
Article:
Gardner, P. orcid.org/0000-0002-1882-9728, Rogers, T.J. orcid.org/0000-0002-3433-3247, Lord, C. orcid.org/0000-0002-2470-098X et al. (1 more author) (2021) Learning model discrepancy: A Gaussian process and sampling-based approach. Mechanical Systems and Signal Processing, 152. 107381. ISSN 0888-3270

https://doi.org/10.1016/j.ymssp.2020.107381

Mechanical Systems and Signal Processing 152 (2021) 107381

Contents lists available at ScienceDirect

Mechanical Systems and Signal Processing

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

Learning model discrepancy: A Gaussian process and sampling-based approach

P. Gardner *, T.J. Rogers, C. Lord, R.J. Barthorpe

Dynamics Research Group, Department of Mechanical Engineering, University of Sheffield, Sheffield S1 3JD, UK

ARTICLE INFO

Article history:
Received 31 March 2020
Received in revised form 16 October 2020
Accepted 23 October 2020
Available online 10 December 2020

Keywords:
Model discrepancy
Gaussian process regression
Importance sampling
Bayesian history matching

ABSTRACT

Predicting events in the real world with a computer model (*simulator*) is challenging. Every simulator, to varying extents, has *model discrepancy*, a mismatch between real world observations and the simulator (given the 'true' parameters are known). Model discrepancy occurs for various reasons, including simplified or missing physics in the simulator, numerical approximations that are required to compute the simulator outputs, and the fact that assumptions in the simulator are not generally applicable to all real world contexts. The existence of model discrepancy is problematic for the engineer as performing calibration of the simulator will lead to biased parameter estimates, and the resulting simulator is unlikely to accurately predict (or even be valid for) various contexts of interest. This paper proposes an approach for inferring model discrepancy that overcomes non-identifiability problems associated with jointly inferring the simulator parameters along with the model discrepancy. Instead, the proposed procedure seeks to identify model discrepancy given some parameter distribution, which could come from a 'likelihood-free' approach that considers the presence of model discrepancy during calibration, such as Bayesian history matching. In this case, model discrepancy is inferred whilst marginalising out the uncertain simulator outputs via a sampling-based approach, therefore better reflecting the 'true' uncertainty associated with the model discrepancy. Verification of the approach is performed before a demonstration on an experiential case study, comprising a representative five storey building structure.

1. Introduction

Computer models (herein defined as *simulators*) never perfectly reflect the real world. This is due to the existence of *model discrepancy*, which will always be present to varying degrees and for a multitude of reasons, e.g. missing physics, numerical approximations, simplified assumptions, etc. This mismatch between simulator and real world — even if the 'true' parameters of the simulator are known — is problematic as performing calibration will lead to biased parameter estimates and predictions from the simulator are unlikely to be accurate (or, in some cases, even valid).

Several methods have been proposed for inferring model discrepancy [1–8], with most forming a Bayesian hierarchical model to solve a joint parameter and model discrepancy inference problem [1–7]. However, this type of approach is known to suffer from non-identifiability issues [4–7], leading to spurious estimations of both the parameter and model discrepancy distributions, even if the mean predictive outputs are accurate. These non-identifiability issues are caused, in particular, by an uninformative prior structure for the model discrepancy, caused by modelling the term with an unconstrained Gaussian

* Corresponding author.

https://doi.org/10.1016/j.ymssp.2020.107381
0888-3270/

process (GP) [1–7]; although this can be made even worse by uninformative priors on the parameters [6]. However, the reason the joint inference approach uses a Gaussian process is because little (if anything) is known about model discrepancy *a priori*, as by definition it involves some level of epistemic uncertainty. Consequently, the joint hierarchical approach utilises Gaussian processes as they offer a natural, Bayesian model over unknown functions, and are a flexible nonparametric method for performing regression (where flexible refers to the fact a GP can fit any arbitrary function well [9]). The main issue with the joint inference approach arises as 'bad' parameter estimates can be adequately compensated for by the Gaussian process model discrepancy term, due to the GPs ability to model any arbitrary discrepancy well, leading to a rather flat likelihood that is insensitive to changes in the parameters; making it a *bad* choice in likelihood. A potential remedy to this problem is sometimes available by specifying (if known) more informative prior parameter distributions, essentially enforcing that the prior dominates in the posterior, when the likelihood is insensitive [6]. Another technique tries to improve the likelihood by applying constraints to the Gaussian process, e.g. it must be positive at a particular input [6]. However, both improved specifications of the priors, and additional constraints on the GP, are often difficult to obtain *a priori*, as this involves obtaining more knowledge about the parameters or the model discrepancy, and still leaves the problem that fundamentally the likelihood does not accurately model reality.

In response to these issues with the joint approach, this paper seeks to develop an alternative method for inferring model discrepancy. Although the approach outlined in this paper is generally applicable to scenarios where some parameter distribution is available for the simulator parameters, the technique is specifically designed to accompany a *'likelihood-free'* approach to the calibration problem. In this scenario, calibration is performed via a 'likelihood-free' method, rather than using a 'true' likelihood function. Model discrepancy is instead introduced as some notation of distance between the simulator and the real world [10–16,8]. It is argued that this is a more appropriate way of specifying a level of epistemic uncertainty than using a formal likelihood function [10–16,8]. Methods such as Bayesian history matching [8] take this approach, and can be seen as forming a decoupled two stage solution to the problem of inferring simulator parameters and model discrepancy, i.e. infer plausible simulator parameters using the 'likelihood-free' approach, and then infer the model discrepancy given the calibrated parameter estimates. The two stage process has the benefit of decoupling the inference problem, meaning that non-identifiability issues, resulting from the GP model, are removed.

As stated, an additional benefit of the proposed approach, is that it can be used even if the viewpoint is taken that *any* calibration procedure will lead to biased parameter estimates, and therefore the best course of action is to better elicit the simulator parameters from physics or experimentation directly; given they can be identified without the use of the simulator in question. The argument here is that knowing the 'true' parameters from the physics (rather than calibration) will allow more robust extrapolation. However, the problem still remains that model discrepancy is present, and even given that the physically 'true' parameters exist and are known, there is no guarantee (in fact it is very unlikely) that the simulator will predict a given real world context well without inferring the functional form of the model discrepancy.

Both the 'likelihood-free' decoupled parameter and model discrepancy inference procedure, and fixing (physical) parameter estimates and inferring model discrepancy, can be seen as requiring the same step, i.e. given some estimates of the simulator parameters what is the model discrepancy?.

This paper proposes a novel method for inferring model discrepancy given *any* arbitrary set of samples of the parameter distribution (whether inferred via a likelihood-free approach or elicited). As a result, the engineer is better able to understand the mismatch between the simulator and real world, enabling them to more informatively target improvements to the simulator. The proposed approach marginalises out the uncertain simulator outputs whilst inferring the model discrepancy as a Gaussian process regression model via a sampling-based technique. As a result, model discrepancy is inferred given all valid parameter values (from the parameter distribution), providing a more accurate estimation of the uncertainty associated with the model discrepancy.

The outline of this paper is as follows. The approach for model discrepancy inference from some given parameter samples is stated, showing the Gaussian process and sampling-based formulation in Section 2. A 'likelihood-free' method for obtaining the calibrated parameter distribution, Bayesian history matching, is introduced in Section 3. Next, two numerical case studies are provided in Section 4, verifying the approach, with the first numerical example benchmarking the proposed approach against the hierarchical Bayesian method. Finally, an experimental case study is provided in Section 5, demonstrating the effectiveness of the method, before conclusions and future work are presented in Section 6.

2. Model discrepancy inference

Model discrepancy is defined as the mismatch between simulator predictions and real world observations, given that the 'true' parameters of the simulator are known. Typically, this means that model discrepancy is assumed to be some additive, corrective function [1] — with the simulator considered as a complicated, black-box model and therefore invasive model discrepancy terms, modifying the underlying simulator equations, are avoided. However if more information about the cause of model discrepancy is known, i.e. it affects some internal states [17], and the simulator can be modified in an invasive manner, then it should be modelled more appropriately. Following these assumptions (as first proposed in [1]), model discrepancy can be mathematically defined for a single output as,

$$z(X) = \eta(X, \boldsymbol{\theta}) + \delta(X) + \boldsymbol{e} \tag{1}$$

where $\boldsymbol{z} \in \mathbb{R}^{N\times 1}$ are real world observations for a set of inputs $X \in \mathbb{R}^{N\times dx}$. The simulator $\eta(\cdot,\cdot)$ depends on a set of inputs X and parameters $\boldsymbol{\theta} \in \mathbb{R}^{M\times 1}$, whereas the model discrepancy term $\delta(\cdot)$ is assumed to only be dependent on the inputs X. Finally, $\boldsymbol{e} \in \mathbb{R}^{N\times 1}$ is assumed Gaussian additive noise. The decision about what the inputs X and parameters $\boldsymbol{\theta}$ of a simulator are will be application specific, where generally, the inputs X are chosen to reflect the measured outputs $\boldsymbol{z}$ and the parameters $\boldsymbol{\theta}$ are additional variables in the simulator that could be tuned. For example, the inputs X could be variables that drive a physical process, such as a force or crack length, or independent variables like spectral lines or frequency bins, with the parameters $\boldsymbol{\theta}$ being, for example, material properties, or even mass, stiffness and damping coefficients, such as in direct model updating [18].

The main difference between the proposed method and existing techniques, such as in [1], is that the parameters $\boldsymbol{\theta}$ are identified before the functional model discrepancy term $\delta(\cdot)$ is inferred; rather than jointly inferring both $\boldsymbol{\theta}$ and $\delta(\cdot)$. This decoupling assumption is made as the alternative joint inference problem is susceptible to non-identifiability issues, caused by modelling model discrepancy $\delta(\cdot)$ with a Gaussian process whilst inferring the parameters [4,6]. In fact, this modelling choice in the joint approach, leads to a rather poor likelihood, where even 'bad' parameter samples are given a high probability in the likelihood function [6], due to the ability of the GP to model any arbitrary function well [9], making the likelihood insensitive. Instead, by decoupling the problem, inferring the parameters and then the model discrepancy, non-identifiability issues and problems with the likelihood can be overcome.

In order to decouple the parameter and model discrepancy inferences, the calibration method must be able to account for model discrepancy in another way. 'Likelihood-free' approaches, such as Bayesian history matching, offer such a technique. These methods incorporate model discrepancy though a notion of distance, removing issues associated with defining a specific likelihood, whilst approximating the parameter posterior distribution $p(\boldsymbol{\theta}|\boldsymbol{z})$. Once obtained, model discrepancy can be inferred using a Gaussian process model — without affecting the parameter posterior distribution. However, the Gaussian process model must be constructed from the uncertain simulator outputs $p(\boldsymbol{y}|X,\boldsymbol{\theta})$ to the real world observations $\boldsymbol{z}$. Unfortunately, it is not possible in closed-form to create a Gaussian process from uncertain inputs, meaning that a sampling-based solution is required.

It is noted that in the joint inference method, the parameters $\boldsymbol{\theta}$ are inferred, given an empirical Bayes estimate of the model discrepancy hyperparameters, requiring the Gaussian process to be conditioned on the parameter prior distributions, e.g. $\int p(\boldsymbol{z}|\boldsymbol{y},\boldsymbol{\phi},\boldsymbol{\theta})p(\boldsymbol{\theta})\mathrm{d}\boldsymbol{\theta}$ [1–7] — this can bias the inferred model discrepancy. This conditioning, typically leads to a restricted choice in prior distributions $p(\boldsymbol{\theta})$ that are conjugate with a Gaussian process, such as a Gaussian [1] or uniform [4] distribution; where non-conjugate priors require an additional expensive sampling procedure, on top of the parameter estimation, which is performed in low dimensions by a quadrature approach, and in high dimensions by another sampling step [1]. These issues are removed by considering the decoupled approach proposed in this paper.

By decoupling the inference procedure, the model discrepancy method can also be applied in scenarios where the parameter distribution is obtained by some elicitation process or experimentation. This makes the technique more generally applicable to a wider range of problems outside of those originally considered by the joint approach.

