Gaussian process model used for discrepancy modelling in this work is a non-parametric model that does not make strong assumptions about the form of the function but depends more on the training data compared to conventional regression models. It should be noted that 'non-parametric' here does not mean there is no model parameter to be inferred. A non-parametric model in this context means that the function is not defined explicitly and instead is defined through mean and covariance functions (which depends on hyperparameters $\boldsymbol{\psi}$).

Finally, the assumption of the measurement error (i.e., i.i.d. zero mean Gaussian) is justified because the main difference between the measured system output and physics-based model are captured by model discrepancy, which may even have captured some element of a coloured noise process. It can therefore be assumed that Gaussian noise on the residual model discrepancy is sufficient in most cases.

3. Bayesian formulation

Based on the specified model introduced in the last section, a Bayesian statistical model is proposed in this section, defining the hierarchical structure that governs the system parameters $\boldsymbol{\theta}$, the hyperparameters $\boldsymbol{\psi}$, and the noise variance σ^2. The sparsity of the model components for the system model is first introduced using the RVM method [21] which allows erroneous model components in the design matrix to be removed. Meanwhile, the posterior distribution of the model discrepancy term is defined through a GP formulation [17]. These two models are combined into the Bayesian formulation proposed in this section. Detailed derivation and discussion are given as follows.

First consider the system model $f(.,.)$. In order to allow some of the model components to be removed from the design matrix, sparsity is introduced to the system parameters $\boldsymbol{\theta}$ through hyperparameters $\boldsymbol{\alpha}$ ($M \times 1$) as

$$p(\boldsymbol{\theta}|\boldsymbol{\alpha}) = \mathcal{N}\left(\mathbf{0}, \boldsymbol{A}^{-1}\right) \tag{8}$$

where $\boldsymbol{A} = diag(\boldsymbol{\alpha})$. Specifically, the hyperparameters $\boldsymbol{\alpha}$ control the variance of the hierarchical prior distribution of the system parameters $\boldsymbol{\theta}$. The effect of this prior $\boldsymbol{\alpha}$ on $\boldsymbol{\theta}$ is that when $\alpha_i \to \infty$, $p(\theta_i|\alpha_i)$ will become a delta function, i.e., θ_i equals zero and the corresponding model component is excluded during inference.

For a Gaussian process model, the probability density function of $\boldsymbol{\delta}$ (i.e., the output of model discrepancy term) given the hyperparameters $\boldsymbol{\psi}$ can be expressed as

$$p(\boldsymbol{\delta}|\boldsymbol{\psi}) = \mathcal{N}(\boldsymbol{m}, \boldsymbol{K}) \tag{9}$$

where $\boldsymbol{m}(N \times 1)$ and $\boldsymbol{K}$ ($N \times N$) are the outputs of the mean and covariance function (i.e., $m(.)$ and $k(.,.)$ in Eq. (3)) for the GP model, respectively.

Following these definitions, the conditional distribution of $\boldsymbol{y}$ given the system parameters $\boldsymbol{\theta}$, the model discrepancy $\boldsymbol{\delta}$ and noise variance σ^2 can be written as

$$p\left(\boldsymbol{y}\middle|\boldsymbol{\theta}, \boldsymbol{\delta}, \sigma^2\right) = \mathcal{N}\left(\boldsymbol{\Phi\theta} + \boldsymbol{\delta}, \sigma^2\boldsymbol{I}\right) \tag{10}$$

The marginal likelihood function $p(\boldsymbol{y}|\boldsymbol{\alpha}, \boldsymbol{\psi}, \sigma^2)$ can then be given by

$$\begin{aligned} p\left(\boldsymbol{y}\middle|\boldsymbol{\alpha}, \boldsymbol{\psi}, \sigma^2\right) &= \iint p\left(\boldsymbol{y}\middle|\boldsymbol{\theta}, \boldsymbol{\delta}, \sigma^2\right) p(\boldsymbol{\theta}|\boldsymbol{\alpha}) p(\boldsymbol{\delta}|\boldsymbol{\psi}) d\boldsymbol{\theta} d\boldsymbol{\delta} \\ &= (2\pi)^{-N/2}\left|\sigma^2\boldsymbol{I} + \boldsymbol{K} + \boldsymbol{\Phi A}^{-1}\boldsymbol{\Phi}^T\right|^{-1/2} exp\left\{ -\frac{1}{2}(\boldsymbol{y} - \boldsymbol{m})^T\left(\sigma^2\boldsymbol{I} + \boldsymbol{K} + \boldsymbol{\Phi A}^{-1}\boldsymbol{\Phi}^T\right)^{-1}(\boldsymbol{y} - \boldsymbol{m}) \right\} \end{aligned} \tag{11}$$

Fundamentally, the hyperparameters $\boldsymbol{\alpha}, \boldsymbol{\psi}$ and σ^2 should be inferred based on the posterior distribution $p(\boldsymbol{\alpha}, \boldsymbol{\psi}, \sigma^2|\boldsymbol{y})$, which is given by

$$p\left(\boldsymbol{\alpha}, \boldsymbol{\psi}, \sigma^2|\boldsymbol{y}\right) = \frac{p(\boldsymbol{y}|\boldsymbol{\alpha}, \boldsymbol{\psi}, \sigma^2) p(\boldsymbol{\alpha}, \boldsymbol{\psi}, \sigma^2)}{p(\boldsymbol{y})} \tag{12}$$

This is generally intractable since the normalising integral (i.e., $p(\boldsymbol{y}) = \int p(\boldsymbol{y}|\boldsymbol{\alpha}, \boldsymbol{\psi}, \sigma^2) p(\boldsymbol{\alpha}, \boldsymbol{\psi}, \sigma^2) d\boldsymbol{\alpha} d\boldsymbol{\psi} d\sigma^2$) cannot be analytically obtained. Instead, the parameter inference can be achieved using a type-II maximum-likelihood method [24]. Assuming a uniform prior $p(\boldsymbol{\alpha}, \boldsymbol{\psi}, \sigma^2)$, the parameters then can be obtained by maximising the marginal likelihood in Eq. (11) or equivalently, its logarithm:

$$\begin{aligned} L\left(\boldsymbol{\alpha}, \boldsymbol{\psi}, \sigma^2\right) = &-\frac{N}{2} log(2\pi) - \frac{1}{2} logdet(\boldsymbol{C}) \\ &-\frac{1}{2}(\boldsymbol{y} - \boldsymbol{m})^T\boldsymbol{C}^{-1}(\boldsymbol{y} - \boldsymbol{m}) \end{aligned} \tag{13}$$

where

$$\boldsymbol{C} = \sigma^2\boldsymbol{I} + \boldsymbol{K} + \boldsymbol{\Phi A}^{-1}\boldsymbol{\Phi}^T \tag{14}$$

It should be noted that different priors over $\boldsymbol{\alpha}$ and σ^2 can also be considered. For example, a suitable choice can be Gamma distribution [24], which may lead to additional terms in the marginal likelihood (see details in Appendix A of [21]). The system parameter $\boldsymbol{\theta}$ can be estimated through its posterior distribution given by

$$\begin{aligned} p\left(\boldsymbol{\theta}\middle|\boldsymbol{y}, \boldsymbol{\alpha}, \boldsymbol{\psi}, \sigma^2\right) &= \frac{p(\boldsymbol{y}|\boldsymbol{\theta}, \boldsymbol{\psi}, \sigma^2) p(\boldsymbol{\theta}|\boldsymbol{\alpha})}{p\left(\boldsymbol{y}\middle|\boldsymbol{\alpha}, \boldsymbol{\psi}, \sigma^2\right)} \\ &= \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma}) \end{aligned} \tag{15}$$

where

$$\Sigma = \left[\Phi^T\left(\sigma^2 I + K\right)^{-1}\Phi + A\right]^{-1} \tag{16}$$

$$\boldsymbol{\mu} = \boldsymbol{\Sigma}\left(\sigma^2\boldsymbol{I} + \boldsymbol{K}\right)^{-1}\boldsymbol{\Phi}^T(\boldsymbol{y} - \boldsymbol{m}) \tag{17}$$

Regression models based on Gaussian process have very strong explanatory power to approximate any arbitrary functions, especially when universal kernels are used [25]. It is possible that the global maximum of the likelihood function (i.e., Eq. (11)) is located at a certain set of hyperparameters $\boldsymbol{\psi}$ with the system parameters $\boldsymbol{\theta}$ equal to zero. Under this situation, the Gaussian process model takes over the role of the system model and $\boldsymbol{y} = \delta(\boldsymbol{x}, \boldsymbol{\psi}) + \boldsymbol{\varepsilon}$, which defeats the initial purpose (i.e., using $\delta(.,.)$ to capture the model discrepancy). To balance the explanatory power of $f(.,.)$ and $\delta(.,.)$ properly, i.e., finding the maximum of the likelihood function where the system model $f(.,.)$ dominates, the EM method is used to infer the parameters of these two models. Details are discussed in the next section.

