Bayesian Analysis (2022) **17**, Number 4, pp. 1301–1350

Deep Gaussian Processes for Calibration of Computer Models (with Discussion)*

Sébastien Marmin† and Maurizio Filippone‡

Abstract. Bayesian calibration of black-box computer models offers an established framework for quantification of uncertainty of model parameters and predictions. Traditional Bayesian calibration involves the emulation of the computer model and an additive model discrepancy term using Gaussian processes; inference is then carried out using Markov chain Monte Carlo. This calibration approach is limited by the poor scalability of Gaussian processes and by the need to specify a sensible covariance function to deal with the complexity of the computer model and the discrepancy. In this work, we propose a novel calibration framework, where these challenges are addressed by means of compositions of Gaussian processes into Deep Gaussian processes and scalable variational inference techniques. Thanks to this formulation, it is possible to obtain a flexible calibration approach, which is easy to implement in development environments featuring automatic differentiation and exploiting GPU-type hardware. We show how our proposal yields a powerful alternative to the state-of-the-art by means of experimental validations on various calibration problems. We conclude the paper by showing how we can carry out adaptive experimental design, and by discussing the identifiability properties of the proposed calibration model.

Keywords: Bayesian inference, neural nets, computer experiments, deep Gaussian process.

MSC2020 subject classifications: Primary 62F15, 62M45; secondary 62K99.

1 Introduction

The task of carrying out inference of parameters of expensive computer models from data is a classical problem in Statistics (Sacks et al., 1989). Such a problem is referred to as *calibration* (Kennedy and O'Hagan, 2001), and the results are often of interest to draw conclusions on parameters that have a direct interpretation of physical quantities (see, e.g., Section 2.2 of Brynjarsdóttir and O'Hagan 2014). Calibration finds numerous applications in fields as diverse as climatology (Sansó et al., 2008; Salter et al., 2018), environmental sciences (Larssen et al., 2006; Arhonditsis et al., 2007), biology (Henderson et al., 2009), and mechanical engineering (Williams et al., 2006), to name a few. There are many fundamental difficulties in calibrating expensive computer models, which we can distinguish between computational and statistical. Computational issues

*MF gratefully acknowledges support from the AXA Research Fund and the Agence Nationale de la Recherche (grant ANR-18-CE46-0002 and ANR-19-P3IA-0002).

†Laboratoire national de métrologie et d'essais (LNE), 29 avenue Hennequin, 78197 Trappes, sebastien.marmin@lne.fr

‡EURECOM, Campus SophiaTech 450 Route des Chappes, 06410 Biot

 https://doi.org/10.1214/21-BA1293

arise from the fact that traditional optimization and inference techniques require running the expensive computer model many times for different values of the parameters, which might be unfeasible within a given computational budget. Statistical limitations, instead, arise from the fact that computer models can only abstract real processes with a given level of accuracy.

Building on previous work from Sacks et al. (1989), in their seminal paper, Kennedy and O'Hagan (2001) propose a statistical model, based on Gaussian processes (GPs; Rasmussen and Williams 2006), which jointly tackles the problems above. In their model, which we will refer to as KOH, the output of a deterministic computer model is emulated through a GP estimated from a set of computer experiments; in this way, computational issues are bypassed by using the predictive distribution of the emulating GP for any given set of parameters in place of expensive runs of the computer model. Observations from the real process, also known as field observations, instead, are modeled through the GP emulating the computer model with the addition of a so-called discrepancy term, which is also assigned a GP prior. The introduction of the discrepancy term is key to avoid biased estimates of the parameters due to misspecifications of the computer model (Brynjarsdóttir and O'Hagan, 2014). The KOH model is treated in a Bayesian way, making it suitable for problems where quantification of uncertainty is an important requirement. This is often the case when one is interested in drawing conclusions on parameters of interest, making predictions for decision-making with specific cost associated to predictions, or when one is interested in iteratively improving the experimental design.

While the KOH model and inference make for an attractive and elegant framework to tackle quantification of uncertainty for calibration of expensive computer models, there are limitations which we aim to overcome with this work. From the modeling perspective, GPs are indeed flexible emulators, provided that a suitable covariance function is chosen, as in the literature of nonstationary GPs (e.g. Paciorek and Schervish 2003). However, more recent approaches like Deep GPs (DGPs; Damianou and Lawrence 2013) have shown great modeling flexibility for many classes of functions and can potentially lead to more accuracy in the emulation of the computer model and the real process compared to GPs. From the computational perspective, limitations are inherited from the poor scalability of GPs (Rasmussen and Williams, 2006), for which inference becomes impractical when the number of runs of the code and the number of real observations are collectively beyond a few thousands. In addition, the use of Markov chain Monte Carlo (MCMC) (Neal, 1993) techniques to carry out inference for GP models can be painfully slow without careful tuning and clever parameterizations (Filippone et al., 2013; Filippone and Girolami, 2014).