The proposed method assumes that some parameter distribution $p(\boldsymbol{\theta}|\boldsymbol{z})$ has been obtained from a 'likelihood-free' calibration method, or $p(\boldsymbol{\theta})$ has been acquired from some elicitation process; where, for simplicity, $p(\boldsymbol{\theta})$ will be used to denote a generic parameter distribution. The approach then seeks to find the additive model discrepancy (and noise) term modelled using Gaussian process (GP) regression. As the simulator outputs, given the parameter distribution, are uncertain, a sampling-based approach is used to marginalise out the simulator outputs, meaning (potentially calibrated and) bias-corrected model predictions can be made, reflecting the uncertainty from the parameter distribution. A brief outline of the approach is as follows:

- Obtain N_s samples from the parameter distribution i.e. for the jth sample, $\boldsymbol{\theta}^{(j)} \sim p(\boldsymbol{\theta})$.
- Propagate those N_s samples through the simulator to obtain N_s simulator output (denoted $\boldsymbol{y} \in \mathbb{R}^{N\times 1}$) samples i.e. $\boldsymbol{y}^{(j)} = \eta\left(X,\boldsymbol{\theta}^{(j)}\right)$.
- Learn a GP mapping for each of the N_s output samples $\boldsymbol{y}^{(j)}$ to a set of training observations $\boldsymbol{z}$ i.e. $\mathcal{GP}^{(j)} : \left\{\boldsymbol{y}^{(j)}, X\right\} \rightarrow \boldsymbol{z}$ and obtain a weight w_j for each regression model — the weights will be formally introduced in Section 2.2.
- Calculate the weighted average of the set of GP regression models generating a bias-corrected model prediction $p(\boldsymbol{z}_*|X_*,\boldsymbol{\theta},\mathcal{D})$.

It is noted that in the case where the simulator is computationally expensive to evaluate, a more computationally efficient emulator, or surrogate model [19], can be constructed. This efficient approximation can be sampled instead of the simulator in step two, where any emulator technique within the literature could be implemented [19–21]; in this paper a Gaussian process emulator is utilised.

2.1. Gaussian process regression

Model discrepancy is modelled in this paper by GP regression as it is a flexible, nonparametric tool, and because it has a Bayesian formulation allowing the uncertainty associated with the inferred functional form to be estimated [22,9]. These

properties, the ability to approximate any unknown function well whilst quantifying the uncertainty in the prediction, are useful as the functional form of the model discrepancy is unknown *a priori* and quantifying the uncertainty associated with this form may aid simulator developers in targeting improvements to their computer model. In addition, by decoupling the inference problem, the choice of modelling model discrepancy with a Gaussian process no longer affects the likelihood in the parameter inference stage, making it a more suitable assumption.

The model discrepancy term is assumed in Eq. (1) to be additive, meaning it can be formed as a map from the simulator outputs $\boldsymbol{y}$ and inputs X to the observational data $\boldsymbol{z}$, $\mathcal{GP}:\{\boldsymbol{y},X\}\rightarrow\boldsymbol{z}$, where the inferred model discrepancy GP can be related back to the inputs as $\delta(X)$. For this reason GP regression is introduced in this section with the simulator outputs $\boldsymbol{y}$ being part of the inputs to the GP along with X, and where the noisy observations $\boldsymbol{z}$ are the outputs of the GP.

A Gaussian process states a prior distribution over a latent function $f(\boldsymbol{y},X)$ (of the noisy function $z(\boldsymbol{y},X)=f(\boldsymbol{y},X)+\boldsymbol{e}$),

$$\boldsymbol{f}\sim\mathcal{GP}\big(m(\boldsymbol{y},X),k((\boldsymbol{y},X),(\boldsymbol{y}',X'))\big) \tag{2}$$

where $\mathcal{GP}(\cdot,\cdot)$ is a Gaussian process, with a mean function $m(\cdot)$ and covariance function $k(\cdot,\cdot)$ which define the prior belief about the types of possible functions that could model the function $\boldsymbol{f}\in\mathbb{R}^{N\times 1}$. Here a zero mean function is assumed, i.e. $m(\boldsymbol{y},X)=\boldsymbol{0}$, although it is trivial to add a non-zero mean function. The covariance function defines the correlation between any two points in the input space (hence being a function of $(\boldsymbol{y},X)$ and $(\boldsymbol{y}',X')$) in a Reproducing Kernel Hilbert Space (RKHS) and is fully specified by a set of hyperparameters $\boldsymbol{\phi}$, i.e. $K=k(\cdot,\cdot;\boldsymbol{\phi})$. The covariance function utilised in this paper is a Matérn 3/2 covariance function, as it is ideal for modelling relatively 'smooth' real world functions,[1] as it is (3/2–1) times mean square differentiable [23], and is defined as,

$$K_{f,f}=k((\boldsymbol{y},X),(\boldsymbol{y}',X'))=\sigma_f^2\Big(1+\sqrt{3}r_y\exp\Big(\sqrt{3}r_y\Big)\Big)\Big(1+\sqrt{3}r_x\exp\Big(\sqrt{3}r_x\Big)\Big) \tag{3}$$

where,

$$r_y=\sqrt{(\boldsymbol{y}-\boldsymbol{y}')^{\mathsf{T}}L_y^{-1}(\boldsymbol{y}-\boldsymbol{y}')} \tag{4}$$

and,

$$r_x=\sqrt{(X-X')^{\mathsf{T}}L_x^{-1}(X-X')} \tag{5}$$

where $K_{f,f}\in\mathbb{R}^{N\times N}$ is the covariance matrix for inputs[2] $\boldsymbol{y}\in\mathbb{R}^{N\times 1}$ and $X\in\mathbb{R}^{N\times d_x}$, σ_f^2 is the signal variance hyperparameter, and $L_y=\mathrm{diag}(l_{y1},\ldots,l_{yd})$ and $L_x=\mathrm{diag}(l_{x1},\ldots,l_{xd})$ are lengthscale hyperparameters (making the covariance function an automatic relevance determination prior, i.e. it reduces the effect of redundant inputs). The covariance structure here separates out $\boldsymbol{y}$ and X allowing them to have an independent relationship with the outputs $\boldsymbol{z}$. The hyperparameter vector for the Matérn 3/2 covariance function is therefore $\boldsymbol{\phi}=\{\sigma_f,L_y,L_x\}$. It is noted that the notation $K_{f,*}=k((\boldsymbol{y},X),(\boldsymbol{y}_*,X_*))$ is used, where f indicates training and $*$ test data. In order to make predictions, the joint Gaussian distribution is formed between a set of training $\mathcal{D}=\{\{\boldsymbol{y},X\},\boldsymbol{z}\}$ and testing data $\{\{\boldsymbol{y}_*,X_*\},\boldsymbol{z}_*\}$, assuming a Gaussian likelihood,

$$\begin{bmatrix}\boldsymbol{z}\\ \boldsymbol{z}_*\end{bmatrix}\sim\left(\begin{bmatrix}\boldsymbol{0}\\ \boldsymbol{0}\end{bmatrix},\begin{bmatrix}K_{f,f}+\mathbb{I}_f\sigma_n^2 & K_{f,*}\\ K_{*,f} & K_{*,*}+\mathbb{I}_*\sigma_n^2\end{bmatrix}\right) \tag{6}$$

where $\mathbb{I}_f\in\mathbb{R}^{N\times N}$ and $\mathbb{I}_*\in\mathbb{R}^{N_*\times N_*}$ are identity matrices and σ_n^2 is a Gaussian noise variance. Following standard Gaussian conditionals, the posterior for a Gaussian process regression model can be formed as,

$$p(\boldsymbol{z}_*|\boldsymbol{y}_*,X_*,\mathcal{D},\boldsymbol{\phi})=\mathcal{N}(\mathbb{E}(\boldsymbol{z}_*),\mathbb{V}(\boldsymbol{z}_*)) \tag{7a}$$

$$\mathbb{E}(\boldsymbol{z}_*)=K_{*,f}\big(K_{f,f}+\mathbb{I}_f\sigma_n^2\big)^{-1}\boldsymbol{z} \tag{7b}$$

$$\mathbb{V}(\boldsymbol{z}_*)=K_{*,*}+\mathbb{I}_*\sigma_n^2-K_{*,f}\big(K_{f,f}+\mathbb{I}_f\sigma_n^2\big)^{-1}K_{f,*}. \tag{7c}$$

Conventionally, GP models are inferred by taking a type-II maximum likelihood approach [9], i.e. finding the hyperparameters that maximise the marginal likelihood, leading to an empirical Bayes estimate of the hyperparameters $\hat{\boldsymbol{\phi}}$. By combining the noise variance with the set of covariance function hyperparameters, i.e. $\boldsymbol{\phi}=\{\sigma_f,L_y,L_x,\sigma_n^2\}$, the empirical Bayes estimates for the set of hyperparameters may be found through optimisation, — here a global optimisation approach is used, specifically via quantum particle swarm [24] — by minimising the negative log marginal likelihood,

$$\hat{\boldsymbol{\phi}}=\arg\min_{\boldsymbol{\phi}}\{-\log p(\boldsymbol{z}|\boldsymbol{y},X,\boldsymbol{\phi})\} \tag{8}$$

where,

[1] It is noted that many other choices of covariance function exist and the reader is referred to [9] for more options.

[2] However, if the simulator produces a multivariate output then the inputs to the GP model may be d-dimensional i.e. $Y\in\mathbb{R}^{N\times d}$, meaning the covariance function is formed from Y.

$$-\log p(\boldsymbol{z}|\boldsymbol{y},X,\boldsymbol{\phi}) = \frac{1}{2}\boldsymbol{z}^{\mathsf{T}}\left(K_{ff}+\mathbb{I}\sigma_n^2\right)^{-1}\boldsymbol{z} + \frac{1}{2}\log|K_{ff}+\mathbb{I}\sigma_n^2| + \frac{N}{2}\log 2\pi. \tag{9}$$

It is noted that a fully Bayesian analysis would require marginalisation of the hyperparameters, which is not solvable in closed-form due to the dependence of the hyperparameters in the covariance function. However, the fully Bayesian solution may be inferred from a sampling-based approach [25–27] and is explored in the following section.

2.2. Sampling-based approach

The method outlined in this paper utilises GP regression in identifying the map $\mathcal{GP}:\{\boldsymbol{y},X\}\rightarrow\boldsymbol{z}$, and therefore inferring the model discrepancy term $\delta(X)$. However, the output from a simulator $\boldsymbol{y}$ will typically be uncertain, i.e. $p(\boldsymbol{y}|X,\boldsymbol{\theta})$, arising from parametric uncertainty in $p(\boldsymbol{\theta})$. However, Gaussian process regression cannot be solve in closed-form for uncertain inputs, and even though the simulator inputs X are deterministic, the simulator outputs $\boldsymbol{y}$ are uncertain. To create bias-corrected predictions that account for this parametric uncertainty, the simulator outputs $\boldsymbol{y}_*$ must be integrated out, forming the following integral,

$$p(\boldsymbol{z}_*|X_*,\boldsymbol{\theta},\mathcal{D},\boldsymbol{\phi}) = \int p(\boldsymbol{z}_*|\boldsymbol{y}_*,X_*,\mathcal{D},\boldsymbol{\phi})p(\boldsymbol{y}_*|X_*,\boldsymbol{\theta})\mathrm{d}\boldsymbol{y}_* \tag{10}$$

where $p(\boldsymbol{z}_*|X_*,\boldsymbol{\theta},\mathcal{D},\boldsymbol{\phi})$ is the bias-corrected predictive output, $p(\boldsymbol{y}_*|X_*,\boldsymbol{\theta})$ is the simulator prediction at test inputs X_* which is conditioned on the parameter distribution $p(\boldsymbol{\theta})$. It is noted that in previous work [8] the bias-corrected outputs have been approximated using the *maximum a posteriori* (*MAP*) estimate of the parameters (and an empirical Bayes estimate of the hyperparameters $\hat{\boldsymbol{\phi}}$) meaning $p\left(\boldsymbol{z}_*|X_*,\boldsymbol{\theta}^{MAP},\mathcal{D},\hat{\boldsymbol{\phi}}\right)$. However, this will not account for the complete parametric uncertainty from $p(\boldsymbol{\theta})$ and may result in a biased estimate of the model discrepancy.