4. Inference method

The proposed formulation in Section 3 allows system parameters to be inferred in a fully Bayesian manner. i.e., the posterior distribution of the parameters is stated in Eq. (12). Although different inference methods can be used to solve the Bayesian model specified in Section 3, this work considers utilising the expectation maximisation (EM) [22] method. The EM method is chosen as the inference method, as it is an efficient parameter estimation technique based on the marginal likelihood function (i.e. Eq. (11)) when a uniform prior is assumed (as is assumed in Section 3). The EM method is a popular approach of maximising the likelihood function in an iterative way by using models that depend on latent variables. Specifically, consider the model parameters $\boldsymbol{\theta}$ as the latent variables. The parameters $\boldsymbol{\alpha}$, $\boldsymbol{\psi}$ and σ^2 can then be updated between the expectation (E-) step and the maximisation (M-) step in the following way:

E-step: Compute the expected log-likelihood function

$$Q\left(\left\{\boldsymbol{\alpha}, \boldsymbol{\psi}, \sigma^2\right\} \middle| \left\{\boldsymbol{\alpha}, \boldsymbol{\psi}, \sigma^2\right\}^{(t)}\right) = \mathrm{E}_{\boldsymbol{\theta}|\boldsymbol{y}, \{\boldsymbol{\alpha}, \boldsymbol{\psi}, \sigma^2\}^{(t)}}\left[log\left(p\left(\boldsymbol{y}|\boldsymbol{\theta}, \boldsymbol{\psi}, \sigma^2\right)p(\boldsymbol{\theta}|\boldsymbol{\alpha})\right)\right] \tag{18}$$

where E[.] is the expectation operation and t is the iteration number. Then

M-step: Maximise $Q\left(\left\{\boldsymbol{\alpha}, \boldsymbol{\psi}, \sigma^2\right\} \middle| \left\{\boldsymbol{\alpha}, \boldsymbol{\psi}, \sigma^2\right\}^{(t)}\right)$ to obtain

$$\left\{\boldsymbol{\alpha}, \boldsymbol{\psi}, \sigma^2\right\}^{(t+1)} = argmaxQ\left(\left\{\boldsymbol{\alpha}, \boldsymbol{\psi}, \sigma^2\right\} \middle| \left\{\boldsymbol{\alpha}, \boldsymbol{\psi}, \sigma^2\right\}^{(t)}\right) \tag{19}$$

Calculating the $Q(.)$ function involves the first and second moment of $\boldsymbol{\theta}$ based on the current estimation of $\{\boldsymbol{\alpha}, \boldsymbol{\psi}, \sigma^2\}$, which can be obtained based on the conditional distribution $p(\boldsymbol{\theta}|\boldsymbol{y}, \boldsymbol{\alpha}, \boldsymbol{\psi}, \sigma^2)$ in Eq. (15). The first moment of $\boldsymbol{\theta}$ is equal to $\boldsymbol{\mu}$ given in Eq. (17) and the second moment can be calculated as

$$\mathrm{E}_{\boldsymbol{\theta}|\boldsymbol{y}, \boldsymbol{\alpha}, \boldsymbol{\psi}, \sigma^2}\left[\boldsymbol{\theta}^2\right] = \boldsymbol{\Sigma} + \boldsymbol{\mu}^T\boldsymbol{\mu} \tag{20}$$

Substituting Eq. (17) and Eq. (20) into Eq. (18), the $Q(.)$ function can then be rewritten as

$$\begin{aligned} Q &= \mathrm{E}_{\boldsymbol{\theta}|\boldsymbol{y}, \{\boldsymbol{\alpha}, \boldsymbol{\psi}, \sigma^2\}^{(t)}}\left[log\left(p\left(\boldsymbol{y}|\boldsymbol{\theta}, \boldsymbol{\psi}, \sigma^2\right)p(\boldsymbol{\theta}|\boldsymbol{\alpha})\right)\right] \\ &= \mathrm{E}_{\boldsymbol{\theta}|\boldsymbol{y}, \{\boldsymbol{\alpha}, \boldsymbol{\psi}, \sigma^2\}^{(t)}}\left[logp(\boldsymbol{\theta}|\boldsymbol{\alpha})\right] + \mathrm{E}_{\boldsymbol{\theta}|\boldsymbol{y}, \{\boldsymbol{\alpha}, \boldsymbol{\psi}, \sigma^2\}^{(t)}}\left[logp\left(\boldsymbol{y}|\boldsymbol{\theta}, \boldsymbol{\psi}, \sigma^2\right)\right] \\ &= Q_1 + Q_2 \end{aligned} \tag{21}$$

where

$$\begin{aligned} Q_1 &= \mathrm{E}_{\boldsymbol{\theta}|\boldsymbol{y}, \{\boldsymbol{\alpha}, \boldsymbol{\psi}, \sigma^2\}^{(t)}}\left[logp(\boldsymbol{\theta}|\boldsymbol{\alpha})\right] \\ &= -\frac{N}{2}log(2\pi) - \sum_{i=1}^{N}\left(loga_i^{-1} + \frac{\boldsymbol{\Sigma}_{ii} + \mu_i^2}{a_i^{-1}}\right) \end{aligned} \tag{22}$$

and

$$\begin{aligned} Q_2 &= \mathrm{E}_{\boldsymbol{\theta}|\boldsymbol{y}, \{\boldsymbol{\alpha}, \boldsymbol{\psi}, \sigma^2\}^{(t)}}\left[logp\left(\boldsymbol{y}|\boldsymbol{\theta}, \boldsymbol{\psi}, \sigma^2\right)\right] \\ &= \mathrm{E}_{\boldsymbol{\theta}|\boldsymbol{y}, \{\boldsymbol{\alpha}, \boldsymbol{\psi}, \sigma^2\}^{(t)}}\left[log\int p\left(\boldsymbol{y}|\boldsymbol{\theta}, \boldsymbol{\delta}, \sigma^2\right)p(\boldsymbol{\delta}|\boldsymbol{\psi})d\boldsymbol{\delta}\right] \\ &= -\frac{N}{2}log(2\pi) - \frac{1}{2}logdet\left(\sigma^2\boldsymbol{I} + \boldsymbol{K}\right) \\ &\quad -\frac{1}{2}[\boldsymbol{y} - (\boldsymbol{m} + \boldsymbol{\Phi}\boldsymbol{\mu})]^T\left(\sigma^2\boldsymbol{I} + \boldsymbol{K}\right)^{-1}[\boldsymbol{y} - (\boldsymbol{m} + \boldsymbol{\Phi}\boldsymbol{\mu})] - \frac{1}{2}\mathrm{tr}\left[\boldsymbol{\Phi}^T\left(\sigma^2\boldsymbol{I} + \boldsymbol{K}\right)^{-1}\boldsymbol{\Phi}\boldsymbol{\Sigma}\right] \end{aligned} \tag{23}$$

It can be seen that $\boldsymbol{\alpha}$ and $\{\boldsymbol{\psi}, \sigma^2\}$ are involved in the $Q(.)$ function through Q_1 and Q_2, respectively, which means they can be updated separately. The hyperparameters $\boldsymbol{\alpha}$ can be inferred by maximising Q_1. Note that $\boldsymbol{\alpha}$ is related to Q_1 through the form of $lnx + c/x$, which has a unique minimum $1 + lnc$ at $x = c$. The hyperparameters $\boldsymbol{\alpha}$ can then be updated as:

$$\alpha_i = \frac{1}{\boldsymbol{\Sigma}_{ii} + \mu_i^2} \tag{24}$$

It can also be shown that Eq. (24) is equivalent to $\partial L(\boldsymbol{\alpha}, \boldsymbol{\psi}, \sigma^2)/\partial \alpha_i = 0$ [21]. Investigation of $L(\boldsymbol{\alpha}, \boldsymbol{\psi}, \sigma^2)$ with respect to $\boldsymbol{\alpha}$ is presented in the next section, which further facilitates the computation. On the other hand, the hyperparameters of the GP model $\boldsymbol{\psi}$ as well as the noise variance σ^2 can be updated by maximising Q_2. Noting that $\sigma^2\boldsymbol{I} + \boldsymbol{K}$ is a symmetric matrix, and so techniques like Cholesky decomposition can be used for estimating its inverse.