This work aims to tackle these issues by proposing the use of recent developments in the GP and DGP literature and variational inference, (i) to extend the modeling capabilities of GPs in emulation using DGPs; (ii) to extend the original framework in Kennedy and O'Hagan (2001), by casting the model as a special case of a DGP; (iii) to adapt techniques based on random feature expansions and stochastic variational inference, building on the work by Cutajar et al. (2017), to obtain a scalable framework for Bayesian calibration of computer models. Thanks to this formulation, it is possible

to obtain a flexible calibration approach, which is easy to implement in development environments featuring automatic differentiation and exploiting GPU-type hardware.

We validate our proposal, which we name DGP-CAL, on a variety of calibration problems, comparing with alternatives from the state-of-the-art. We demonstrate the flexibility and the scalability of DGP-CAL, as well as the ability to capture the uncertainty in model parameters and model discrepancy. We conclude the paper by showing how we can carry out adaptive experimental design, and by discussing the identifiability properties of the proposed calibration model. The code to replicate all the results in the paper is available at the following url:

https://github.com/SebastienMarmin/variational-calibration.

2 Background

In this section, we introduce the problem of calibration of computer models and we present the KOH model. We then introduce Gaussian processes (GPs), which are the main modeling ingredients in the KOH model. Motivated by the difficulties associated with carrying out inference with GPs, we present random feature expansions as a way to reduce complexity and being able to exploit recent advances in approximate inference. In particular, we focus on variational inference (VI) techniques that are able to operate on mini-batches of data and that can be easily implemented in developing environments featuring automatic differentiation. We conclude this section by showing how we can increase flexibility of GPs by composing processes, obtaining DGPs, for which we can extend the use of random feature expansions and VI. This background material gives us all the elements to present our proposed calibration model in Section 3.

2.1 Bayesian Calibration

Consider prediction and uncertainty quantification for a phenomenon approximated by a computer model, which is expensive to evaluate. Throughout the paper, we will assume that the output of the computer model is univariate, but we will discuss ways in which we can deal with multiple responses. We denote observations from the real phenomenon of interest by $y \in \mathbb{R}$, and we assume that we have n of these available $\boldsymbol{y} = [y_1, \dots, y_n]^\top$ for a number of inputs $X = [\boldsymbol{x}_1, \dots, \boldsymbol{x}_n]^\top$, with $\boldsymbol{x}_i \in \mathcal{D}_1 \subset \mathbb{R}^{d_1}$. For example, in climate modeling, $\boldsymbol{y}$ could correspond to temperature measurements at n locations identified by latitude and longitude (in this case, $\mathcal{D}_1 = [-90, 90] \times [-180, 180[$). The computer model simulating the real phenomenon requires the so-called calibration parameters $\boldsymbol{\theta} \in \mathcal{D}_2 \subset \mathbb{R}^{d_2}$, as well as input variables $\boldsymbol{x} \in \mathcal{D}_1$. Calibration parameters may have a physical meaning (e.g., exchange rates determining the carbon cycle) and inference over these is a central goal of Bayesian calibration.

Beside the observations $\boldsymbol{y}$ associated with X, the computer model is run at (possibly different) inputs $X^* = [\boldsymbol{x}_1^*, \dots, \boldsymbol{x}_N^*]^\top$ and calibration parameters $T = [\boldsymbol{t}_1, \dots, \boldsymbol{t}_N]^\top$, yielding a collection of outputs $\boldsymbol{z} = [z_1, \dots, z_N]^\top$. Note that we denote by T the collection of N parameter configurations at which the computer model is run, while $\boldsymbol{\theta}$

denotes the true (unknown) parameter we are interested in inferring. Generally N is larger than n as running the computer model is easier compared to obtaining real world observations (albeit computationally expensive).

Following the definition of the KOH model in Kennedy and O'Hagan (2001), we assume that $\boldsymbol{y}$ and $\boldsymbol{z}$ are drawn from $p(y_i|f_i)$ and $p(z_j|\eta_j^*)$, which determine the likelihood functions. The vectors $\boldsymbol{f} = [f_1, \ldots, f_n]^\top$ and $\boldsymbol{\eta}^* = [\eta_1^*, \ldots, \eta_N^*]^\top$ result from mapping $(\boldsymbol{x}_i, \boldsymbol{\theta})_{i=1,\ldots,n}$ and $(\boldsymbol{x}_j^*, \boldsymbol{t}_j)_{j=1,\ldots,N}$ through random functions f and η, respectively. The link between the computer model with latent representation η, and the real phenomenon with latent representation f, is modeled by

$$f(\boldsymbol{x}, \boldsymbol{t}) = \eta(\boldsymbol{x}, \boldsymbol{t}) + \delta(\boldsymbol{x}), \tag{2.1}$$

where $\delta(\boldsymbol{x})$ represents the discrepancy between the computer model and the real process. Figure 1 illustrates the KOH calibration model.

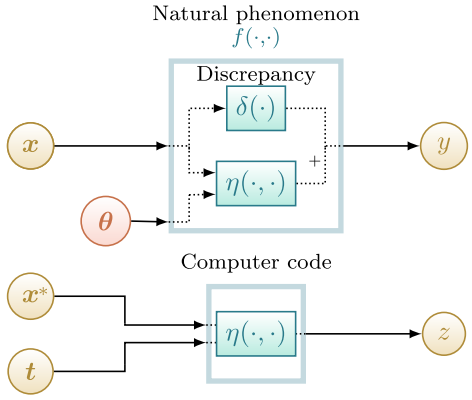

Figure 1: KOH calibration model.