The proposed method seeks to approximate Eq. (10) via a sampling-based approach — specifically from an importance sampling viewpoint. Importance sampling is a technique for obtaining unbiased estimates of expectation integrals [28], such as in Eq. (10), and can be generalised as,

$$\mathbb{E}_p(f(x)) = \int f(x)p(x)\mathrm{d}x = \int q(x)\frac{f(x)p(x)}{q(x)}\mathrm{d}x = \mathbb{E}_q\left(\frac{f(X)p(X)}{q(X)}\right) \tag{11}$$

where $f(x)$ is a function, $p(x)$ the nominal distribution over the variable x and $q(x)$ is the proposal distribution, where $X\sim q$ are independent draws from the proposal distribution. The expectation in Eq. (11) can be formed as,

$$\mathbb{E}_p(f(x)) \approx \frac{1}{N}\sum_{i=1}^{N}\frac{f(X)p(X)}{q(X)} = \frac{1}{N}\sum_{i=1}^{N}f(X)w(X) \tag{12}$$

where N are the number of samples and $w(X)=p(X)/q(X)$ are the importance weights [28].

Eq. (10) can be approximated by setting the nominal and proposal distributions equal to the simulator output predictive distribution $p(\boldsymbol{y}_*|X_*,\boldsymbol{\theta})$. This means that N_s samples can be obtained from the parameter distribution, i.e. $\boldsymbol{\theta}^{(j)}\sim p(\boldsymbol{\theta})$ and propagated through the simulator to obtain output samples $\boldsymbol{y}_*^{(j)}\sim p\left(\boldsymbol{y}_*|X_*,\boldsymbol{\theta}^{(j)}\right)=q\left(\boldsymbol{y}_*|X_*,\boldsymbol{\theta}^{(j)}\right)$, meaning that the weight for each sample equals one, i.e. $w(\boldsymbol{y}_*{}^{(j)})=1$ (this effectively is the same as approximating the integral via Monte Carlo sampling, however the language of importance sampling will useful later in this section). The predictive equation $p(\boldsymbol{z}_*|X_*,\boldsymbol{\theta},\mathcal{D},\boldsymbol{\phi})$ can now be approximated using the set of weights and Gaussian process predictions for each sample, and the laws of total expectation and variance,

$$p(\boldsymbol{z}_*|X_*,\boldsymbol{\theta},\mathcal{D},\boldsymbol{\phi}) \approx \mathcal{N}(\mathbb{E}(\boldsymbol{z}_*\mid X_*,\boldsymbol{\theta},\mathcal{D},\boldsymbol{\phi}),\mathbb{V}(\boldsymbol{z}_*\mid X_*,\boldsymbol{\theta},\mathcal{D},\boldsymbol{\phi})) \tag{13a}$$

$$\mathbb{E}(\boldsymbol{z}_*\mid X_*,\boldsymbol{\theta},\mathcal{D},\boldsymbol{\phi}) = \frac{1}{N_s}\sum_{j=1}^{N_s}w^{(j)}\mathbb{E}\left(\boldsymbol{z}_*^{(j)}\right) \tag{13b}$$

$$\begin{aligned}\mathbb{V}(\boldsymbol{z}_*\mid X_*,\boldsymbol{\theta},\mathcal{D},\boldsymbol{\phi}) = {}& \tfrac{1}{N_s}\sum_{j=1}^{N_s}w^{(j)}\left(\mathbb{V}\left(\boldsymbol{z}_*^{(j)}\right)+\mathbb{E}\left(\boldsymbol{z}_*^{(j)}\right)\mathbb{E}\left(\boldsymbol{z}_*^{(j)}\right)^{\mathsf{T}}\right)\\ & -\mathbb{E}(\boldsymbol{z}_*\mid X_*,\boldsymbol{\theta},\mathcal{D},\boldsymbol{\phi})\mathbb{E}(\boldsymbol{z}_*\mid X_*,\boldsymbol{\theta},\mathcal{D},\boldsymbol{\phi})^{\mathsf{T}}\end{aligned} \tag{13c}$$

where $\mathbb{E}(\boldsymbol{z}_*)$ and $\mathbb{V}(\boldsymbol{z}_*)$ are the GP predictive mean and covariance from Eq. (7). The bias-corrected predictions are approximately Gaussian, given that they are formed from weighted averaged Gaussian processes. The method is outlined in Algo-

rithm 1. The main computational expense is in sampling the simulator outputs, which can be reduced by replacing the simulator with a computationally efficient emulator [19–21]. One problem with this approach is that the predictions are still dependent on a set of hyperparameters (that have been inferred from the GP associated with the parameter *MAP* estimates). However, these hyperparameters can also be marginalised out of the predictive equations using importance sampling, as discussed below.

Algorithm 1. Model discrepancy inference dependent on empirical Bayes estimates of $\boldsymbol{\phi}$

Training;

Optimise $\hat{\boldsymbol{\phi}} = \arg\min_{\boldsymbol{\phi}} \left\{ -\log p\left(\boldsymbol{z} \mid \boldsymbol{y}, X, \boldsymbol{\phi}, \boldsymbol{\theta}^{MAP}\right) \right\}$ ▷ Obtain hyperparameter estimates $\hat{\boldsymbol{\phi}}$ from simulator output $p\left(\boldsymbol{y} \mid X, \boldsymbol{\theta}^{MAP}\right)$ at $\boldsymbol{\theta}^{MAP}$

for $j = 1 : N_s$ **do**

  Sample $\boldsymbol{\theta}^{(j)} \sim p\left(\boldsymbol{\theta}\right)$ ▷ Sample the parameter distribution

  $w^{(j)} = 1$ ▷ Set the weight for the jth sample to one

end for

Prediction;

for $j = 1 : N_s$ **do**

  Predict $p\left(\boldsymbol{y}_*^{(j)} \mid X_*, \boldsymbol{\theta}^{(j)}\right)$ ▷ Obtain simulator output sample $\boldsymbol{y}_*^{(j)}$ by propagating the parameter sample $\boldsymbol{\theta}^{(j)}$

  Predict $\mathbb{E}\left(\boldsymbol{z}_*^{(j)}\right)$ and $\mathbb{V}\left(\boldsymbol{z}_*{}^{(j)}\right)$ ▷ Make a prediction from the GP conditioned on $\hat{\boldsymbol{\phi}}$ for the jth sample

end for

Predict the approximation of $p\left(\boldsymbol{z}_* \mid X_*, \boldsymbol{\theta}, \mathcal{D}, \boldsymbol{\phi}\right)$ via Eq. (13)

The following integrals can be solved to generate a bias-corrected prediction not dependent on either the simulator outputs or GP hyperparameters,

$$p(\boldsymbol{z}_*|X_*, \boldsymbol{\theta}, \mathcal{D}) = \int \left(\int p(\boldsymbol{z}_*|\boldsymbol{y}_*, X_*, \mathcal{D}, \boldsymbol{\phi}) p(\boldsymbol{y}_*|X_*, \boldsymbol{\theta}) \mathrm{d}\boldsymbol{y}_* \right) p(\boldsymbol{\phi}|\mathcal{D}) \mathrm{d}\boldsymbol{\phi} \tag{14}$$

requiring the posterior of the hyperparameters $p(\boldsymbol{\phi}|\mathcal{D})$, which can also be approximated via importance sampling. By solving Eq. (14), rather than optimising the GP via a type-II maximum likelihood technique, the fully Bayesian solution of the hyperparameters can be acquired.

The posterior distribution of the hyperparameters $p(\boldsymbol{\phi}|\mathcal{D}) \propto p(\boldsymbol{z}|\boldsymbol{y}, X, \boldsymbol{\phi}) p(\boldsymbol{\phi})$ (where $\mathcal{D} = \{\{\boldsymbol{y}, X\}, \boldsymbol{z}\}$), can be approximated with importance sampling by setting the unnormalised nominal distribution as $p(\boldsymbol{z}|\boldsymbol{y}, X, \boldsymbol{\phi}) p(\boldsymbol{\phi})$. By keeping the proposal and nominal distributions for the parameters the same as in the first approach, and by setting the proposal distribution for the hyperparameters equal to the prior distribution, i.e. $\boldsymbol{\phi}^{(k)} \sim p(\boldsymbol{\phi}) = q(\boldsymbol{\phi})$, the weights for each of the N_ϕ hyperparameter samples are equal to $p\left(\boldsymbol{z}|\boldsymbol{y}, X, \boldsymbol{\phi}^{(k)}\right)$, i.e. the marginal likelihood of the GP model (in negative log-form in Eq. (9)). This formulation now allows both $\boldsymbol{y}_*$ and $\boldsymbol{\phi}$ to be integrated out with importance sampling forming $p(\boldsymbol{z}_*|X_*, \boldsymbol{\theta}, \mathcal{D})$ as,

Algorithm 2. Model discrepancy inference marginalising $\boldsymbol{\phi}$

Training;
for $j = 1 : N_s$ **do**
  Sample $\boldsymbol{\theta}^{(j)} \sim p(\boldsymbol{\theta})$ ▷ Sample the parameter distribution
  Predict $p\left(\boldsymbol{y}^{(j)} \mid X, \boldsymbol{\theta}^{(j)}\right)$ ▷ Obtain simulator output sample $\boldsymbol{y}^{(j)}$ by propagating the parameter sample $\boldsymbol{\theta}^{(j)}$
  for $k = 1 : N_\phi$ **do**
    Sample $\boldsymbol{\phi}^{(j,k)} \sim q(\boldsymbol{\phi})$ ▷ Sample the GP hyperparameter proposal distribution
    $\tilde{w}^{(j,k)} = p\left(\boldsymbol{z} \mid \boldsymbol{y}^{(j)}, X, \boldsymbol{\phi}^{(j,k)}, \boldsymbol{\theta}^{(j)}\right)$ ▷ Set the unnormalised weight as the GP marginal likelihood
  end for
end for
Normalise weights $w^{(j,k)} = \tilde{w}^{(j,k)} / \sum_{j=1}^{N_s} \sum_{k=1}^{N_\phi} \tilde{w}^{(j,k)}$ ▷ Normalise the importance weights

Prediction;
for $i = 1 : N_s$ **do**
  Predict $p\left(\boldsymbol{y}_*^{(j)} \mid X_*, \boldsymbol{\theta}^{(j)}\right)$ ▷ Obtain simulator output sample $\boldsymbol{y}_*^{(j)}$ by propagating the parameter sample $\boldsymbol{\theta}^{(j)}$
  for $k = 1 : N_\phi$ **do**
    Predict $\mathbb{E}\left(\boldsymbol{z}_*^{(j,k)}\right)$ and $\mathbb{V}\left(\boldsymbol{z}_*{}^{(j,k)}\right)$ ▷ Make a prediction from the GP for the jth kth sample
  end for
end for
Predict the approximation of $p(\boldsymbol{z}_* \mid X_*, \boldsymbol{\theta}, \mathcal{D})$ via Eq. (15)

$$p(\boldsymbol{z}_* | X_*, \boldsymbol{\theta}, \mathcal{D}) \approx \mathcal{N}(\mathbb{E}(\boldsymbol{z}_* \mid X_*, \boldsymbol{\theta}, \mathcal{D}), \mathbb{V}(\boldsymbol{z}_* \mid X_*, \boldsymbol{\theta}, \mathcal{D})) \tag{15a}$$

$$\mathbb{E}(\boldsymbol{z}_* \mid X_*, \boldsymbol{\theta}, \mathcal{D}) = \sum_{j=1}^{N_s} \sum_{k=1}^{N_\phi} w^{(j,k)} \mathbb{E}\left(\boldsymbol{z}_*^{(i,k)}\right) \tag{15b}$$

$$\begin{aligned} \mathbb{V}(\boldsymbol{z}_* \mid X_*, \boldsymbol{\theta}, \mathcal{D}) = & \sum_{j=1}^{N_s} \sum_{k=1}^{N_\phi} w^{(j,k)} \left(\mathbb{V}\left(\boldsymbol{z}_*^{(j,k)}\right) + \mathbb{E}\left(\boldsymbol{z}_*^{(j,k)}\right) \mathbb{E}\left(\boldsymbol{z}_*^{(j,k)}\right)^{\mathsf{T}}\right) \\ & - \mathbb{E}(\boldsymbol{z}_* \mid X_*, \boldsymbol{\theta}, \mathcal{D}) \mathbb{E}(\boldsymbol{z}_* \mid X_*, \boldsymbol{\theta}, \mathcal{D})^{\mathsf{T}} \end{aligned} \tag{15c}$$

where the overall procedure is outlined in Algorithm 2.

Is it noted that the approaches rely on importance sampling, which will suffer from the curse of dimensionality as the dimension of Y and $\boldsymbol{\phi}$ increases, as it will be less likely that a given sample will carry some meaningful weight. This issues can be mitigated by an adaptive approach [26], which is left for further research.