The foregoing E and M steps are iteratively repeated until convergence of the marginal likelihood function, where the convergence criteria are discussed at the end of Section 6. Based on the assumption that the system model can capture the main behaviour of the real system, the EM iteration can be started with initial values of $\{\boldsymbol{\psi}, \sigma^2\}$ set to zero (i.e., the measured output contains system model response only). By doing so, $\{\boldsymbol{\alpha}, \boldsymbol{\psi}, \sigma^2\}$ converges to the target optimum estimation corresponding to a balance between $f(.,.)$ and $\delta(.,.)$ in which $f(.,.)$ is the dominant term. It should be noted that unlike conventional inference methods that try to find the global maximum of the likelihood function, the proposed method in this work uses an EM method to find the physically relevant maximum (regardless of whether it is global or not) where the system model dominates by deliberately controlling the initialisation point (and hence the searching space).

5. Analysis of marginal likelihood

The properties of the log-likelihood function $L(\boldsymbol{\alpha}, \boldsymbol{\psi}, \sigma^2)$ (i.e., Eq. (13)) with respect to $\boldsymbol{\alpha}$ are further investigated in this section. The sparsity due to $\boldsymbol{\alpha}$ is revealed, showing that $\boldsymbol{\alpha}$ can be updated analytically given the remaining parameters. The analysis in this section helps to facilitate computation in the proposed method.

Consider the dependence of $L(\boldsymbol{\alpha}, \boldsymbol{\psi}, \sigma^2)$ on a single hyperparameter α_i. The covariance function $\boldsymbol{C}$ can be decomposed as:

$$\begin{aligned} \boldsymbol{C} &= \sigma^2\boldsymbol{I} + \boldsymbol{K} + \sum_{m \neq i} a_m^{-1}\Phi_m\Phi_m^T + a_i^{-1}\Phi_i\Phi_i^T \\ &= \boldsymbol{C}_{-i} + a_i^{-1}\Phi_i\Phi_i^T \end{aligned} \tag{25}$$

The determinant and inverse of $\boldsymbol{C}$ can then be given as

$$det(\boldsymbol{C}) = det(\boldsymbol{C}_{-i})det\left(1 + a_i^{-1}\Phi_i^T\boldsymbol{C}_{-i}^{-1}\Phi_i\right) \tag{26}$$

$$\boldsymbol{C}^{-1} = \boldsymbol{C}_{-i}^{-1} - \frac{\boldsymbol{C}_{-i}^{-1}\Phi_i\Phi_i^T\boldsymbol{C}_{-i}^{-1}}{\alpha_i + \Phi_i^T\boldsymbol{C}_{-i}^{-1}\Phi_i} \tag{27}$$

The log-likelihood function $L(\boldsymbol{\alpha}, \boldsymbol{\psi}, \sigma^2)$ can be rearranged as

$$\begin{aligned} L\left(\boldsymbol{\alpha}, \boldsymbol{\psi}, \sigma^2\right) &= -\frac{1}{2}\left[Nlog(2\pi) + logdet(\boldsymbol{C}_{-i}) + (\boldsymbol{y} - \boldsymbol{m})^T\boldsymbol{C}_{-i}^{-1}(\boldsymbol{y} - \boldsymbol{m})\right] \\ &\quad + \frac{1}{2}\left\{log\alpha_i - log\left(\alpha_i + \Phi_i^T\boldsymbol{C}_{-i}^{-1}\Phi_i\right) - \frac{\left[\Phi_i^T\boldsymbol{C}_{-i}^{-1}(\boldsymbol{y} - \boldsymbol{m})\right]^2}{\alpha_i + \Phi_i^T\boldsymbol{C}_{-i}^{-1}\Phi_i}\right\} \\ &= L(\boldsymbol{\alpha}_{-i}) + l(\alpha_i) \end{aligned} \tag{28}$$

where $L(\boldsymbol{\alpha}_{-i})$ is the term which does not depend on α_i. The maximum of $L\left(\boldsymbol{\alpha}, \boldsymbol{\psi}, \sigma^2\right)$ with respect to α_i can then be obtained by calculating $\partial l(\alpha_i)/\partial\alpha_i = 0$, which gives

$$\alpha_i = \frac{s_i^2}{q_i^2 - s_i} \quad \left(q_i^2 > s_i\right) \tag{29}$$

$$\alpha_i = \infty \quad \left(q_i^2 \leq s_i\right) \tag{30}$$

where

$$q_i = \Phi_i^T\boldsymbol{C}_{-i}^{-1}(\boldsymbol{y} - \boldsymbol{m}) \tag{31}$$

$$s_i = \Phi_i^T\boldsymbol{C}_{-i}^{-1}\Phi_i \tag{32}$$

This means when $q_i^2 > s_i$, θ_i does not equal zero and the corresponding basis function should be involved in the design matrix. When $q_i^2 \leq s_i$, $p(\theta_i|\alpha_i)$ becomes a delta function at zero, which means the corresponding basis function can be removed from the model. It can also be shown that Eq. (29) is equivalent to Eq. (24) when the corresponding Φ_i is currently in the model. For computation and updating, it is more convenient to compute

$$Q_i = \Phi_i^T\boldsymbol{C}^{-1}(\boldsymbol{y} - \boldsymbol{m}) \tag{33}$$

$$S_i = \Phi_i^T \boldsymbol{C}^{-1} \Phi_i \tag{34}$$

such that q_i and s_i can be given by

$$q_i = \frac{\alpha_i Q_i}{\alpha_i - S_i} \tag{35}$$

$$s_i = \frac{\alpha_i S_i}{\alpha_i - S_i} \tag{36}$$

Noting that when $\alpha_i \to \infty$, $q_i = Q_i$ and $s_i = S_i$.

The above analysis shows that given the remaining parameters, the parameter $\boldsymbol{\alpha}$ can be updated analytically through Eqs. (29) and (30) without resorting to brute force optimisation, which helps to facilitate computation.

6. Summary of procedure

The proposed method can be conducted as follows:

1. Initialise $\{\boldsymbol{\psi}, \sigma^2\}$.
2. Initialise with a single basis function in the design matrix, e.g., the first basis function, and calculate the initial α_1 using Eq. (29), set other $\alpha_i = \infty$.
3. Update $\{\boldsymbol{\mu}, \boldsymbol{\Sigma}\}$ using Eqs. (15) and (16), together with all Q_i and S_i using Eqs. (33) and (34).
4. Select a candidate basis Φ_i from the whole design matrix $\boldsymbol{\Phi}$ and compute q_i and s_i.
5. Compare q_i^2 and s_i:
   a. If $q_i^2 > s_i$ and $\alpha_i \neq \infty$ (i.e., Φ_i is currently in the model), re-estimate α_i using Eq. (29).
   b. If $q_i^2 > s_i$ and $\alpha_i = \infty$, add Φ_i into the model with updated α_i using Eq. (29).
   c. If $q_i^2 \leq s_i$, delete Φ_i from the model and set $\alpha_i = \infty$.
6. Update $\{\boldsymbol{\psi}, \sigma^2\}$ by maximising Eq. (23).
7. Go to Step 3 until convergence.

In this work, the hyperparameters $\boldsymbol{\psi}$ are initialised as zero (i.e., without a Gaussian process model involved in the beginning) and the noise variance σ^2 is set as a nominal value, 1% (say) of the variance of the system output to start iteration. The convergence criteria can be set based on the marginal likelihood L in Eq. (13) or based on the change in the parameters $\{\boldsymbol{\alpha}, \boldsymbol{\psi}, \sigma^2\}$. In this work, it is set as $\frac{|L_{new} - L_{old}|}{|L_{new}| + |L_{old}|} < 10^{-6}$.

7. Illustrative examples

Three examples are presented in this section to illustrate the proposed method. A parametric study has been conducted in the first example based on a simple one-dimension polynomial function. The properties of the likelihood function for both the conventional sparse Bayesian method (i.e., RVM) and the proposed method (referred to as GP-RVM in this Section for conciseness) are compared. The second example focuses on identifying the system parameters of a Duffing oscillator, a nonlinear system with position-dependent stiffness, where again the inferred parameters from the RVM and GP-RVM methods are compared. In addition, this example considers different forms of basis functions that can be used to capture the nonlinearity of the system. Finally, the applicability of the GP-RVM method in an experimental setting is investigated in the third example, where the experimental data from the Silverbox system [26] is considered.

It is noted that in the following illustrative examples, the GP-RVM method is applied using a zero mean function and a squared exponential covariance function. That is, $\boldsymbol{m} = \boldsymbol{0}$ and the (i, j) entry of the covariance matrix $\boldsymbol{K}$ is given by

$$\boldsymbol{K}_{i,j} = \sigma_f^2 exp\left(-\frac{\left\| \boldsymbol{x}_i - \boldsymbol{x}_j \right\|^2}{2l^2} \right) \tag{37}$$

where $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ denote the i-th and j-th index. This choice of covariance function means the hyperparameter set becomes $\boldsymbol{\psi} = \left\{\sigma_f^2, l^2\right\}$, denoting the signal variance and length-scale respectively.