In their Bayesian formulation, Kennedy and O'Hagan (2001) assume $\eta(\boldsymbol{x}, \boldsymbol{t})$ and $\delta(\boldsymbol{x})$ to be independent Gaussian processes (GPs); in other words, the KOH model assumes a given prior distribution over these functions, which takes the form of a GP. In addition, they assume a prior over $\boldsymbol{\theta}$, and they aim to characterize the posterior distribution over $\boldsymbol{\theta}$ given the observations of the real process and runs of the computer model. In order to keep the notation uncluttered, we denote by $\boldsymbol{\psi}$ the set of GP parameters for $\eta(\boldsymbol{x}, \boldsymbol{t})$ and $\delta(\boldsymbol{x})$, and we denote by U the collection of all input locations X, X^*, T. The marginal likelihood of the KOH model is

$$p(\boldsymbol{y}, \boldsymbol{z} \,|\, U, \boldsymbol{\psi}) = \int p(\boldsymbol{y} \,|\, \boldsymbol{\eta}_{\boldsymbol{\theta}} + \boldsymbol{\delta}) p(\boldsymbol{z} \,|\, \boldsymbol{\eta}^*) p(\boldsymbol{\delta} \,|\, X, \boldsymbol{\psi}) p(\boldsymbol{\eta}_{\boldsymbol{\theta}}, \boldsymbol{\eta}^* \,|\, \boldsymbol{\theta}, U, \boldsymbol{\psi}) p(\boldsymbol{\theta}) \mathrm{d}\boldsymbol{\eta}^* \, \mathrm{d}\boldsymbol{\delta} \, \mathrm{d}\boldsymbol{\eta}_{\boldsymbol{\theta}} \, \mathrm{d}\boldsymbol{\theta},$$

with vectors $\boldsymbol{\eta}^* = [\eta(\boldsymbol{x}_1^*, \boldsymbol{t}_1), \ldots, \eta(\boldsymbol{x}_N^*, \boldsymbol{t}_N)]^\top$, $\boldsymbol{\eta}_{\boldsymbol{\theta}} = [\eta(\boldsymbol{x}_1, \boldsymbol{\theta}), \ldots, \eta(\boldsymbol{x}_n, \boldsymbol{\theta})]^\top$ and $\boldsymbol{\delta} = [\delta(\boldsymbol{x}_1), \ldots, \delta(\boldsymbol{x}_n)]^\top$. The high dimensionality and the nontrivial dependence of this integrand with respect to the parameters of interest $\boldsymbol{\theta}$, makes their inference intractable,

thus requiring approximations. Similar considerations can be made for the predictive distribution, that is the one made by the KOH model at new input values. In the original work by Kennedy and O'Hagan (2001), inference is carried out using MCMC, which offers guarantees of convergence to the true posterior distribution over model parameters, but it can be extremely slow and impractical when the number of field observations and computer simulations is large. This is due to the poor scalability properties of GPs, which adds up to the need to repeatedly solve these within MCMC.

It is worth mentioning the identifiability issues associated with the KOH model, brought up in the discussion of the paper of Kennedy and O'Hagan (2001). Such issues arise due to the over-parameterization of the model, whereby it is possible to confound the effects of calibration parameters and model discrepancy. In particular, the KOH calibration model yields a joint posterior distribution over $\boldsymbol{\theta}$ and $\delta(\boldsymbol{x})$, but as the number of observations increases, the KOH model concentrates this posterior over the manifold

$$\mathcal{M} = \{(\boldsymbol{t}, \delta(\boldsymbol{x})) \,|\, \eta(\boldsymbol{x}, \boldsymbol{t}) + \delta(\boldsymbol{x}) = f(\boldsymbol{x}, \boldsymbol{t}) \quad \text{with} \quad \boldsymbol{x} \in X \cup X^*\}\,.$$

Brynjarsdóttir and O'Hagan (2014) nicely illustrate this problem as they study how removing the discrepancy would result in a model where the estimation of model parameters $\boldsymbol{\theta}$ is biased. The argument is that if $\eta(\boldsymbol{x}, \boldsymbol{\theta})$ does not model the real physical process exactly and there is no discrepancy, the estimate of $\boldsymbol{\theta}$ would be based on a misspecified model. Increasing the number of observations would not cure this fundamental issue with model misspecification. The conclusion is that discrepancy in the KOH model is necessary to hope for a sound inference over calibration parameters $\boldsymbol{\theta}$ of the computer model, and imposing good priors on $\boldsymbol{\theta}$ and $\delta(\boldsymbol{x})$ becomes of fundamental importance to mitigate the lack of identifiability. Alternatively, one can improve identifiability when multiple responses are available, and these are mutually dependent on the same set of calibration parameters $\boldsymbol{\theta}$ (Arendt et al., 2012). In the literature, there are works addressing the issue of identifiability of the KOH models with alternative formulations, such as loss minimization (Tuo and Wu, 2016) or frequentist formulations (Wong et al., 2017). Within this line of works, assuming that the optimal $\boldsymbol{\theta}$ is the optimizer of a loss function, Plumlee (2017) showed that the prior over $\delta(\boldsymbol{x})$ should be orthogonal to the gradient of the computer model. We remark that the issue of identifiability affects in a similar way the model that we propose here, and we dedicate part of the discussion to comment on how we can adopt current strategies from the literature to deal with this.