3. Obtaining the parameter distribution

As aforementioned, the proposed model discrepancy inference approach is primarily designed to accompany a 'likelihood-free' calibration process, where model discrepancy is account though a distance measure. These approaches allow for $p(\boldsymbol{\theta})$ to be obtained without the GP model discrepancy influencing the posterior parameter distribution, removing problems associated with non-identifiability. This section briefly introduces Bayesian history matching as one such approach that can be used in combination with the proposed procedure (for more details on Bayesian history matching the interested reader is referred to [8]).

3.1. Bayesian history matching

Bayesian history matching (BHM) is an approximate Bayesian approach for calibrating statistical models of the form in Eq. (1). The method seeks to determine whether parameter combinations are 'implausible', i.e. they are not likely to have produced the observations $\boldsymbol{z}$, based on a criteria such that the remaining non-implausible parameter space is identified; leading to an approximation of the posterior distribution $p(\boldsymbol{\theta}|\boldsymbol{z})$. The criteria for discarding implausible samples is a combination of an *implausibility metric* and a threshold T, where the implausibility metric accounts for model discrepancy though a notion of distance.

BHM assumes that the simulator is computationally expensive to evaluate, and hence replaces the simulator with a computationally efficient GP emulator $\eta(X,\boldsymbol{\theta}) \sim \mathcal{GP} : \{X,\boldsymbol{\theta}\} \rightarrow \boldsymbol{y}$. This replacement of the simulator for an emulator is possible as a GP estimates the uncertainty associated with the approximation through the predictive variance $\mathbb{V}(y(X,\boldsymbol{\theta}))$ from $\eta(X,\boldsymbol{\theta}) \approx \mathcal{N}(\mathbb{E}(y(X,\boldsymbol{\theta})), \mathbb{V}(y(X,\boldsymbol{\theta})))$ (formed in a similar manner to Eq. (7)). This means that parameter combinations are only discarded when the approximation is certain enough, given the other uncertainty in the implausibility metric. The implausibility metric assess the distance between the mean emulator prediction $\mathbb{E}(\boldsymbol{\eta}_*)$ and the observed data $\boldsymbol{z}$, weighted by several uncertainties,

$$I(X,\boldsymbol{\theta}) = \frac{|z(X) - \mathbb{E}(y(X,\boldsymbol{\theta}))|}{(V_o + V_m + \mathbb{V}(y(X,\boldsymbol{\theta})))^{1/2}} \tag{16}$$

where V_o and V_m are variances associated with observational and model discrepancy uncertainties.

BHM acts in an iterative manner as follows:

- A parameter space Θ_i is proposed and sampled, $\boldsymbol{\theta}^{(j)} \sim \Theta$.
- For each parameter sample $\boldsymbol{\theta}^{(j)}$ the implausibility $I(X,\boldsymbol{\theta})$ is assessed.
- Given the criteria $I\left(X,\boldsymbol{\theta}^{(j)}\right) > T$ the sample $\boldsymbol{\theta}^{(j)}$ is rejected (where the sample is kept if the inverse is true).
- The set of samples that are not discard define the new parameter space Θ_{i+1}.
- The simulator is run at a set budget of new parameter combinations from the new parameter space $\boldsymbol{\theta}_{new} \in \Theta_{i+1}$, i.e. $\boldsymbol{y}_{new} = \eta(X,\boldsymbol{\theta}_{new})$.
- The emulator is updated based on the new training data $\{\{X,\boldsymbol{\theta}_{new}\},\boldsymbol{y}_{new}\}$.
- Repeat until convergence.

The threshold is set given the statistical properties of the implausibility metric [8], e.g. for Eq. (16) the threshold can be set as $T = 3$ given Pukelsheim's 3σ rule [29]. By updating the emulator approximation at each iteration, the criteria can discard more parameter combinations with increased confidence. Finally, convergence is reached when either the uncertainty in the emulator is lower than the remaining uncertainties i.e. $V_o + V_m > \mathbb{V}(y(X,\boldsymbol{\theta}))$ or all the parameters are discarded.

4. Case study: numerical verification problems

The proposed two stage calibration and model discrepancy inference was verified on two numerical case studies; one where $\boldsymbol{x} \in \mathbb{R}^{N\times 1}$ and the other $\boldsymbol{x} \in \mathbb{R}^{N\times 2}$. In addition, the first numerical case study is used to benchmark the proposed decoupled approach, using Bayesian history matching and the sampling-based model discrepancy procedure, against the hierarchical Bayesian model formulation, where the model discrepancy and model parameters are jointly inferred together. In both case studies the simulator modelled the tip deflection of a cantilever beam subject to an open crack with a point force of 10kN at the tip. The stiffness reduction model for an open crack used in this case study was that proposed by Christides and Barr [30],

$$EI(x) = \frac{EI_0}{1 + C\exp(-2\alpha|x - l_{loc}|/t)} \tag{17}$$

where the stiffness along the beam $EI(\cdot)$ is a function of the length along the beam x, Young's modulus E, the second moment of area for the undamaged beam I_0, the beam thickness t, the crack location l_{loc} and α, a coefficient experimentally defined by Christides and Barr as 0.667. The constant $C = (I_0 - I_c)/I_c$ is a function of the undamaged I_0 and damaged second moments of area I_c, which for a rectangular beam are $I_0 = (wt^3)/12$ and $I_c = w(t - l_{cr})^3/12$; where w is the beam width and l_{cr} is the crack length. The tip deflection was numerically estimated via Euler–Bernoulli bending beam equation,

$$\frac{\partial^2 y}{\partial x^2} = -\frac{M(x)}{EI(x)} \tag{18}$$

where $M(\cdot)$ is the moment along the beam. In both case studies the beam used in the analysis was rectangular with the following dimensions: $l = 1$m, $w = 0.5$m and $t = 0.1$m.

4.1. Numerical case study: one input problem

The first illustrative case study considers a scenario where the input was crack location $\boldsymbol{x} = \boldsymbol{l}_{loc}$, the parameters were Young's modulus E and crack length l_{cr}, i.e. $\boldsymbol{\theta} = \{l_{cr}, E\}$, and the output (both from the simulator and experiments) was the tip deflection. In this analysis the true parameters were defined as $\hat{l}_{cr} = 38$mm and $\hat{E} = 68$GPa, i.e. $\hat{\boldsymbol{\theta}} = \{38, 68\}$; the simulator evaluated at these parameters is depicted in Fig. 1. The training inputs for both the simulator and experimental data were 13 equally spaced points from 0.1m to 0.9m, and the simulator parameters were evaluated between 0mm and 50mm, and 50GPa and 90GPa, resulting in 25 equally spaced data points. These training inputs were used to construct an emulator with a linear mean and Matérn 3/2 covariance. The model discrepancy was defined as,

$$\delta(\boldsymbol{x}) = 0.3(1.5 - \boldsymbol{x}) \times \sin(1.8(\boldsymbol{x} - 0.2) \times 2\pi) \tag{19}$$

shown in Fig. 1. The experimental data $\boldsymbol{z}$ was formed from the simulator output plus the additive model discrepancy where the observation noise was Gaussian distributed with variance 0.001, where the experimental data is displayed in Fig. 1. The prior model discrepancy uncertainty (used in BHM) was $V_m = 0.05$, reflecting the expected magnitude of the model discrepancy, where the error bars on the experimental data in Fig. 1 show the total prior uncertainties.

BHM was used to find the approximate posterior distribution, shown in Fig. 2 where the true parameters (shown in red) are close to the mode of the joint posterior distribution. Samples from the posterior distribution are shown in Fig. 1, where the output from the mode of the posterior distribution is visually in good agreement with the output at the true parameter values.

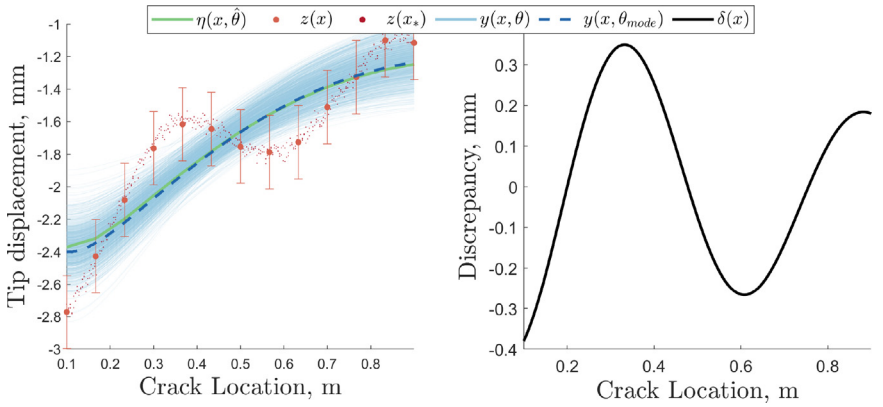

Fig. 1. The simulator, experimental data and model discrepancy for the first numerical case study. The left panel shows experimental training data $\boldsymbol{z}$ (●) with error bars indicating the total prior BHM uncertainties and testing data $\boldsymbol{z}_*$ (•). The simulator evaluated at the true parameters (—) is compared to the BHM samples (—) where the mode (- -) is indicated. The right panel depicts the model discrepancy.

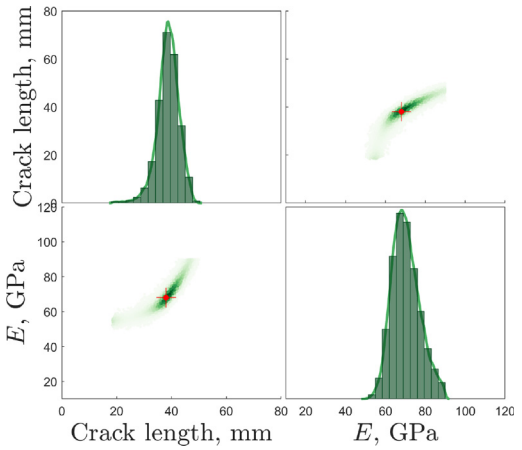

Fig. 2. Posterior joint parameter distribution from Bayesian history matching for the first numerical case study where red markers indicate true parameter values.

The proposed model discrepancy inference procedure was subsequently run for three scenarios:

1. Using a *MAP* estimate of the simulator parameters $\boldsymbol{\theta}^{MAP}$ and an empirical Bayes estimate of the GP hyperparameters $\hat{\boldsymbol{\phi}}$ (when the simulator output for $\boldsymbol{\theta}^{MAP}$ is used).
2. Marginalising out the simulator outputs $\boldsymbol{y}_*$ using importance sampling, with an empirical Bayes estimate of the GP hyperparameters $\hat{\boldsymbol{\phi}}$ (when the simulator output for $\boldsymbol{\theta}^{MAP}$ is used); $N_s = 1000$.
3. Marginalising out both the simulator outputs $\boldsymbol{y}_*$ and GP hyperparameters $\boldsymbol{\phi}$ via importance sampling; $N_s = 1000$ and $N_\phi = 500$.

For each scenario a zero mean and Matérn 3/2 covariance function were used.

The hierarchical Bayesian approach was also applied to the same training dataset where the prior parameter distributions were $l_{cr} \sim \mathcal{N}(35, 10)$ and $E \sim \mathcal{N}(70, 36)$ and the model discrepancy Gaussian process was also modelled with a Matérn 3/2 covariance function. These prior parameter distributions are more informative than those consider in BHM (which can be understood as a uniform prior), where a comparison is shown in Fig. 3. The reason for using more informative priors in the hierarchical Bayesian approach than in BHM, is due to the findings of Brynjarsdóttir and O'Hagan [6], where to reduce the problem of non-identifiability, one solution is to apply more informative priors, constraining the posterior when the likelihood is relatively flat as a result of the model discrepancy Gaussian process; although it is noted that obtaining more informative priors is challenging in practical applications. Other aspects of the hierarchical Bayesian analysis were kept the same as in the BHM approach, such that objective comparisons could be made. The posterior distributions from the hierarchical Bayesian approach were obtained via an adaptive Markov chain Monte Carlo scheme [31,32], such that 100,000 posterior samples were obtained with a 50,000 sample burn-in period. The autocorrelations of the chains were checked for stationarity in order to confirm convergence. The posterior distribution is shown in Fig. 4, where it can be seen that the parametric uncertainties are much larger than in the posterior distribution from Bayesian History Matching (BHM). The effect arises as the likelihood has dominated in the posterior distribution, enlarging the area of probably parameter values due to the insensitive likelihood function [4,6] — which arises as a result of modelling the model discrepancy as a Gaussian process during joint inference approach.