7.1. Parametric study

Consider a simple polynomial function defined as

$$y = \boldsymbol{\Phi}_{true} \boldsymbol{\theta} + \boldsymbol{\varepsilon} \tag{38}$$

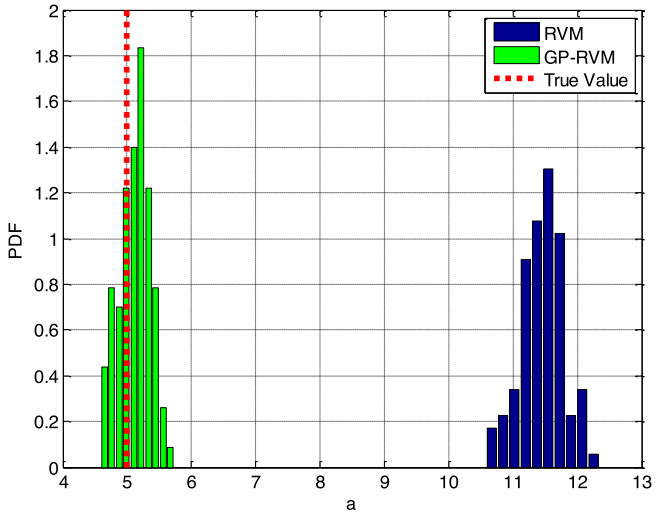

Fig. 1. Histograms of identified parameter a from the training data sets (green bar: GP-RVM method; blue bar: RVM method; dashed line: true value).

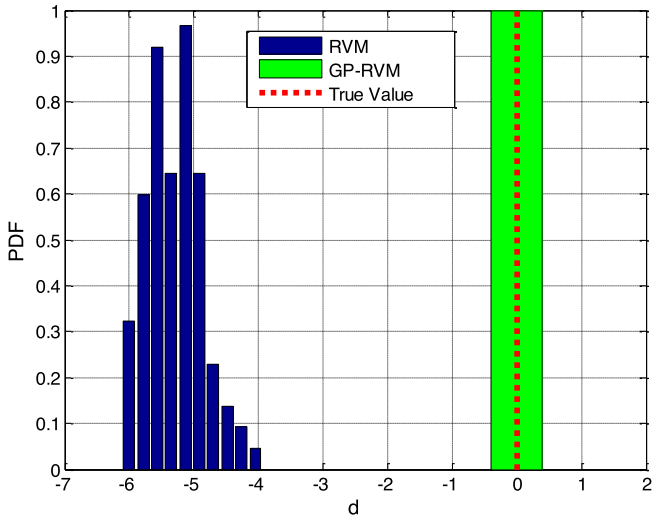

Fig. 2. Histograms of identified parameter d from the training data sets (green bar: GP-RVM method; blue bar: RVM method; dashed line: true value).

where $\boldsymbol{\Phi}_{true}$ is the true design matrix given as $\boldsymbol{\Phi}_{true} = \begin{bmatrix} x^3 & x^2 & sinx \end{bmatrix}$, $\boldsymbol{\theta}$ is the associated system parameters set $\boldsymbol{\theta} = \begin{bmatrix} a & b & c \end{bmatrix}^T = \begin{bmatrix} 5 & 1 & 2 \end{bmatrix}^T$, $\boldsymbol{\varepsilon}$ is the measurement noise. The input data is sampled uniformly within the range of [-1, 1] with 100 points. The noise is randomly generated as zero mean Gaussian. The standard deviation of the noise is set to be 10% of the system output.

The aim of this example is to explore the misspecification of basis function terms and missing 'physics'. Consider a candidate design matrix of $\boldsymbol{\Phi}_1 = \begin{bmatrix} x^3 & x^5 \end{bmatrix}$ with the associated parameter $\boldsymbol{\theta}_1 = \begin{bmatrix} a & d \end{bmatrix}^T$. Both the conventional sparse Bayesian method (i.e., RVM) and the proposed method (i.e., GP-RVM) have been applied to estimate the value of the associated system parameters. Fig. 1 and Fig. 2 show the distributions of the identified system parameters a and d over 100 data sets (independent draws of noise process for each data set), respectively. Due to the model terms $\begin{bmatrix} x^2 & sinx \end{bmatrix}$ being excluded from the design matrix as well as the erroneously included term x^5, the estimated system parameters from the original RVM method are biased. On the other hand, the GP-RVM method provides a much better estimation, as shown in Fig. 1. The estimated parameter a is very close to the true value that is used to generate the data and the erroneous term x^5 is excluded after inference for all of the datasets (i.e., equal to zero) when using the GP-RVM method.

Fig. 3 shows the output mean model prediction as well as ± 2 posterior standard deviation based on these two methods for a typical training data set. The normalised mean square errors (NMSE) of the mean model prediction of RVM and GP-RVM methods (compared to the true function values) are calculated to be 3.4×10^{-4} and 8.8×10^{-5}, respectively. It can be seen that the model prediction based on the GP-RVM method is much closer to the true system output compared to the RVM method.

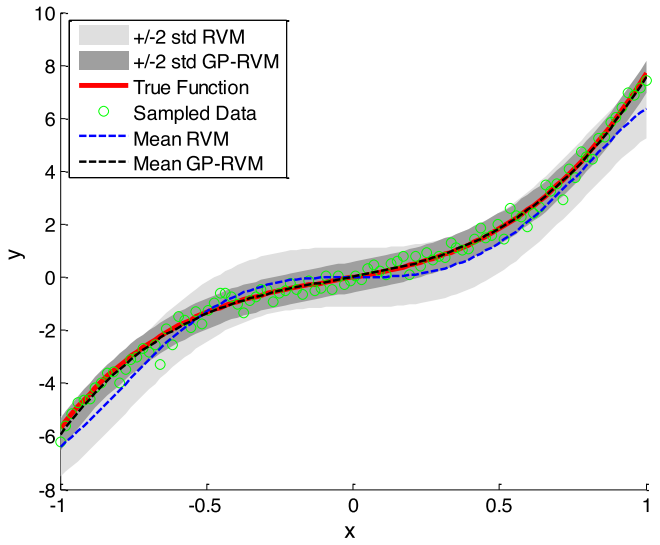

Fig. 3. Comparison of the true system output against model predictions from the RVM and GP-RVM methods (red solid line: true function; circle: sampled data; blue dashed line: predicted mean of RVM method; black dashed line: predicted mean of GP-RVM method; light grey area: ±2 standard deviation of RVM method; dark grey area: ±2 standard deviation of GP-RVM method).

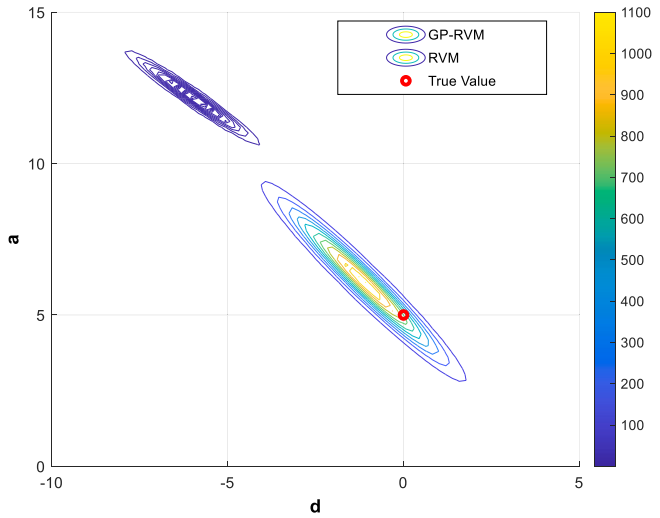

Fig. 4. Marginal likelihood against system parameters (red dot: true value; coloured contour: GP-RVM method; blue contour: RVM method).

Fig. 4 shows the marginal likelihood function (i.e., $p(\boldsymbol{y}|\boldsymbol{\theta})$) against the system parameters. It can be seen that the maximum of the marginal likelihood based on the GP-RVM method is closer to the true values compared to that of the RVM method.

For reference, the number of iterations before convergence for the GP-RVM method is around 20 for each training dataset.

7.2. Duffing oscillator

The second example considers a more realistic engineering problem, namely considering the equation discovery and parameter inference of a Duffing oscillator. The data is simulated based on the following equation:

$$m\ddot{y}(t) + c\dot{y}(t) + k(y(t))y(t) = u(t) \tag{39}$$

where $u(t)$ denotes the excitation applied to the system and $y(t)$ is the displacement. Parameters m and c denote the mass and damping of the system. An overdot represents differentiation with respect to time. The nonlinearity is introduced by position-dependent stiffness

Table 1
System parameters for the simulated duffing oscillator.

Parameter	m	c	a	b
Value (in SI units)	1	2.151	10	10^5

Table 2
Parameter identification results for the duffing oscillator example, first scenario where true values are m = 1 and c = 2.151.