2.2 Gaussian Process and Random Features Expansions

A Gaussian process (GP) is a set of random variables such that any subset of these is jointly distributed as a Gaussian (Rasmussen and Williams, 2006). This definition makes them suitable for assigning priors over functions. Imposing a GP prior over a function $g_{\text{exact}}(\boldsymbol{u}), \boldsymbol{u} \in \mathcal{D} \subset \mathbb{R}^d$ means assigning a prior over the realizations of the function $[g_{\text{exact}}(\boldsymbol{u}_1), \ldots, g_{\text{exact}}(\boldsymbol{u}_l)]^\top$ at a set of l inputs $\boldsymbol{u}_1, \ldots, \boldsymbol{u}_l$, such that this is multivariate Gaussian; this is because of the properties of marginals of multivariate Gaussian distributions. What needs to be specified is a mean function and a covariance function $c(\boldsymbol{u}_i, \boldsymbol{u}_j)$, which determines how realizations of the function at different inputs

covary and therefore the properties of the functions that can be drawn from the GP. For simplicity, we assume a constant zero mean, but adding a parametric mean function is straightforward. Inference in models involving GPs quickly becomes intractable when l grows beyond a few thousands. The reason is that sampling from GPs and posterior inference requires algebraic operations with the covariance matrix obtained by evaluating the covariance function among all possible pairs of inputs. These operations usually involve $\mathcal{O}(l^3)$ operations, and require storing $\mathcal{O}(l^2)$ entries for the covariance matrix. In this work, we bypass these limitations by making a model approximation which lowers both complexities to $\mathcal{O}(l)$, as we discuss shortly.

Note that in this short presentation of GPs we assume that the function is univariate, that is $g_{\text{exact}}(\boldsymbol{u}) \in \mathbb{R}$. Extending this to multivariate functions $\boldsymbol{g}_{\text{exact}}(\boldsymbol{u}) \in \mathbb{R}^o$ is rather straightforward. What needs to be specified is a richer covariance structure, which is able to characterize the covariance between $(g_{\text{exact}})_r(\boldsymbol{u}_i)$ and $(g_{\text{exact}})_s(\boldsymbol{u}_j)$; see, e.g., Álvarez and Lawrence (2011) for an extensive treatment of these scenarios. When we assume a zero covariance across functions, we are effectively modeling each $g_r(\boldsymbol{u})$ as an independent GP. We are free to parameterize each GP separately, or to use a common covariance $c(\boldsymbol{u}_i, \boldsymbol{u}_j)$, so that covariance parameters are shared across the o functions. In this paper we prefer this latter approach to avoid introducing too many parameters, although we can easily incorporate more advanced modeling assumptions in our implementation.

For a large class of covariance functions, it is possible to show that draws from the GP prior are a linear combination of a possibly infinite number of basis functions with Gaussian-distributed weights (Neal, 1996; Rasmussen and Williams, 2006). This can be formulated for $\boldsymbol{u} \in \mathcal{D}$ as an infinite sum

$$g_{\text{exact}}(\boldsymbol{u}) = \boldsymbol{\phi}_\infty(\boldsymbol{u})^\top \boldsymbol{w}_\infty, \tag{2.2}$$

with $\boldsymbol{w}_\infty$ infinite dimensional random vector with i.i.d. standard normal components, and $\boldsymbol{\phi}_\infty$ the evaluation of an infinite set of basis functions at $\boldsymbol{u}$. The exact covariance of g_{exact} is readily obtained as $\forall \boldsymbol{u}, \boldsymbol{u}' \in \mathcal{D}$

$$c(\boldsymbol{u}, \boldsymbol{u}') = \mathbb{E}\left[\boldsymbol{\phi}_\infty(\boldsymbol{u})^\top \boldsymbol{w}_\infty \boldsymbol{w}_\infty^\top \boldsymbol{\phi}_\infty(\boldsymbol{u}')\right] = \boldsymbol{\phi}_\infty(\boldsymbol{u})^\top \boldsymbol{\phi}_\infty(\boldsymbol{u}'). \tag{2.3}$$

The infinite representation induced by the covariance function suggests a way to approximate GPs by means of a finite dimensional truncation of $\boldsymbol{\phi}_\infty(\boldsymbol{u})$, which we denote by $\boldsymbol{\phi}(\boldsymbol{u}) \in \mathbb{R}^p$, so that

$$c(\boldsymbol{u}, \boldsymbol{u}') = \boldsymbol{\phi}_\infty(\boldsymbol{u})^\top \boldsymbol{\phi}_\infty(\boldsymbol{u}') \approx \boldsymbol{\phi}(\boldsymbol{u})^\top \boldsymbol{\phi}(\boldsymbol{u}'). \tag{2.4}$$