The results from the three decoupled approaches and the hierarchical Bayesian approach are shown in Fig. 5, were it can be seen that all of the methods have managed to accurately predict the tip deflection well, reflected in low normalised mean squared errors (NMSEs) in Table 1 for a 200 point independent test dataset. However, due to the large parametric uncertainty

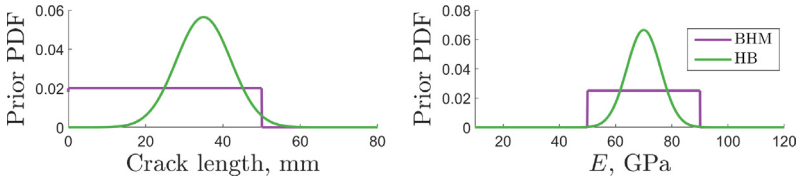

Fig. 3. Comparison of prior distributions for Bayesian history matching (BHM) and hierarchical Bayesian (HB) approaches.

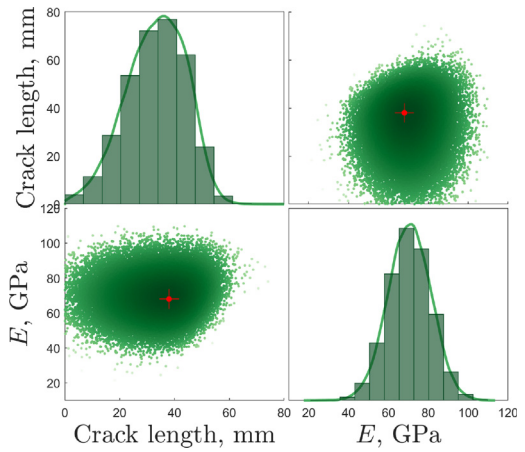

Fig. 4. Posterior joint parameter distribution from the hierarchical Bayesian approach for the first numerical case study where red markers indicate true parameter values.

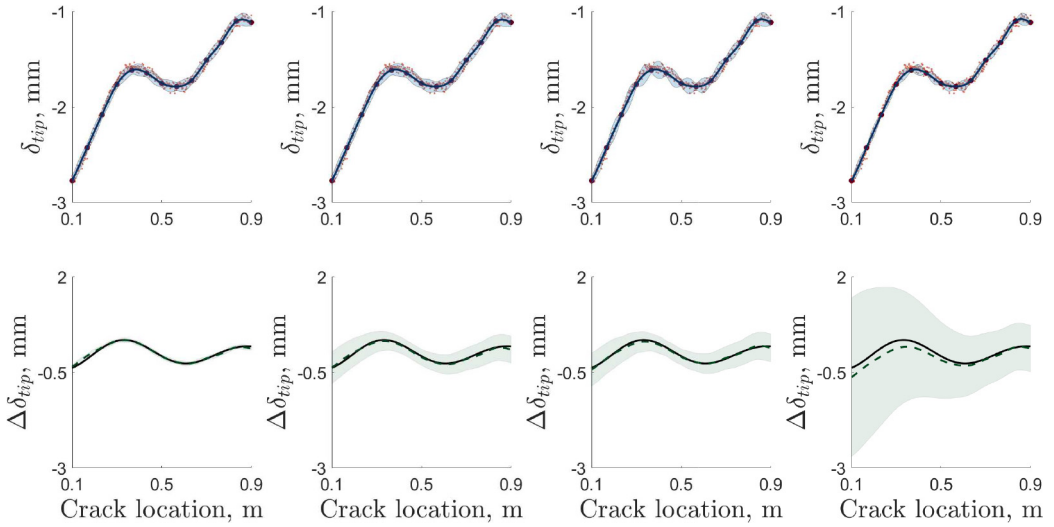

Fig. 5. A comparison of the three scenarios and the hierarchical Bayesian model (left to right) indicating the calibrated and bias corrected predictions (top panels) and model discrepancy (bottom panels) for the first numerical case study. The mean (—) and $\pm 3\sigma$ (blue shaded region) for the calibrated and bias corrected predictions are compared against the training $\boldsymbol{z}$ (●) and testing data $\boldsymbol{z}_*$ (●). The model discrepancy mean (- - -) and $\pm 3\sigma$ (green shaded region) are compared to the true model discrepancy (—). (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.).

Table 1
A comparison of the normalised mean squared errors for the experimental (z) and model discrepancy (δ) mean predictions for the first numerical case study. HB denotes the hierarchical Bayesian model.

	Scenario	1	2	3	HB
NMSE	z	0.768	0.765	0.765	0.770
NMSE	δ	1.756	2.288	1.654	39.349

in the posterior distribution from the hierarchical Bayesian approach, the inferred model discrepancy not only has a large variance, but also a less accurate mean prediction with a NMSE of 39.349, over 17 times larger than the highest NMSE from the decoupled approach. This shows the challenges an insensitive likelihood causes on the inference process, and why a decoupled solution is one approach that can be used to overcome these challenges. Furthermore, the hierarchical Bayesian approach has an underestimated predictive variance for the tip deflection δ_{tip}, with a large number of data points, particularly around the first peak at 0.36 m, exceeding a three standard deviation interval. This relatively simple numerical case study shows the problems with a hierarchical Bayesian approach and further motives the need for alternative solutions to the model discrepancy inference problem, such as the decoupled approach proposed in the paper.

In terms of comparing the three decoupled-based approaches, the main difference, as expected, is in the estimated uncertainty for the model discrepancy. Scenario one has the smallest uncertainty, with a larger number of experimental test data points outside a 3σ range when compared to the other two scenarios. The first scenario is also overconfident in the model discrepancy predictions, especially around 0.1m, where the true model discrepancy is outside of the 3σ range. Scenarios two and three increase the uncertainty in the model discrepancy, reflecting the parameter uncertainty in the posterior distribution, meaning the true model discrepancy remains within the 3σ variance range. The NMSE is lowest for the model discrepancy in scenario three, with scenario two producing the largest error in its mean prediction. It can be argued from the results, that scenario one is overconfident and although the mean prediction is better than scenario two, its distribution could be misleading and less helpful to the engineer by not reflecting the true uncertainty in the analysis. Finally, the posterior of the Gaussian process hyperparameters is obtainable as part of scenario three, and presented in Fig. 6.

4.2. Numerical case study: two input problem

The second case study considers a scenario with multiple inputs, where $X = \{\boldsymbol{l}_{loc}, \boldsymbol{l}_{len}\}$ i.e. crack location and length, where the output is tip deflection. The parameter in this analysis is the Young's modulus $\theta = E$, where the true parameter value is 68GPa. The training inputs for both the simulator and experimental data were 64 data points evenly spaced between 0.1m and 0.5m, and 0mm and 50mm (for the crack location and length respectively) where the outputs are shown in Fig. 7. The simulator parameters were evaluated at four points between 50GPa and 90GPa and an emulator was constructed using a linear mean and Matérn 3/2 covariance function. The model discrepancy was defined as,

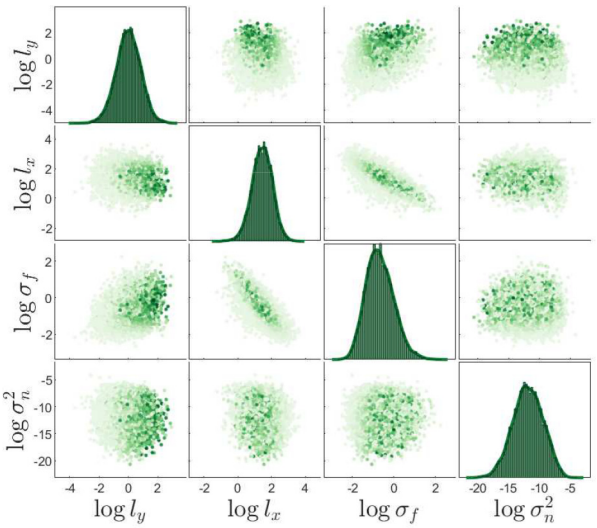

Fig. 6. The posterior hyperparameter distribution for scenario three for the first numerical case study; marginal (diagonals) and pairwise joint posterior (off-diagonal) distributions.

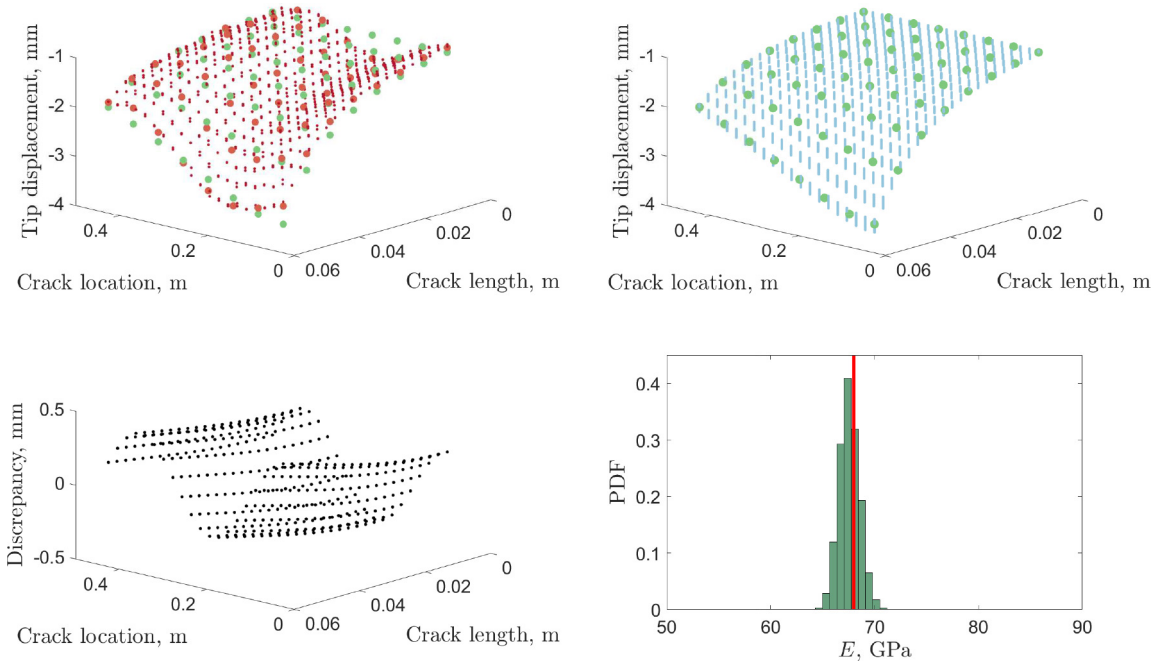

Fig. 7. The experimental data (top left panel), simulator (top right panel), model discrepancy (bottom left panel) and posterior parameter distribution (bottom right panel) for the second numerical case study. The top left panel compares the experimental training $\boldsymbol{z}$ (●), testing $\boldsymbol{z}_*$ (•) data and true simulator outputs (●). The top right panel compares the true simulator output (●) and BHM output samples (●). The bottom right panel shows the posterior parameter distribution compared to the true value (—).

$$\delta(\boldsymbol{x}) = 0.3\sin(3\boldsymbol{x}_1 \times 2\pi - 0.2) - 0.2(1 - \boldsymbol{x}_1)\cos(6\boldsymbol{x}_2 \times 2\pi) \tag{20}$$

and displayed in Fig. 7. Again the experimental data, shown in Fig. 7, was formed from the simulator at the true parameters plus the model discrepancy with Gaussian additive noise with a variance of 0.001. The prior model discrepancy variance was $V_m = 0.05$.

The approximate posterior from BHM is presented in Fig. 7 where the difference between the mode and true parameter value is 0.9%. Samples from the posterior are shown in Fig. 7 showing the simulator has been adequately calibrated.