Data Set No.	RVM		GP-RVM	
	m	c	m	c
1	0.973	2.051	1.001	2.154
2	0.994	2.096	1.000	2.146
3	1.023	2.166	1.000	2.156
4	0.983	2.192	1.000	2.135
5	0.982	2.121	0.999	2.149
6	0.975	2.078	1.001	2.157
7	0.982	2.059	1.000	2.153
8	1.014	2.134	1.000	2.150
9	0.989	2.090	1.000	2.149
10	1.001	2.334	1.000	2.147

$$k(y(t)) = a + by^2(t) \tag{40}$$

where a and b are the associated stiffness parameters.

Table 1 summarises the values of system parameters used for generating the system response data. The system is excited using Gaussian white noise with a standard deviation of 1kN. The system response is contaminated with a Gaussian white noise with a standard deviation equal to 1% of the standard deviation of the response signal. In this example, ten seconds of data are recorded for each data set with a sampling rate of 50 Hz. Ten sets of data are used for analysis for each scenario.

In this example, the displacement and velocity response of the system (i.e., y and $\dot{y}$, respectively) are assumed to be known and considered as input to the design matrix together with the applied excitation (i.e., u). The acceleration response (i.e., $\ddot{y}$) is considered as the system output. In this context, the system can be written in the form of a design matrix and system parameters as

$$\ddot{y} = \boldsymbol{\Phi}_{true}\boldsymbol{\theta} \tag{41}$$

where $\boldsymbol{\Phi}_{true} = \begin{bmatrix} u & \dot{y} & y & y^3 \end{bmatrix}$ and $\boldsymbol{\theta} = [1/m \quad -c/m \quad -a/m \quad -b/m]^T$. When conducting equation discovery using the GP-RVM method, the input of the GP model (for the model discrepancy term) is assumed to be the displacement of the system. This is justified based on the assumption that it is known the nonlinearity of the system depends on the displacement, but the exact form of the nonlinearity is unknown.

In the first scenario, consider a linear system without the stiffness term as

$$\ddot{y}(t) = \frac{1}{m}u(t) - \frac{c}{m}\dot{y}(t) = \boldsymbol{\Phi}_1\boldsymbol{\theta}_1 \tag{42}$$

where $\boldsymbol{\Phi}_1 = \begin{bmatrix} u & \dot{y} \end{bmatrix}$ and $\boldsymbol{\theta}_1 = [1/m \quad -c/m]^T$. The term $k(y(t))y(t)$ is completely ignored in this case and becomes the model discrepancy term. It is uncommon in real applications that the stiffness term is not considered but here such model is used to investigate this situation where the missing physics is *extremely* significant.

Both the RVM method and the GP-RVM method (where the Gaussian process model is expected to capture the model discrepancy term) have been applied to identify the model parameters. Table 2 shows the estimated system parameters for 10 independent realisations from the simulation. The identified system parameters from the RVM analysis are biased due to the absence of the term $k(y(t))y(t)$ in the model. The averaged bias of the mass and damping parameters are calculated to be 1.60% and 3.11%, respectively. This is not the case for the GP-RVM method however. The identified mass and damping parameters are close to the actual values used for data generation, with the average bias being 0.03% and 0.21%, respectively. Fig. 5 shows model predictions (mean and ± 2 posterior standard deviation) of these two methods (trained using data set No.10), as well as the simulated response based on an independent test data set. The NMSEs of the mean prediction from the RVM and GP-RVM methods are calculated to be 4.4×10^{-4} and 9.4×10^{-6}, respectively. The residuals between the mean prediction and the true data for both the RVM and GP-RVM are plotted in Fig. 6. It can be seen that the mean prediction from the GP-RVM method fits well with the measured response compared to that from the RVM method. The residual of the GP-RVM method is very close to zero, indicating that the Gaussian process model can appropriately capture the behaviour of the model discrepancy due to the absence of the stiffness term. The average iteration number for the GP-RVM method is 4 in this scenario.

In the second scenario, consider the system model as:

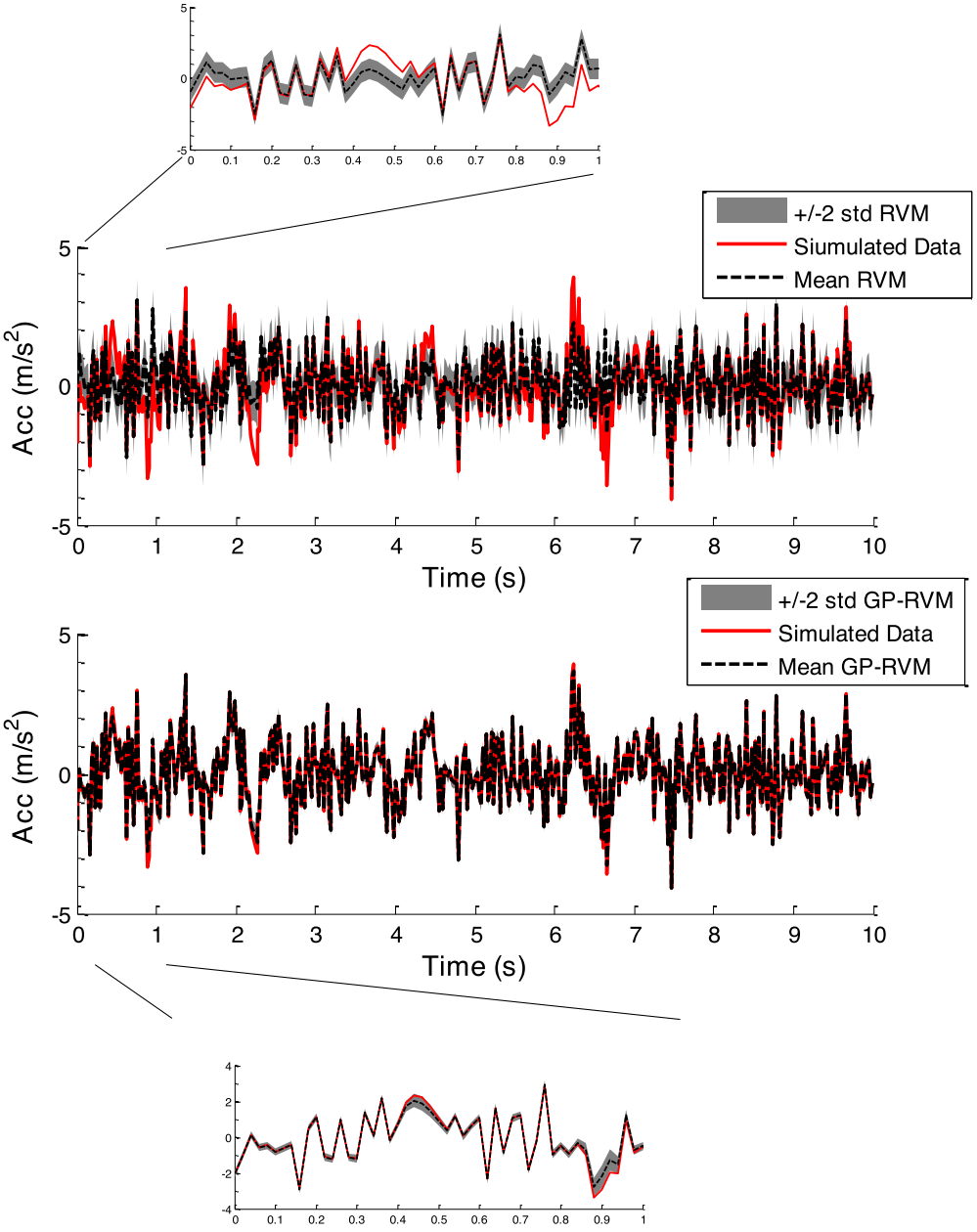

Fig. 5. Comparison of the simulated response and model predictions from the RVM and GP-RVM methods for the first scenario (red solid line: simulated value; black dashed line: predicted mean of applied method; grey area: ±2 standard deviation of applied method).

$$\ddot{y}(t) = \frac{1}{m}u(t) - \frac{c}{m}\dot{y}(t) - (dy(t) + ey^3(t))y(t) = \boldsymbol{\Phi}_2\boldsymbol{\theta}_2 \tag{43}$$

where $\boldsymbol{\Phi}_2 = \begin{bmatrix} u & \dot{y} & y^2 & y^4 \end{bmatrix}$ and $\boldsymbol{\theta}_2 = [1/m \quad -c/m \quad -d/m \quad -e/m]^T$. The position dependant stiffness is considered in the model but in an incorrect form. Similar to the first scenario, the associated parameters are identified using both the RVM method and the GP-RVM method based on ten datasets (independent realisations from the simulation). Table 3 summarises the identification results. For the RVM method, the estimated mass and damping parameters are biased, with the average bias being 0.49% and 4.55%, respectively. The erroneous stiffness forms with associated parameters (i.e., d and e) are not excluded through the sparsity criteria (which are not involved in the true model). On the other hand, the GP-RVM method provides a better estimation result, where the identified mass and

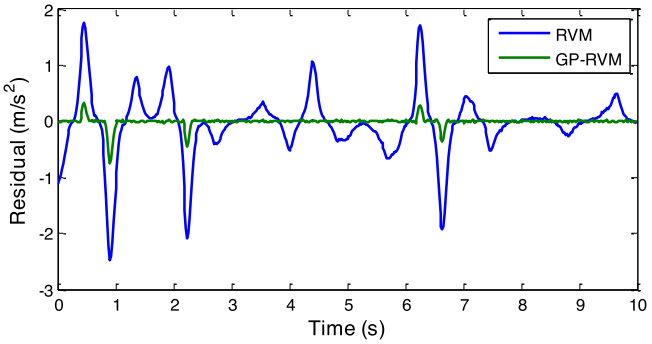

Fig. 6. Residual between the predictive mean and simulated response for RVM and GP-RVM method, first scenario (blue line: RVM method; green line: GP-RVM method).