The function $g_{\text{exact}}(\boldsymbol{u})$ is then approximated by

$$g_{\text{exact}}(\boldsymbol{u}) \approx g(\boldsymbol{u}) = \boldsymbol{\phi}(\boldsymbol{u})^\top \boldsymbol{w}. \tag{2.5}$$

When using GPs in modeling problems with l observations, the truncation has the advantage of avoiding the need to solve expensive algebraic operations with the covariance matrix. Instead, the truncation turns GPs into generalized linear models. In order to retain the probabilistic flavor of GPs, it is natural to treat these models in a Bayesian way,

and this requires algebraic operations with matrices of size $p \times p$ (cost $\mathcal{O}(p^3)$), while the complexity with respect to l is linear.

Random feature expansions (Rahimi and Recht, 2008; Lázaro-Gredilla et al., 2010) offer an elegant framework to construct a finite p-dimensional representation $\boldsymbol{\phi}(\boldsymbol{u})$, which are referred to as the *random features*. As a working example, throughout this paper we consider the Gaussian covariance (or kernel) function, also known as the squared exponential or radial basis covariance function:

$$c(\boldsymbol{u}, \boldsymbol{u}') = \sigma^2 \exp\left(-\frac{1}{2}(\boldsymbol{u} - \boldsymbol{u}')^\top A^{-1}(\boldsymbol{u} - \boldsymbol{u}')\right). \tag{2.6}$$

The symmetric positive definite matrix A controls the scaling and mixing of the inputs, whereas σ^2 controls the marginal variance of the GP. When the covariance is shift-invariant, it is possible to express the covariance as the Fourier transform of a positive measure (Rahimi and Recht, 2008). Applying the Fourier transform to the Gaussian covariance, and defining $\iota = \sqrt{-1}$, we obtain:

$$c(\boldsymbol{u}_i, \boldsymbol{u}_j) = \sigma^2 \int p(\boldsymbol{\omega}) \exp\left(\iota(\boldsymbol{u}_i - \boldsymbol{u}_j)^\top \boldsymbol{\omega}\right) d\boldsymbol{\omega}, \tag{2.7}$$

which immediately suggests that the distribution $p(\boldsymbol{\omega})$ is also Gaussian, and it has the form $p(\boldsymbol{\omega}) = \mathcal{N}(\mathbf{0}, A)$. By sampling from $p(\boldsymbol{\omega})$, we can approximate the integral in the Fourier formulation by means of Monte Carlo, thus obtaining a finite dimensional representation of the covariance:

$$c(\boldsymbol{u}_i, \boldsymbol{u}_j) \approx \frac{2\sigma^2}{N_{\mathrm{RF}}} \sum_{r=1}^{N_{\mathrm{RF}}/2} \exp\left(\iota(\boldsymbol{u}_i)^\top \tilde{\boldsymbol{\omega}}_r\right) \exp\left(-\iota(\boldsymbol{u}_j)^\top \tilde{\boldsymbol{\omega}}_r\right). \tag{2.8}$$

In this expression, we sampled $N_{\mathrm{RF}}/2$ values of $\boldsymbol{\omega}$, denoted by $\tilde{\boldsymbol{\omega}}_r$, and exploited the property of shift invariance to split the complex exponential in two parts. While the basis functions are complex, by realizing that the left-hand side is a real number, with simple trigonometric manipulations of the right-hand side, it is possible to express the previous equation in an equivalent form as:

$$c(\boldsymbol{u}_i, \boldsymbol{u}_j) \approx \frac{2\sigma^2}{N_{\mathrm{RF}}} \sum_{r=1}^{N_{\mathrm{RF}}/2} \left[\sin\left(\boldsymbol{u}_i^\top \tilde{\boldsymbol{\omega}}_r\right), \cos\left(\boldsymbol{u}_i^\top \tilde{\boldsymbol{\omega}}_r\right)\right] \left[\sin\left(\boldsymbol{u}_j^\top \tilde{\boldsymbol{\omega}}_r\right), \cos\left(\boldsymbol{u}_j^\top \tilde{\boldsymbol{\omega}}_r\right)\right]^\top. \tag{2.9}$$

Therefore, introducing $\boldsymbol{\phi} : \mathbb{R}^{N_{\mathrm{RF}}} \to \mathbb{R}^{N_{\mathrm{RF}}}$ as the element-wise application of sine (for the first $N_{\mathrm{RF}}/2$ components) and cosine (for the last $N_{\mathrm{RF}}/2$ components), the resulting basis functions are

$$\boldsymbol{\phi}(\boldsymbol{u}) = \sqrt{\frac{2\sigma^2}{N_{\mathrm{RF}}}}\left[\sin\left(\boldsymbol{u}^\top \Omega\right), \cos\left(\boldsymbol{u}^\top \Omega\right)\right]^\top, \tag{2.10}$$

where $\Omega = [\boldsymbol{\omega}_1, \ldots, \boldsymbol{\omega}_{N_{\mathrm{RF}}/2}]$, and the sin and cos functions are applied element-wise to their argument. The basis functions are also called random features, because they

are obtained by multiplying the inputs $\boldsymbol{u}_i$ with a random matrix Ω, followed by the application of a nonlinearity. Similar considerations can be made, for instance, for the Matérn covariance, where the $\boldsymbol{\omega}$'s are sampled according to a multivariate Student-t distribution. See also Cho and Saul (2009) for an alternative derivation showing how the arc-cosine covariance of order one can be approximated in a similar fashion by sampling $\boldsymbol{\omega}$ from $p(\boldsymbol{\omega}) = \mathcal{N}(\mathbf{0}, A)$ and by employing a rectified linear unit nonlinearity ($h(x) = x$ if $x > 0$ and $h(x) = 0$ otherwise), which is very popular in the literature of deep neural networks.