The model discrepancy inference procedure was run for three scenarios:

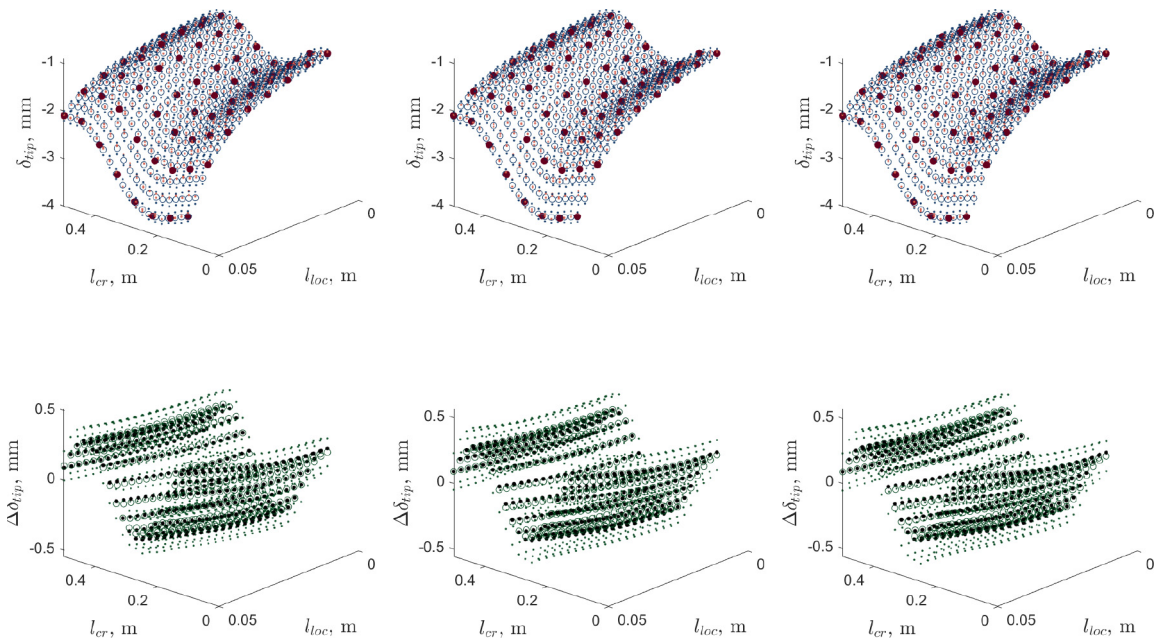

Fig. 8. A comparison of the three scenarios (left or right) indicating the calibrated and bias corrected predictions (top panels) and model discrepancy (bottom panels) for the second numerical case study. The mean (○) and $\pm 3\sigma$ (•) for the calibrated and bias corrected predictions are compared against the training $\boldsymbol{z}$ (●) and testing data $\boldsymbol{z}_*$ (•). The model discrepancy mean (•) and $\pm 3\sigma$ (•) are compared to the true model discrepancy (•).

Table 2
A comparison of the normalised mean squared errors for the experimental (z) and model discrepancy (δ) mean predictions for the second numerical case study.

	Scenario	1	2	3
NMSE	z	0.433	0.425	0.423
NMSE	δ	0.587	0.481	0.519

1. Using a *MAP* estimate of the simulator parameters $\boldsymbol{\theta}^{MAP}$ and an empirical Bayes estimate of the GP hyperparameters $\hat{\boldsymbol{\phi}}$ (when the simulator output for $\boldsymbol{\theta}^{MAP}$ is used).
2. Marginalising out the simulator outputs $\boldsymbol{y}_*$ using importance sampling, with an empirical Bayes estimate of the GP hyperparameters $\hat{\boldsymbol{\phi}}$ (when the simulator output for $\boldsymbol{\theta}^{MAP}$ is used); $N_s = 1000$.
3. Marginalising out both the simulator outputs $\boldsymbol{y}_*$ and GP hyperparameters $\boldsymbol{\phi}$ via importance sampling; $N_s = 1000$ and $N_\phi = 1000$.

For each scenario a zero mean and Matérn 3/2 automatic relevance determination covariance function were used. The results in Fig. 8 show adequate predictions for all the scenarios with scenario 3 producing the lowest predictive NMSE on a 400 point independent test dataset (see Table 2). Interestingly, the second approach achieves a lower NMSE on the model discrepancy when compared to scenario three (with both performing better than scenario one). It can be seen in Fig. 8 that the model discrepancy has been correctly captured by all three approaches. Finally, scenario three obtained the posterior hyperparameter distribution, useful in understanding the extracted model discrepancy. (see Fig. 9).

5. Case study: five storey shear structure

In order to demonstrate the effectiveness of the proposed approach an experimental case study is presented. This case study seeks to infer the model discrepancy of a modal finite element model used to predict the change in natural frequency when different masses are applied to the fourth floor (simulating a damage scenario). Estimation of the parameter distribution was performed using Bayesian history matching as outlined in [8]. A brief overview of the calibration process is introduced below, where the reader is referred to [8] for more details.

5.1. Calibration using Bayesian history matching

Bayesian history matching was applied to infer the material properties $\boldsymbol{\theta} = \{E, \nu, \rho\}$ (Young's modulus, Poisson's ratio and density) of a finite element model of a five storey shear structure ($\eta(\cdot,\cdot)$) known to have model-form errors due to modelling

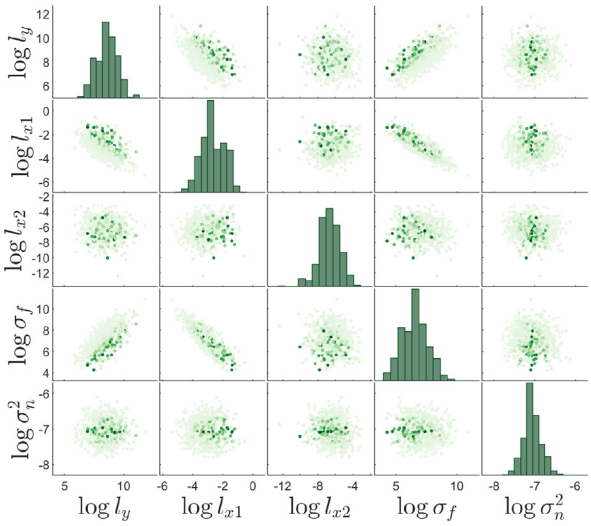

Fig. 9. The posterior hyperparameter distribution for scenario three for the second numerical case study; marginal (diagonals) and pairwise joint posterior (off-diagonal) distributions.

simplifications (particularly of the boundary condition between the structure and fixing). Observational data were the first five bending modes of a representative building structure, $\boldsymbol{z} = \{\boldsymbol{\omega}_1, \ldots, \boldsymbol{\omega}_5\}$, constructed from aluminium 6082, depicted in Fig. 10. These data were obtained via modal testing, where an electrodynamic shaker applied a Gaussian noise excitation with a bandwidth of 409.6 Hz, and five uniaxial accelerometers were used to capture the acceleration response at each of the five floors (where sample rate and time were chosen to allow frequency resolution of 0.05 Hz). Masses were incrementally added to the fourth floor of the structure $\boldsymbol{m} = \{0, 0.1, \ldots, 0.5\}$kg, representing pseudo-damage, and were treated as the inputs in this analysis i.e. $\boldsymbol{m} = \boldsymbol{x}$. Ten repeat estimates of the natural frequencies were obtained for each mass providing a representation of observational uncertainty.

Calibration was performed on training data, which were the ten repeat observations of the bending natural frequencies when $\boldsymbol{x} = \{0, 0.3, 0.5\}$. The testing data were the ten repeat observations of the bending natural frequencies when $\boldsymbol{x}_* = \{0.1, 0.2, 0.4\}$. The prior parameter bounds were $\pm 15\%$ of typical material properties for aluminium 6082; $E = 71$GPa, $\nu = 0.33$, $\rho = 2770$kg/m^3. These parameter bounds behave in a similar way to a uniform prior over the space. The approximate posterior distribution of the parameters, identified from the Bayesian history matching analysis, is displayed in Fig. 11. Samples of the simulator output distribution (for each of the five natural frequencies) $\boldsymbol{y}_{*,i}^{(j)} \sim p\left(\boldsymbol{y}_{*,i} | \boldsymbol{x}_*, \boldsymbol{\theta}^{(j)}\right) \; \forall i \in \{1:5\}$, given samples of the posterior distribution $\boldsymbol{\theta}^{(j)}$, are depicted in Fig. 12, where the error bars define to the prior model discrepancy and observational uncertainties. It is clear from Fig. 12 that there is a large amount of model discrepancy for the first natural frequency.

(a) (b)

Fig. 10. Representative five storey building structure. Panel (a) show the test setup and panel (b) presents an example of the pseudo-damage, glued added masses, applied to the fourth floor [8].

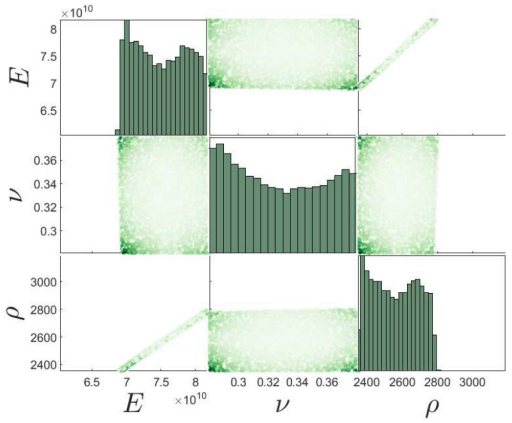

Fig. 11. Calibrated simulator posterior parameter distributions; marginal (diagonals) and pairwise joint posterior (off-diagonal) distributions [8].

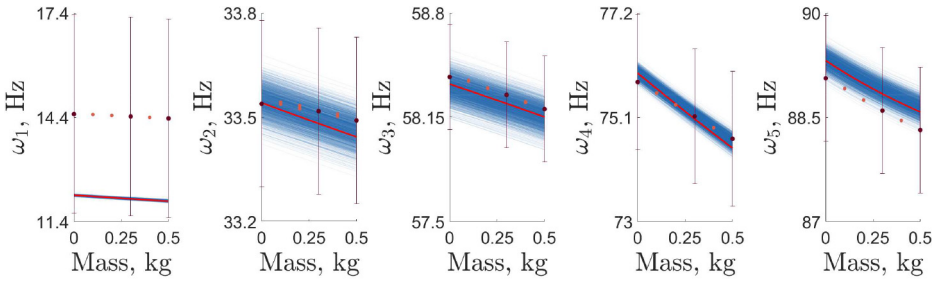

Fig. 12. Calibrated simulator outputs; simulator output samples $\eta_i\left(\boldsymbol{x}_*, \boldsymbol{\theta}^{(j)}\right)$ (—), *MAP* simulator output $\eta_i(\boldsymbol{x}_*, \boldsymbol{\theta}^{MAP})$ (—), training data $\boldsymbol{z}_i$ (●), testing data $\boldsymbol{z}_{*,i}$ (●) $\forall i \in \{1:5\}$. The error bound indicate the prior observational and model discrepancy uncertainties used in the Bayesian history matching analysis [8].

5.2. Model discrepancy inference

Inference of the model discrepancy from the Bayesian history matching analysis in Section 5.1 was performed using three approaches:

1. Using a *MAP* estimate of the simulator parameters $\boldsymbol{\theta}^{MAP}$ and an empirical Bayes estimate of the GP hyperparameters $\hat{\boldsymbol{\phi}}$ (when the simulator output for $\boldsymbol{\theta}^{MAP}$ is used).
2. Marginalising out the simulator outputs $\boldsymbol{y}_*$ using importance sampling, with an empirical Bayes estimate of the GP hyperparameters $\hat{\boldsymbol{\phi}}$ (when the simulator output for $\boldsymbol{\theta}^{MAP}$ is used); $N_s = 1000$.
3. Marginalising out both the simulator outputs $\boldsymbol{y}_*$ and GP hyperparameters $\boldsymbol{\phi}$ via importance sampling; $N_s = 1000$ and $N_\phi = 500$.

For each of the three methods the model discrepancy was inferred as a map $\mathcal{GP}_i : \{Y, X\} \rightarrow \boldsymbol{z}_i \; \forall i \in \{1:5\}$. It is noted that a multiple output Gaussian process could be implemented [33], meaning only one map would need to be inferred from $\{Y, X\}$ to Z. This would not change the general formulation of the approach and is therefore left for further research. The priors for each of the five GP models were zero mean functions with Matérn 3/2 automatic relevance determination covariance functions (specified by Eqs. (3)–(5)).