Table 3
Parameter identification results for the duffing oscillator example, second scenario, where the true values are m = 1, c = 2.151, d = 0 and e = 0.

Data Set No.	RVM				GP-RVM			
	m	c	$d(\times 10^3)$	$e(\times 10^6)$	m	c	$d(\times 10^3)$	$e(\times 10^6)$
1	1.006	2.283	1.185	6.573	1.000	2.140	0.000	0.000
2	0.998	2.248	3.556	12.612	1.000	2.158	0.000	0.000
3	0.998	2.136	0.495	4.560	1.000	2.151	0.000	0.000
4	1.003	2.146	2.107	12.389	1.000	2.147	0.000	0.000
5	1.003	2.162	0.000	0.694	1.000	2.148	0.000	0.000
6	1.002	2.152	2.764	16.634	1.000	2.151	0.000	0.000
7	1.009	2.190	1.014	2.264	1.000	2.142	0.000	0.000
8	0.999	2.159	1.986	8.654	1.000	2.148	0.000	0.000
9	0.992	2.284	0.000	3.292	0.999	2.141	0.000	0.000
10	0.991	2.137	0.299	1.338	1.000	2.140	0.000	0.000

Table 4
Parameter identification results for the duffing oscillator example, third scenario where true values are m = 1, c = 2.151, a = 10, b = 10^5.

Data Set No.	RVM				GP-RVM			
	m	c	a	$b(\times 10^5)$	m	c	a	$b(\times 10^5)$
1	1.000	2.142	9.943	0.849	1.000	2.139	9.877	0.851
2	1.000	2.141	10.068	0.771	1.000	2.138	10.016	0.772
3	1.000	2.161	9.998	1.120	1.000	2.159	9.945	1.121
4	0.999	2.141	10.165	0.906	0.999	2.139	10.111	0.907
5	1.001	2.159	10.043	0.981	1.001	2.157	9.987	0.982
6	1.000	2.149	10.069	0.698	1.000	2.146	10.014	0.699
7	1.000	2.155	9.989	1.047	1.000	2.153	9.925	1.048
8	1.001	2.155	10.032	1.045	1.001	2.152	9.985	1.046
9	1.000	2.153	10.065	0.841	1.000	2.150	10.029	0.841
10	1.000	2.149	10.016	1.000	1.000	2.146	9.958	1.001

damping parameters are much closer to the true values. The erroneous forms are removed and the associated stiffness parameters are equal to zero in all datasets. In this scenario, the average iteration number of the GP-RVM method is 5.

In the third scenario, consider the system model as the true form (i.e., identical to the design matrix and system parameters in Eq. (41)). In this case, there are no missing terms or erroneous terms involved in the design matrix. Table 4 lists the identification results based on the RVM and the GP-RVM methods. The identified system parameters based on these two methods are close to each other and they are also very close to the true values. This shows that the GP-RVM method works similarly to the classic RVM method when there is no model discrepancy involved; the flexibility of the GP does not reduce the explanatory power of the system model terms. The average iteration number for the GP-RVM method is 4.8 in this scenario.

From these three scenarios, it can be seen that compared to the RVM method, the GP-RVM method provides better parameter estimation results with less bias involved when model discrepancy exists. However, it should be noted that in this example the displacement and velocity response of the system are assumed to be known so that the system output (i.e., the acceleration response) can be written as a linear combination of the input force, displacement and velocity with associated system parameters. In real applications, it is not always possible to measure all the system responses. Normally only the acceleration response is available.

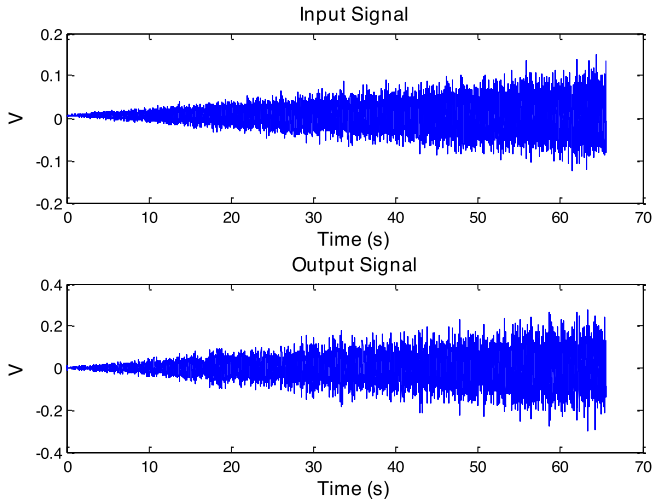

Fig. 7. Input and output signals from the Silverbox training data set.

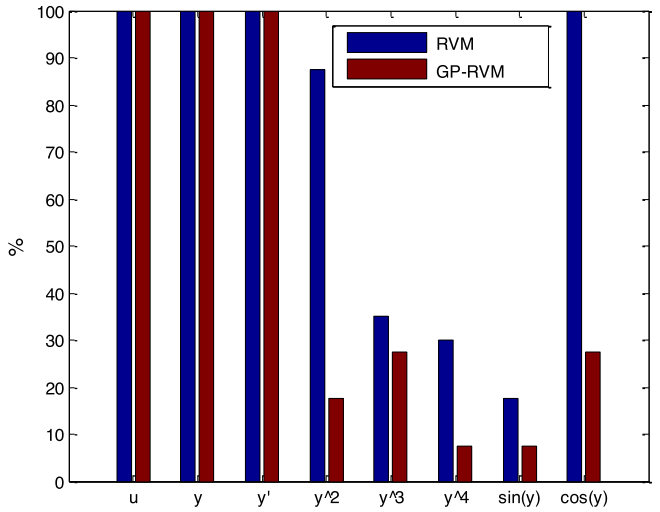

Fig. 8. Percentage of each candidate model form picked by the methods for the training datasets (blue bar: RVM method; magenta bar: GP-RVM method).

Integrating the measured acceleration to get displacement and velocity data may cause problems when the measurement noise is significant. It can also be challenging when considering the excitation as the system input only, in which case the system parameters will be coupled and it is not trivial to estimate system parameters through sparse Bayesian inference.

It should also be noted that in this example the nonlinearity is known to be a function of the displacement. If this is unknown, one way of applying the GP-RVM method is to set all system states as the input of the GP model. However, this may lead to potential identifiability issues since now the GP model will become more flexible and more likely to take over the role of the system model. In this case, balancing the explanatory power between the system model and the GP model can be challenging and it would form a potential topic for future work.

7.3. Silverbox benchmark test

The applicability of the GP-RVM method to real data are investigated in this example. The investigated system is an electric circuit simulating a mass-spring system with nonlinearity introduced by a position dependent stiffness term, forming the Silverbox benchmark dataset [26]. The system's behaviour is similar to a Duffing oscillator following Eq. (39) (the system investigated in Section 7.2). Details about the experimental configurations can be found in [26]. The true system parameter values are unknown and the major focus here is to investigate if the GP-RVM method can pick the correct assumed model components of the system and also assess its model performance.

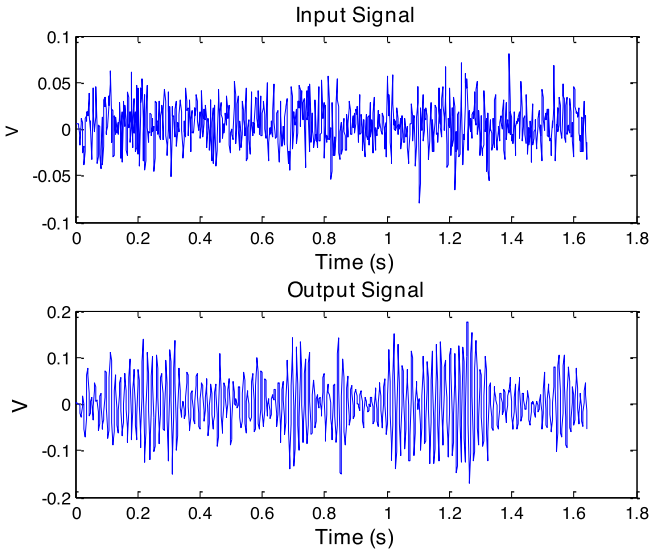

Fig. 9. Input and output signal of Silverbox test data set.