2.3 Deep Gaussian Processes and Random Feature Expansions

A Deep Gaussian Process (DGP) is defined as a composition of functions:

$$g_{\text{deep}}(\boldsymbol{u}) = g^{(L)}(g^{(L-1)}(\dots(g^{(1)}(\boldsymbol{u}))\dots)), \tag{2.11}$$

where each function $g^{(i)}(\cdot)$ is assigned a GP prior (Damianou and Lawrence, 2013; Neal, 1996). Again, for simplicity we focus our presentation on univariate GPs, but we will discuss ways in which we can deal with multivariate GPs shortly. The composition can be interpreted as a way to feed the output of $g^{(i)}$ as input to another $g^{(i+1)}$. In a parallel with deep neural networks, each GP can be thought of as a "layer". The composition operation makes DGP priors substantially different from GPs; that is, the realizations of $g^{(L)}(\boldsymbol{u})$ at U are no longer multivariate Gaussian in general. We refer the reader to Neal (1996); Duvenaud et al. (2014); Matthews et al. (2018) for some in-depth discussions and illustrations on the composition of GPs.

The DGP prior induced by the choice of GP priors over the functions in the composition can be used as a prior over functions in statistical models. After choosing an appropriate likelihood function, one is usually interested in optimizing all model parameters, which include the covariance parameters of the GPs at all layers, characterizing the posterior over $g^{(L)}(\boldsymbol{u})$ at U, and making predictions for any $\boldsymbol{u}_*$. For DGPs, these tasks are analytically intractable due to the nontrivial dependence introduced by the composition. Most of the literature on DGPs extends approximations and inference techniques developed for "shallow" GPs. For instance, Hensman and Lawrence (2014); Salimbeni and Deisenroth (2017) extend the use of Nyström-type approximations (also known as inducing points approximations) to DGPs and carry out inference using variational techniques, whereas Bui et al. (2016) employs expectation propagation.

This work focuses on random feature expansions for DGPs instead, which were proposed and studied in Gal and Ghahramani (2016); Cutajar et al. (2017). In this framework, each GP in the composition is approximated by means of random features, as shown in the previous section. With this approximation, each GP layer becomes a linear model with a given distribution over the weights. Denoting by $\mathbf{a}$ the input to layer (i), the GP at the i^{th} layer approximated with random features implements the following operations:

$$\boldsymbol{\phi}^{(i)}(\mathbf{a}) = \sqrt{\frac{2(\sigma^2)^{(i)}}{{N_{\text{RF}}}^{(i)}}}[\sin(\Omega^{(i)}\mathbf{a})^\top, \cos(\Omega^{(i)}\mathbf{a})^\top]^\top, \qquad g^{(i)}(\mathbf{a}) = (\boldsymbol{\phi}^{(i)})^\top \mathbf{w}^{(i)}. \tag{2.12}$$

With this approximation, each GP layer can be seen as a two-layer neural network. The first layer implements a multiplication by a random matrix $\Omega^{(i)}$ and applies a nonlinearity by means of trigonometric functions. The second layer implements a linear combination of the inputs. Therefore, composing these approximate GPs gives rise to a particular form of a Bayesian deep neural network. In traditional deep neural networks nonlinearities are applied at each layer and all the weights are optimized; in the approximate DGP viewed as a Bayesian deep neural network, nonlinearities are applied every other layer, and only the $\mathbf{w}^{(i)}$ weights are inferred, whereas the $\Omega^{(i)}$ are random. See Cutajar et al. (2017) for an in-depth discussion on the connection with Bayesian deep neural networks and other ways in which $\Omega^{(i)}$ can be treated in order to improve performance.

The model approximation with random features bypasses the challenges of carrying out inference having to deal with the composition of GPs, but the composition of the resulting linear models is still intractable from a Bayesian perspective due to the nonlinearities introduced by the basis functions. In the next section, we present variational inference as a way to derive a tractable and scalable inference scheme for DGPs approximated with random features.

2.4 Stochastic Variational Inference

In this work, we make use of Variational Inference (VI) techniques to carry out inference over model parameters. We give a brief overview of VI here, and we will show how this is applied to the proposed calibration model in the next section. We consider a modeling problem for a set of l pairs of input/labels observations $(\boldsymbol{u}_i, v_i)$, with $\boldsymbol{u} \in \mathbb{R}^d$, and $v \in \mathbb{R}$. Let $U = [\boldsymbol{u}_1, \ldots, \boldsymbol{u}_l]^\top$ and $\boldsymbol{v} = [v_1, \ldots, v_l]^\top$. Imagine developing a statistical model with parameters Θ and with likelihood function $p(\boldsymbol{v}|U, \Theta)$, and assume a prior $p(\Theta)$.