For the third approach — the marginalisation of both the simulator outputs and GP hyperparameters — independent Gaussian priors were defined for each hyperparameter in the set. The priors for lengthscale hyperparameters assumed that the process will change slowly with the input (i.e. large lengthscales), $\log l^y_{i,j} \sim \mathcal{N}(8,1) \; \forall i,j \in \{1:5\}$ and $\log l^x_i \sim \mathcal{N}(8,1) \; \forall i \in \{1:5\}$. The signal variance priors were $\log \sigma_{f,i} \sim \mathcal{N}(0,10) \; \forall i \in \{1:5\}$, and the noise variance priors were $\log \sigma_{n,i} \sim \mathcal{N}(-11,4) \; \forall i \in \{1:5\}$.

The calibrated and bias-corrected predictions for each of the three approaches are displayed in Fig. 13, with the inferred model discrepancies shown in Fig. 14. Firstly, it can be seen that the mean predictions of all three approaches, both in terms of their output predictions' and inferred model discrepancies', are visually similar. The main difference between all three approaches is their estimation of the uncertainty, with significant differences in the inferred model discrepancy uncertainty.

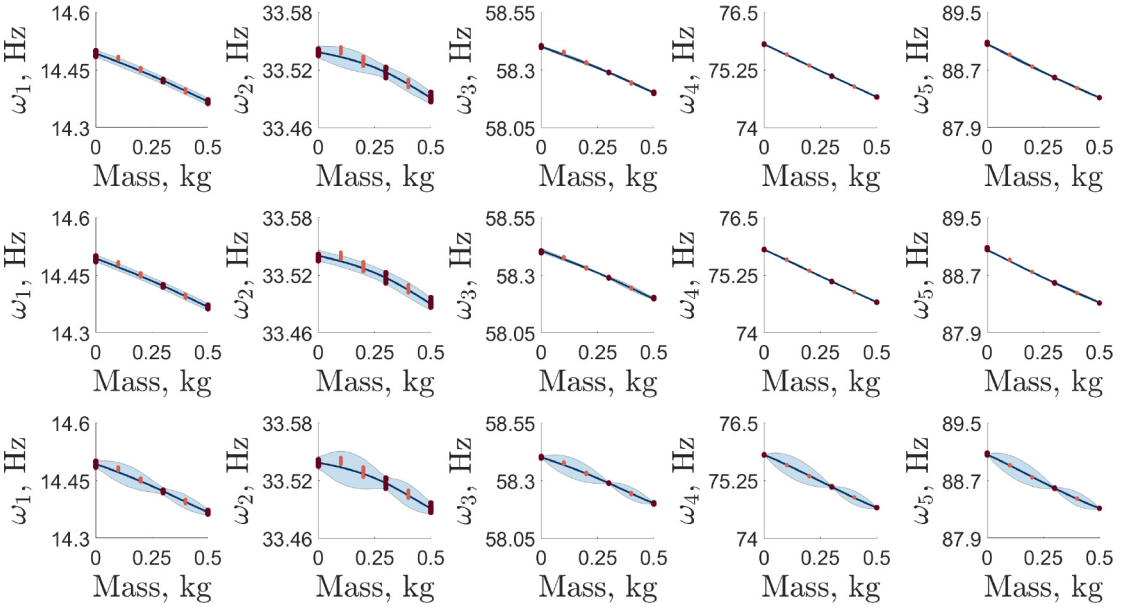

Fig. 13. Calibrated and bias corrected natural frequency outputs; mean (—) and $\pm 2\sigma$ (blue shaded region), training data $\boldsymbol{z}_i$ (●), testing data $\boldsymbol{z}_{*,i}$ (●). First row, predictions from scenario 1, i.e. $p\left(\boldsymbol{z}_{*,i} \mid \boldsymbol{x}_*, \boldsymbol{\theta}^{MAP}, \mathcal{D}, \hat{\boldsymbol{\phi}}_i\right) \forall i \in \{1:5\}$; second row, predictions from scenario 2, i.e. $p\left(\boldsymbol{z}_{*,i} \mid \boldsymbol{x}_*, \boldsymbol{\theta}, \mathcal{D}, \hat{\boldsymbol{\phi}}_i\right) \forall i \in \{1:5\}$; third row, predictions from scenario 3, i.e. $p(\boldsymbol{z}_{*,i} \mid \boldsymbol{x}_*, \boldsymbol{\theta}, \mathcal{D}) \forall i \in \{1:5\}$.

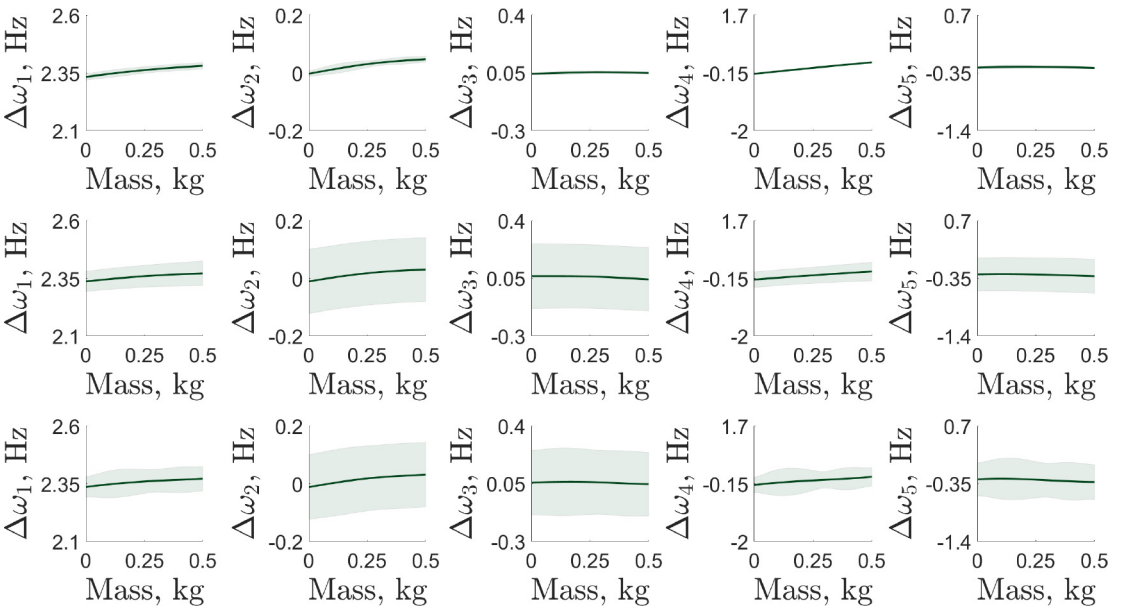

Fig. 14. Inferred model discrepancies; mean (—) and $\pm 2\sigma$ (green shaded region). First row, predictions from scenario 1; second row, predictions from scenario 2; third row, predictions from scenario 3.

The second and third methods have propagated the posterior parameter uncertainty through to the model discrepancy and the output predictions, unlike the first approach which collapses this uncertainty down to the parameter *MAP* estimates. This increase in uncertainty in the model discrepancy from methods two and three is useful to the engineer as it provides a better reflection of the underlying model discrepancy and will be helpful in identifying simulator improvements, as the 'true' model discrepancy is more likely to be contained within the confidence intervals. This is particularly clear given the results in the numerical case studies, where method one resulted in model discrepancy predictions where the 'true' model discrepancy

occasionally exceeded 3σ. In terms of output prediction, each of the methods visually appear to have captured the noise process, with the third method showing increases in uncertainty outside of the training observations. This effect is likely to be caused by the small number of training observations, causing the uncertainty to increase away from the training data, indicating that the prior has a large effect on the posterior due to the small number of observations in the likelihood. However, the extra uncertainty quantified by marginalising out the posterior hyperparameter distributions is useful for gaining an insight into the level of trust in the identified model discrepancy given the limited training data used to estimate the discrepancy.

Several validation metrics have been applied in order to quantitatively assess the performance of the inferred models for each of the three scenarios. The first metric, the normalised mean squared error (NMSE) (the sum of the squared errors divided by the variance and the number of data points), assesses the performance of the mean prediction. The second metric is the maximum mean discrepancy (MMD) distance, a measure of the distance between two distributions [34]. The distance is the difference between the means of two kernel embeddings of the data (where here a Gaussian kernel is used, where the scale parameter is inferred via a median heuristic [35]). The third metric is the posterior likelihood, and is a measure of the probability of the data coming from the inferred GP model. The validation metrics are applied to the predictions from each three scenario and are shown in Fig. 15 and Table 3.

In terms of the mean predictions, the NMSEs indicate that on average the third approach provided the best mean performance. In fact, both the second and third methods outperformed the first method in all of their mean predictions (apart from method three's predictions of the fourth natural frequency). This shows that including and propagating these sources of uncertainties are beneficial for the overall mean predictive performance, with the results supporting the conclusions from the numerical case studies. Helpfully, the models for all three scenarios show the same general mean predictive behaviour across the five natural frequencies, with predictions being poorest for the second natural frequency. Comparing the output distributions, the MMD distances for each scenario are relatively comparable, with the third method performing best on average. The reason for similar MMD distances is that the data distribution is being inferred from 10 observations at every input, and that this has a greater effect on the distance than each scenarios change in predictive distribution. This demonstrates the challenges in validating predictive distributions when only a few number of validation data points are available. In comparison to the NMSEs, the posterior likelihood indicates a different assessment of which natural frequency is predicted best — the second natural frequency is most likely to have produced the observational data. The posterior likelihoods indicate that on average the first method was more likely to have produced the observational data than the other approaches. However, the second method produces the highest posterior likelihoods for the third and fourth natural frequencies. Furthermore, the second and third methods have comparable posterior likelihoods for the first to third natural frequencies, which are very similar to method one. It is noted that the first method does not reflect the uncertainty associated with the parameters, and therefore may be overconfident in it's predictions, given the training observations.

From the third approach it is possible to obtain the posterior distribution of the GP hyperparameters. Fig. 16 depicts two of the posterior hyperparameters distributions; the first and fifth natural frequencies. The posteriors show that the lengthscales for the simulator output L_y are all uncorrelated (as expected by the automatic relevance determination and distance metric assumptions in the covariance function, i.e. L_y is diagonal). Furthermore, both show that the modes of the signal variance and noise variance are fairly constant when compared to the lengthscales. This means that the noise and signal variance have been well-identified, and the output uncertainty is mainly attributed the uncertainty in the lengthscales. Resultantly, the posterior hyperparameter distributions provide a high level of insight into the inferred model discrepancy. With more observational data, these posterior distributions will provide insight into the type of missing functional-form in the simulator, as the lengthscale distributions will be expected to decrease in uncertainty.

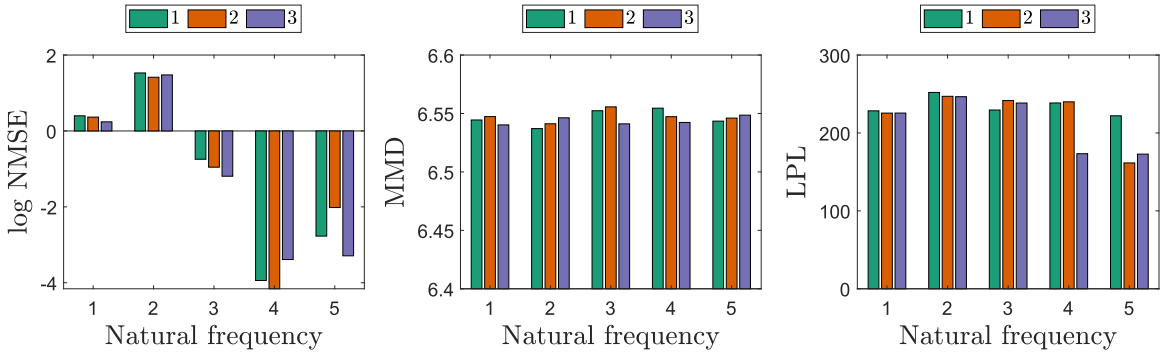

Fig. 15. Validation metrics for each scenario showing the differences between the bias-corrected output predictions and experimental data. NMSE - Normalised Mean Square Error; MMD - Maximum Mean Discrepancy; LPL - Log Posterior Likelihood.

Table 3
Validation metrics for each scenarios showing the differences between the bias-corrected output predictions and experimental data. NMSE - Normalised Mean Square Error; MMD - Maximum Mean Discrepancy; LPL - Log Posterior Likelihood.