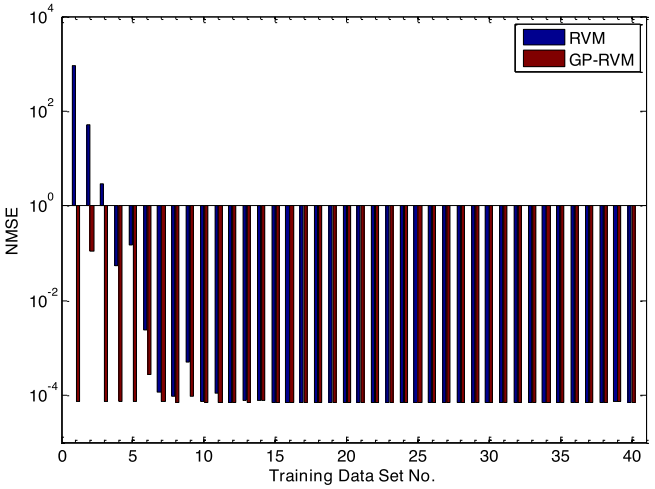

Fig. 10. NMSEs of the mean model prediction from the RVM and GP-RVM methods for the Silverbox test dataset (blue bar: RVM method; magenta bar: GP-RVM method).

The training data used in this example were measured by exciting the system with a Gaussian white noise (filtered by a 9th order discrete time Butterworth filter with a cut-off frequency of 200 Hz) with increasing amplitude. Both the input and output data (40000 samples) were recorded with a sampling rate of $10^7/2^{14} \approx 610.35$Hz. Fig. 7 shows the input and output data from the silver box system. The initial data were chopped into 40 non-overlapping datasets (each contains 1000 samples) for analysis.

In order to apply the equation discovery methods, all of the system states need to be obtained. Numerical differentiation has been conducted to get the first and second derivatives of the output measurement. Similar to Section 7.2, considering the second derivatives of the output measurement as the system response of interest, the system then can be rewritten as

$$\ddot{y} = \boldsymbol{\Phi}\boldsymbol{\theta} \tag{44}$$

where

$$\boldsymbol{\Phi} = \begin{bmatrix} u & y & \dot{y} & y^3 \end{bmatrix} \tag{45}$$

$$\boldsymbol{\theta} = \begin{bmatrix} \frac{1}{m} & \frac{a}{m} & \frac{c}{m} & \frac{b}{m} \end{bmatrix}^T \tag{46}$$

Assuming that the model form of the position dependent stiffness is unknown, the candidate design matrix used in this example are

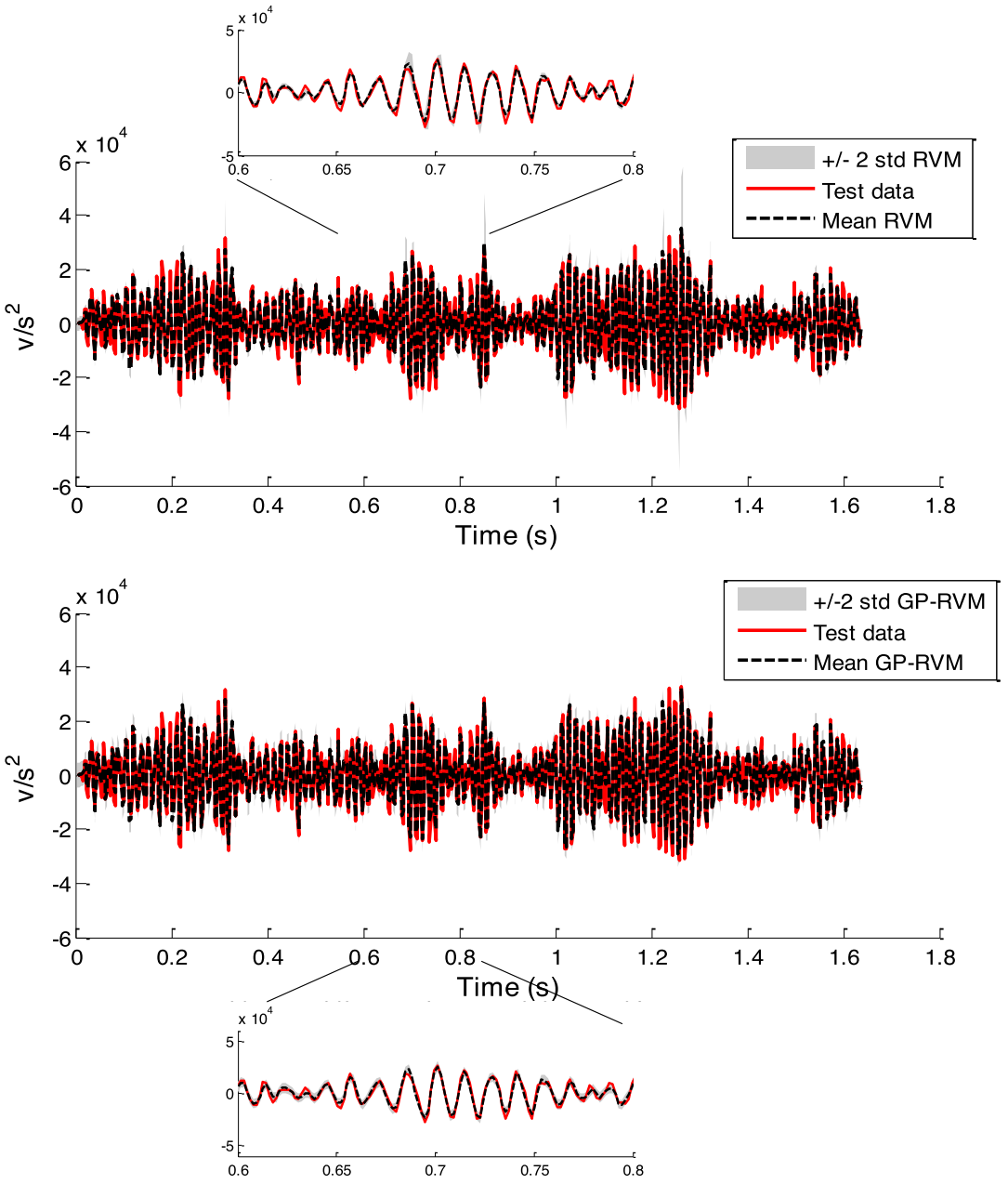

Fig. 11. Comparison of the Silverbox test data and model predictions from the RVM and GP-RVM methods (red solid line: simulated value; black dashed line: predicted mean of applied method; grey area: +/-2 standard deviation of applied method).

set as

$$\mathbf{\Phi}_{candi} = \begin{bmatrix} u & y & \dot{y} & y^2 & y^3 & y^4 & sin(y) & cos(y) \end{bmatrix} \tag{47}$$

Both the classic RVM method and the GP-RVM method are applied to infer the system's equation. Fig. 8 shows the percentage of each model component in the design matrix picked by these two methods after inference among these 40 training datasets. Both methods are able to pick out the linear components of the system (i.e., the first three model forms in the candidate design matrix) correctly. The percentages of the position dependent stiffness term y^3 involved in these two methods are about 30%. This is reasonable considering that the nonlinearity of the system is not significant, especially when the excitation amplitude is low. However, the RVM method is more likely to pick the erroneous model components in the design matrix, especially the y^2 and $cos(y)$ term. Compared to the

RVM method, these erroneous forms are less likely to be picked for the GP-RVM method. The proposed method can exclude misrepresentative model components when constructing the physics-based model, illustrating its capability of revealing more about the physics of the real system.

The model predictions from the RVM and GP-RVM methods are validated using an independent test data set; the response to a random odd multi-sine excitation. Details of the signal can be found in [26]. Fig. 9 shows the input and output signal of this data set from the Silverbox system. The NMSEs of the model predictions from the RVM and GP-RVM methods are plotted in Fig. 10. The NMSEs of the RVM models based on the first four training data set are relatively high, which are 930, 52.7, 2.96 and 0.05, respectively. This is mainly due to the low excitation amplitude in these training data sets, which causes inaccuracy in inferring the system parameters, particularly for the nonlinear component. The NMSEs of the RVM models based on the remaining training data sets are similar to those using the GP-RVM method, which are around 7×10^{-5}. Fig. 11 shows the test data set as well as the model predictions (mean and ± 2 posterior standard deviation) based on these two methods (trained using data set No.10) as a reference. It can be seen the model performances of these two methods are similar. However, it should be noted that the GP-RVM method can reveal more physics of the system with less misrepresentative model components involved.