VI is useful when the posterior over Θ, that is $p(\Theta|\boldsymbol{v}, U)$ is intractable. In VI, an approximation $q(\Theta) \in \mathcal{Q}$ to the posterior is introduced, and the objective is to make it as close as possible to the actual posterior $p(\Theta|\boldsymbol{v}, U)$. The standard way to do so in VI is to set up the following optimization problem:

$$\underset{q(\Theta)\in\mathcal{Q}}{\operatorname{argmin}} \left\{ \mathrm{D}_{\mathrm{KL}} \left[q(\Theta) \parallel p(\Theta|\boldsymbol{v}, U) \right] \right\}, \tag{2.13}$$

where D_{KL} is the Kullback-Leibler divergence measuring how different the two distributions are. With simple manipulations, it is possible to show that an equivalent problem is the one of maximizing the following lower bound to the log-marginal likelihood with respect to $q(\Theta)$ (see, e.g., Jordan et al. 1999; Graves 2011; Blei et al. 2017):

$$\log p(\boldsymbol{v}|U) \geq \mathbb{E}_{q(\Theta)}[\log(p(\boldsymbol{v}|\Theta, U))] - \mathrm{D}_{\mathrm{KL}} \left[q(\Theta) \parallel p(\Theta) \right]. \tag{2.14}$$

In other words, a candidate $q(\Theta)$ providing the highest lower bound also minimizes the divergence to the exact posterior.

With an expression for the lower bound of the marginal likelihood, we can now attempt to maximize it with respect to $q(\Theta)$, which generally means to optimize it

w.r.t. its parameters. Therefore, the family of distributions, $\mathcal{Q}$, for the candidate approximation $q(\Theta) \in \mathcal{Q}$ needs to be chosen before inference. While the approximation is constrained by the choice of the family $\mathcal{Q}$, the complexity of $\mathcal{Q}$ impacts the complexity of the lower bound maximization. The trade-off between the inference speed and the quality of the approximation is an active research domain. For instance Rezende and Mohamed (2015) and Liu and Wang (2016) increase the expressiveness of the variational distribution while keeping the inference tractable, at the cost of increasing the number of parameters of $q(\Theta)$.

The lower bound contains two terms: the first is a model fitting term, whereas the second is a regularization term which penalizes approximations that deviate too much from the prior. This D_{KL} term can be computed analytically when priors and approximate posteriors have particular forms. For example, when $p_1(x) = \mathcal{N}(\mu_1, \sigma_1^2)$ and $p_2(x) = \mathcal{N}(\mu_2, \sigma_2^2)$, the Kullback-Leibler divergence between the two has the form:

$$\mathrm{D}_{\mathrm{KL}}\left(p_1(x) \| p_2(x)\right) = \frac{1}{2}\left[\log\left(\frac{\sigma_2^2}{\sigma_1^2}\right) - 1 + \frac{\sigma_1^2}{\sigma_2^2} + \frac{(\mu_1 - \mu_2)^2}{\sigma_2^2}\right]. \tag{2.15}$$

The first term in the lower bound (2.14) depends on $q(\Theta)$ through the expectation of the log-likelihood. This complication is usually bypassed by employing stochastic optimization using Monte Carlo:

$$\mathbb{E}_{q(\Theta)}[\log(p(\boldsymbol{v}|\Theta, U))] \approx \frac{1}{N_{\mathrm{MC}}} \sum_{k=1}^{N_{\mathrm{MC}}} \log\left(p(\boldsymbol{v}|\tilde{\Theta}_{(k)}, U)\right), \tag{2.16}$$

with $\tilde{\Theta}_{(k)}$ i.i.d. samples from $q(\Theta)$. The Monte Carlo approximation is unbiased, and so it is its derivative with respect to any of the parameters of $q(\Theta)$. This means that we can employ stochastic gradient optimization to adapt the parameters of $q(\Theta)$ to maximize the lower bound with guarantees to reach a local optimum of the objective (Robbins and Monro, 1951; Graves, 2011). The only precaution to take to make this viable, is to reparameterize the samples from $q(\Theta)$ using the so-called reparameterization trick (Kingma and Welling, 2014). In its simplest form, assuming a fully factorized Gaussian posterior over all parameter components $\theta_\ell \in \Theta$, the expression $(\tilde{\theta}_\ell)_{(k)} = \mu_\ell + (\tilde{\epsilon}_\ell)_{(k)}\sigma_\ell$ separates out the stochastic ($(\tilde{\epsilon}_\ell)_{(k)} \sim \mathcal{N}(0,1)$) and deterministic ($\mu_\ell$ and σ_ℓ) components in the way samples from the approximate posterior are generated. In this way, the lower bound can be differentiated with respect to the variational parameters μ_ℓ and σ_ℓ (with the $(\tilde{\epsilon}_\ell)_{(k)}$ variables fixed), and it is therefore possible to perform gradient-based optimization (Graves, 2011; Kingma and Welling, 2014; Cutajar et al., 2017). The gradients are stochastic because the $(\tilde{\epsilon}_\ell)_{(k)}$ variables are random, but the Monte Carlo estimate of the objective guarantees that the estimate of the gradient is unbiased, allowing for the use of stochastic gradient-based optimization (Robbins and Monro, 1951). This is the reason why this implementation of VI is also referred to as stochastic variational inference (SVI).