	Scenario	ω_1	ω_2	ω_3	ω_4	ω_5	Average
NMSE	1	1.490	4.610	0.474	0.019	0.063	1.331
	2	1.440	**4.118**	0.385	**0.016**	0.133	1.218
	3	**1.271**	4.378	**0.303**	0.034	**0.037**	**1.205**
MMD	1	6.544	**6.537**	6.552	6.555	**6.543**	6.546
	2	6.547	6.541	6.556	6.547	6.546	6.548
	3	**6.540**	6.546	**6.541**	**6.542**	6.549	**6.544**
LPL	1	**228.4**	**252.0**	229.5	238.5	**222.0**	**234.1**
	2	225.5	247.0	**241.6**	**239.9**	161.5	223.1
	3	225.5	246.5	238.4	173.4	172.8	211.3

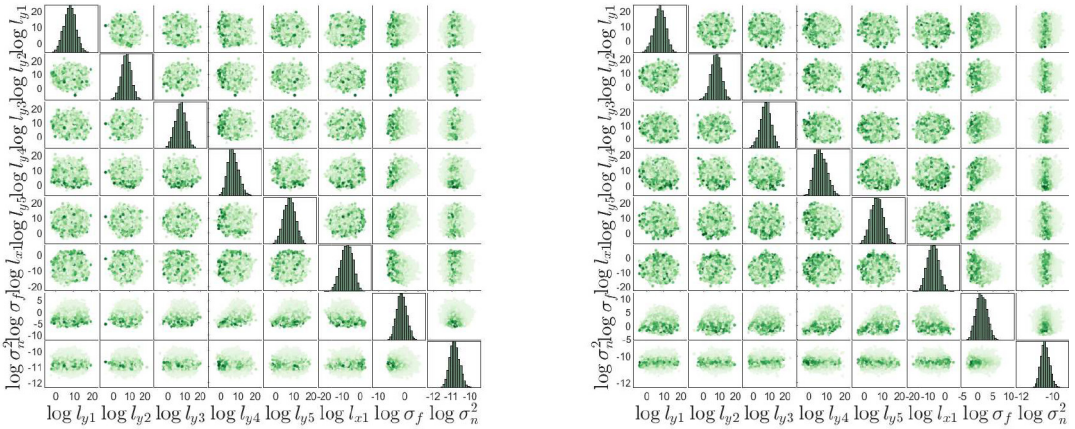

Fig. 16. Inferred posterior distributions of Gaussian process hyperparameters for the first (ω_1) and fifth (ω_5) natural frequencies.

6. Conclusions

Every computer model (here defined as a *simulator*) will imperfectly reflect the real world due to some level of model discrepancy (whether due to missing physics, simplifications or approximations etc.). Without identifying the level of model discrepancy within a simulator, predictions are likely to be inaccurate. This paper proposes a method based on Gaussian process (GP) regression and a sampling-based approach for identified model discrepancy given some parameter distribution. This method has been demonstrated to be effective on numerical examples and an experimental case study of a representative five storey building structure.

The approach in this paper allows for bias-corrected predictions to be constructed that marginalise out the simulator outputs, with the additional ability to marginalise out the GP hyperparameters. By performing this process, the bias-corrected predictive distributions better reflect the parameter and hyperparameter uncertainty in the predictions and help the engineer identify 'true' improvements to a simulator, rather than those based on overconfident estimations of the model discrepancy. The technique relies on generating a set of Gaussian process maps from the uncertain simulator outputs and deterministic inputs to observational data and performing weighted averages to form the bias-corrected predictive distributions.

Three scenarios were investigated: using a *MAP* estimate of the simulator parameters and simulator outputs, and an empirical Bayes estimate of the GP hyperparameters; marginalising out the simulator outputs using importance sampling, and an empirical Bayes estimate of the GP hyperparameters; and marginalising out both the simulator outputs and the GP hyperparameters via importance sampling.

The numerical case studies show that a two stage decoupled process, utilising Bayesian history matching and the proposed model discrepancy procedure, is appropriate for calibrating a simulator and extracting model discrepancy. In addition, the first numerical case study demonstrated issues associated with the hierarchical Bayesian approach that seeks to jointly infer the parameters and model discrepancy. In accordance with the literature [4,6], the approach leads to an insensitive likelihood, and can cause non-identifiability issues. In the numerical case study the hierarchical Bayesian approach inferred a model discrepancy distribution that was more uncertain than the proposed decoupled approaches, and had a worse mean predictive performance. Furthermore, both numerical case studies showed that considering both the simulator output

and hyperparameter uncertainties provides more information about the model discrepancy function, and in improves the performance of the mean prediction.

Finally, an experimental case study was provided, where it was shown that the third approach (marginalising both the simulator outputs and the GP hyperparameters) produced the best mean predictions. In addition, the uncertainty associated with the parameters was propagated onto the model discrepancy and output predictions, better reflecting the uncertainty quantified by the BHM parameter posterior distribution. The inclusion of the parametric uncertainty is valuable in understanding the 'true' model discrepancy, and is beneficial in determining what is known about the model discrepancy from the analysis. In addition, the third approach provides posterior distributions of the hyperparameter, which provide further insight into the model discrepancy process.

Declaration of Competing Interest

None.

Acknowledgements

The authors would like to acknowledge the support of the UK Engineering and Physical Sciences Research Council via grants EP/R006768/1 and EP/N010884/1.

References

[1] Marc C Kennedy, Anthony O'Hagan, Bayesian calibration of computer models, J. Roy. Stat. Soc.: Ser. B (Stat. Methodol.) 63 (3) (2001) 425–464.
[2] James O Berger, James Cavendish, Rui Paulo, Chin-Hsu Lin, John A. Cafeo, Jerry Sacks, Maria J. Bayarri, Jian Tu, A Framework for Validation of Computer Models, Technometrics 49 (2) (2007) 138–154.
[3] Dave Higdon, James Gattiker, Brian Williams, Maria Rightley, Computer model calibration using high-dimensional output, J. Am. Stat. Assoc. 103 (482) (2008) 570–583.
[4] Paul D. Arendt, Daniel W. Apley, Wei Chen, Quantification of model uncertainty: calibration, model discrepancy, and identifiability, J. Mech. Des. 134 (10) (2012) 100908.
[5] Paul D. Arendt, Daniel W. Apley, Wei Chen, David Lamb, David Gorsich, Improving identifiability in model calibration using multiple responses, J. Mech. Des. 134 (10) (2012).
[6] Jenny Brynjarsdóttir, A. O'Hagan, Learning about physical parameters: the importance of model discrepancy, Inverse Prob. 30 (11) (2014) 114007.
[7] Paul D. Arendt, Daniel W. Apley, Wei Chen, A preposterior analysis to predict identifiability in the experimental calibration of computer models, IIE Trans. (Inst. Industr. Eng.) 48 (1) (2016) 75–88.
[8] Paul Gardner, Charles Lord, Robert J. Barthorpe, Bayesian history matching for structural dynamics applications, Mech. Syst. Signal Process. 140 (2020) 106828.
[9] Carl E. Rasmussen, Christopher K.I. Williams, Gaussian processes for machine learning, MIT Press, 2006.
[10] Peter S. Craig, Michael Goldstein, Allan H. Seheult, James A. Smith. Pressure Matching for Hydrocarbon Reservoirs: A Case Study in the Use of Bayes Linear Strategies for Large Computer Experiments. In Lecture Notes in Statistics, pages 37–93. 1997.
[11] Neil R Edwards, David Cameron, Jonathan Rougier, Precalibrating an intermediate complexity climate model, Clim. Dyn. 37 (7–8) (2011) 1469–1482.
[12] Michael Goldstein, External Bayesian analysis for computer simulators. In Bayesian Statistics 9, number 1996, pages 201–228. Oxford University Press, 2011.
[13] Daniel Williamson, Michael Goldstein, Lesley Allison, Adam Blaker, Peter Challenor, Laura Jackson, Kuniko Yamazaki, History matching for exploring and reducing climate model parameter space using observations and a large perturbed physics ensemble, Clim. Dyn. 41 (7–8) (2013) 1703–1729.
[14] Ian Vernon, Michael Goldstein, Richard Bower, Galaxy formation: Bayesian history matching for the observable universe, Stat. Sci. 29 (1) (2014) 81–90.
[15] Ioannis Andrianakis, Ian R. Vernon, Nicky McCreesh, Trevelyan J. McKinley, Jeremy E. Oakley, Rebecca N. Nsubuga, Michael Goldstein, Richard G. White, Bayesian History Matching of Complex Infectious Disease Models Using Emulation: A Tutorial and a Case Study on HIV in Uganda, PLoS Comput. Biol. 11 (1) (2015).
[16] I. Andrianakis, I. Vernon, N. McCreesh, T.J. McKinley, J.E. Oakley, R.N. Nsubuga, M. Goldstein, R.G. White, History matching of a complex epidemiological model of human immunodeficiency virus transmission by using variance emulation, J. Roy. Stat. Soc.: Ser. C (Appl. Stat.) 66 (4) (2017) 717–740.
[17] Andreas Svensson, Arno Solin, Simo Särkkä, Thomas B Schön, Computationally Efficient Bayesian Learning of Gaussian Process State Space Models. 2015a.
[18] M.I. Friswell, J.E. Mottershead, Finite element model updating in structural dynamics, 1995.
[19] Jerome Sacks, William J. Welch, Toby J. Mitchell, Henry P. Wynn, Design and analysis of computer experiments, Stat. Sci. 4 (4) (1989) 409–423.
[20] Habib N. Najm, Uncertainty quantification and polynomial chaos techniques in computational fluid dynamics, Annu. Rev. Fluid Mech. 41 (1) (2009) 35–52.
[21] Rohit Tripathy, Ilias Bilionis, Deep UQ: Learning deep neural network surrogate models for high dimensional uncertainty quantification, 2018.
[22] Anthony O'Hagan, John F.C. Kingman, Curve fitting and optimal design for prediction, J. Roy. Stat. Soc. Ser. B (Methodol.) 40 (1) (1978) 1–42.
[23] Michael L. Stein, Interpolation of spatial data: Some theory for Kriging. Springer Series in Statistics, 1999.
[24] Jun Sun, Bin Feng, Xu Wenbo, Particle swarm optimization with particles having quantum behavior, in: Proceedings of the 2004 Congress on Evolutionary Computation, 2004, pp. 325–331.
[25] Robert B. Gramacy, Nicholas G. Polson, Particle learning of Gaussian process models for sequential design and optimization, J. Comput. Graph. Stat. (2011).
[26] Dejan Petelin, Matej Gašperin, Václav Šmídl, Adaptive importance sampling for Bayesian inference in Gaussian process models, IFAC Proc. Vol. (IFAC-PapersOnline) (2014).
[27] Andreas Svensson, Johan Dahlin, Thomas B. Schön, Marginalizing Gaussian process hyperparameters using sequential Monte Carlo, in: 2015 IEEE 6th International Workshop on Computational Advances in Multi-Sensor Adaptive Processing, CAMSAP 2015, 2015b.
[28] Eric C. Anderson, Monte Carlo Methods and Importance Sampling. Lecture Notes, 1999.
[29] Friedrich Pukelsheim, The three sigma rule, Am. Stat. 48 (2) (1994) 88–91.
[30] S. Christides, A.D.S. Barr, One-dimensional theory of cracked Bernoulli-Euler beams, Int. J. Mech. Sci. 26 (11–12) (1984) 639–648.
[31] Heikki Haario, Marko Laine, Antonietta Mira, Eero Saksman, DRAM: Efficient adaptive MCMC, Stat. Comput. 16 (4) (2006) 339–354.
[32] Ralph C Smith, Uncertainty quantification: theory, implementation, and applications, Soc. Industr. Appl. Math. (2013).
[33] Thomas E. Fricker, Jeremy E. Oakley, Nathan M. Urban, Multivariate Gaussian process emulators with nonseparable covariance structures, Technometrics 55 (1) (2013) 47–56.

[34] Arthur Gretton, Karsten M. Borgwardt, Malte J. Rasch, Bernhard Scholkopf, Alexander Smola, A kernel two-sample test, J. Mach. Learn. Res. 13 (1) (2012) 723–773.
[35] Arthur Gretton, Kenji Fukumizu, Choon Hui Teo, Le Song, Bernhard Schölkopf, Alexander J. Smola, A kernel statistical test of independence, in: Neural Information Processing Systems, 2008, pp. 585–592.