8. Conclusions

An equation discovery method is proposed in this work that estimates system model parameters and captures model discrepancy simultaneously. The method is based on a sparse Bayesian formulation and allows potentially erroneous components from a set of candidate model terms to be removed during inference. The novelty of the proposed method is that it uses a Gaussian Process model to capture the model discrepancy. A Bayesian formulation is developed to encapsulate these two methods where the associated uncertainty is captured in the posterior distributions. An expectation maximisation (EM) algorithm has been adopted to perform inference, as the approach can be used to find the physically relevant maximum where the system model dominates. Numerical and experimental illustrative examples have been presented and it has been shown that compared to a classical sparse Bayesian inference method (i.e., the RVM), the proposed method can provide better parameter estimation results with less bias, especially when erroneous model forms occur in the design matrix. Furthermore, it has been shown that the model discrepancy due to missing physics can also be appropriately captured by the Gaussian process model. The results show that the EM algorithm is an appropriate choice for parameter inference since it can properly constrain the explanatory power of the GP model.

One assumption in this work is that the system model can be expressed as a linear combination of basis functions and system parameters (i.e., in the form shown in Eq. (2)) in order to apply the proposed method. In addition, the system parameters are assumed to be independent from each other and all the system inputs are known. Future work will seek to apply sparsity and Gaussian process models to more complex systems. Normally, not all the states of the dynamic systems are measured. For example, when only input force and acceleration responses are measured without knowing the displacement and velocity (which is usually the case in real applications), the latent states (i.e., displacement and velocity) should also be inferred when applying the proposed method. In this case, additional numerical sampling methods (e.g. Markov chain Monte Carlo sampling) shall be encapsulated into the proposed method for latent states inference. Illustrative examples in this work focused on single-degree-of-freedom dynamic systems. Applying the proposed method to multiple-degree-of-freedom systems will be investigated in future work, where the correlation between different degrees-of-freedom needs be properly accounted for. Similar to the latent state situation, numerical sampling methods may be required.

CRediT authorship contribution statement

Yi-Chen Zhu: Conceptualization, Methodology, Software, Validation, Formal analysis, Investigation, Writing - original draft, Writing - review & editing, Visualization. **Paul Gardner:** Conceptualization, Methodology, Writing - review & editing. **David J. Wagg:** Supervision, Project administration, Funding acquisition, Writing - review & editing. **Robert J. Barthorpe:** Supervision, Writing - review & editing. **Elizabeth J. Cross:** Supervision, Funding acquisition, Writing - review & editing. **Ramon Fuentes:** Conceptualization, Methodology.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

This paper is supported by UK Engineering & Physical Research Council Grant EP/R006768/1. The financial support is gratefully acknowledged. Elizabeth J. Cross also gratefully acknowledges the support of UK Engineering & Physical Research Council Grant EP/S001565/1.

References

[1] J.L. Beck, Bayesian system identification based on probability logic, Struct. Control Health Monitor. 17 (2010) 825–847, https://doi.org/10.1002/stc.424.

[2] H. Akaike, A New Look at the Statistical Model Identification, IEEE Trans. Autom. Control (1974), https://doi.org/10.1109/TAC.1974.1100705.
[3] L. Wasserman, Bayesian model selection and model averaging, J. Math. Psychol. (2000), https://doi.org/10.1006/jmps.1999.1278.
[4] D.J. Ewins, Modal testing : theory, practice, and application, Research Studies Press, Baldock, 2000.
[5] J.E. Mottershead, M.I. Friswell, Model updating in structural dynamics: a survey, J. Sound Vib. (1993), https://doi.org/10.1006/jsvi.1993.1340.
[6] R. Fuentes, N. Dervilis, K. Worden, E.J. Cross, Efficient parameter identification and model selection in nonlinear dynamical systems via sparse Bayesian learning, J. Phys.: Conf. Ser., IOP Publishing 1264 (1) (2019) 012050, https://doi.org/10.1088/1742-6596/1264/1/012050.
[7] R. Fuentes, R. Nayek, P. Gardner, N. Dervilis, T. Rogers, K. Worden, E.J. Cross, Equation discovery for nonlinear dynamical systems:a Bayesian viewpoint, Mech. Syst. Sig. Process. 154 (2021) 107528, https://doi.org/10.1016/j.ymssp.2020.107528.
[8] G.E.P. Box, N.R. Draper, Empirical model-building and response surfaces, John Wiley & Sons, 1987.
[9] M.C. Kennedy, A. O'Hagan, Bayesian calibration of computer models, J. R. Stat. Soc.: Ser. B (Statistical Methodology) 63 (2001) 425–464, https://doi.org/10.1111/1467-9868.00294.
[10] J.B. Nagel, B. Sudret, A unified framework for multilevel uncertainty quantification in Bayesian inverse problems, Probab. Eng. Mech. (2016), https://doi.org/10.1016/j.probengmech.2015.09.007.
[11] C. Li, S. Mahadevan, Role of calibration, validation, and relevance in multi-level uncertainty integration, Reliab. Eng. Syst. Saf. (2016), https://doi.org/10.1016/j.ress.2015.11.013.
[12] M.J. Bayarri, J.O. Berger, R. Paulo, J. Sacks, J.A. Cafeo, J. Cavendish, C.-H. Lin, J. Tu, A framework for validation of computer models, Technometrics 49 (2) (2007) 138–154, https://doi.org/10.1198/004017007000000092.
[13] D. Higdon, J. Gattiker, B. Williams, M. Rightley, Computer model calibration using high-dimensional output, J. Am. Stat. Assoc. 103 (482) (2008) 570–583, https://doi.org/10.1198/016214507000000888.
[14] E. Simoen, G. De Roeck, G. Lombaert, Dealing with uncertainty in model updating for damage assessment: a review, Mech. Syst. Sig. Process. 56–57 (2015) 123–149, https://doi.org/10.1016/j.ymssp.2014.11.001.
[15] P.D. Arendt, D.W. Apley, W. Chen, Quantification of model uncertainty: calibration, model discrepancy, and identifiability, J. Mech. Des. Trans. ASME 341 (2012), https://doi.org/10.1115/1.4007390.
[16] J. Brynjarsdóttir, A. Ohagan, Learning about physical parameters: The importance of model discrepancy, Inverse Prob. 30 (2014), https://doi.org/10.1088/0266-5611/30/11/114007.
[17] C.E. Rasmussen, C.K.I. Williams, Gaussian processes for machine learning, Int. J. Neural Syst. 14 (2004) 69–106, https://doi.org/10.1142/S0129065704001899.
[18] P.D. Arendt, D.W. Apley, W. Chen, A preposterior analysis to predict identifiability in the experimental calibration of computer models, IIE Trans. (Institute of Industrial Engineers). 48 (2016) 75–88, https://doi.org/10.1080/0740817X.2015.1064554.
[19] P.D. Arendt, W. Chen, D.W. Apley, Improving identifiability in model calibration using multiple responses, Proceedings of the ASME Design Engineering Technical Conference (2011), https://doi.org/10.1115/DETC2011-48623.
[20] P. Gardner, T.J. Rogers, C. Lord, R.J. Barthorpe, Learning of model discrepancy for structural dynamics applications using Bayesian history matching, J. Phys. Conf. Ser. (2019), https://doi.org/10.1088/1742-6596/1264/1/012052.
[21] M.E. Tipping, Sparse bayesian learning and the relevance vector machine, J. Mach. Learn. Res. (2001), https://doi.org/10.1162/15324430152748236.
[22] A.P. Dempster, N.M. Laird, D.B. Rubin, Maximum likelihood from incomplete data via the EM algorithm, J. Roy. Stat. Soc.: Ser. B (Methodol.) (1977), https://doi.org/10.1111/j.2517-6161.1977.tb01600.x.
[23] R. Tibshirani, Regression Shrinkage and Selection Via the Lasso, J. Roy. Stat. Soc.: Ser. B (Methodol.) (1996), https://doi.org/10.1111/j.2517-6161.1996.tb02080.x.
[24] J.O. Berger, Statistical decision theory and Bayesian analysis, Springer Science & Business Media, 2013.
[25] C.A. Micchelli, Y. Xu, H. Zhang, Universal kernels, J. Mach. Learn. Res. (2006).
[26] T. Wigren, J. Schoukens, Three free data sets for development and benchmarking in nonlinear system identification, in: 2013 European Control Conference, ECC 2013, 2013. 10.23919/ecc.2013.6669201.