Finally, it is worth noting that it is possible to considerably reduce the variance of the stochastic gradients, thus increases convergence speed of the optimization by means of the so-called *local reparameterization trick* (Kingma et al., 2015). In this approach, instead of sampling $(\tilde{\epsilon}_\ell)_{(k)}$, one samples from the distribution of the product of the inputs to a layer and samples from $q(\Theta)$; see Kingma et al. (2015) for more details.

Mini-Batch-Based Learning and Automatic Differentiation Part of the huge success of deep learning is due to the exploitation of mini-batch-based optimization and automatic differentiation (Graves, 2011). The former enables scalability, as the model is updated by iteratively processing subsets of data. The latter, instead, allows one to tremendously simplify the implementation of complex models, as one has to implement the objective function and automatic differentiation takes care of computing its derivatives based on the graph of computations. Naïvely operating with mini-batches in GPs ignores the covariance among observations, which is crucial for effective GP modeling.

The proposed GP and DGP approximation and SVI allow us to exploit mini-batch-based optimization. Assuming that the likelihood factorizes across observations, $p(\boldsymbol{v}|\Theta, U) = \prod_j^n p(v_j|\Theta, U)$, the terms within the approximation of the fitting term in the lower bound can be estimated without bias by selecting $\mathcal{I}$, a set of m out of n indices (Graves, 2011)

$$\mathbb{E}_{q(\Theta)}[\log(p(\boldsymbol{v}|\Theta, U))] \approx \frac{1}{N_{\mathrm{MC}}} \sum_{i=1}^{N_{\mathrm{MC}}} \frac{n}{m} \sum_{\ell \in \mathcal{I}} \log\left(p(u_\ell|\tilde{\Theta}_{(k)}, U)\right). \tag{2.17}$$

This approximation introduces an extra level of stochasticity in the optimization (without introducing bias), but it allows one to scale the inference of these models to virtually any number of observations; previous work has reported results on DGPs for 10^7 observations with a single-machine implementation (Cutajar et al., 2017). Another important remark is that this approximation and inference approach for DGPs can be implemented relying exclusively on matrix-matrix and matrix-vector products, which can be accelerated by using GPU-type hardware.

3 DGPs for Calibration of Computer Models

In this section we present our contribution, which we refer to as DGP-CAL. We begin by observing that the KOH calibration model can be seen as a special case of a DGP, and this allows us to generalize the original formulation of the KOH model to more flexible ones. We then show how we can leverage the advances in approximation and inference for DGPs presented in the previous sections, namely random feature expansions and stochastic variational inference, in order to obtain a scalable framework for calibration, while retaining the flexibility offered by the use of DGPs. We conclude the section by discussing implementation details.

3.1 Generalization of the KOH Calibration Model as a DGP

The original formulation of the KOH calibration model involves the use of GPs to emulate the computer model and to model the additive discrepancy. As pointed out by Kennedy and O'Hagan (2001), additive discrepancy is somewhat specific and it can be generalized (see, e.g., Qian and Wu 2008). We propose to do so by assuming that the function underlying the real process is obtained by a warping function γ applied to the emulator:

$$f(\boldsymbol{x}, \boldsymbol{t}) = \gamma\left(\eta(\boldsymbol{x}, \boldsymbol{t}), \boldsymbol{x}\right). \tag{3.1}$$

We retrieve the KOH formulation (2.1) when the warping applies the identity to $\eta(\boldsymbol{x}, \boldsymbol{t})$ and adds it to a GP on $\boldsymbol{x}$. In other words, we can think of the function $f(\boldsymbol{x}, \boldsymbol{t})$ in the KOH model as a composition of two functions; the first is a GP that, given the inputs $\boldsymbol{x}$ and $\boldsymbol{t}$, yields $\eta(\boldsymbol{x}, \boldsymbol{t})$, whereas the second applies a linear combination of η and another GP $\delta(\boldsymbol{x})$ with fixed unit weights.

In the original formulation of the KOH model, the discrepancy between the computer code and the real process is modeled through the additive discrepancy term $\delta(\boldsymbol{x})$. In the proposed generalization of the KOH model, instead, we assume a GP prior over $\gamma\left(\eta(\boldsymbol{x}, \boldsymbol{t}), \boldsymbol{x}\right)$, so that we apply a warping as a function of $\boldsymbol{x}$. Similarly to the original KOH model, the analysis of the warping function $\gamma\left(\eta(\boldsymbol{x}, \boldsymbol{t}), \boldsymbol{x}\right)$ allows one to reason about the discrepancy between the computer model and the real process.

A further possible extension, which we implement in our work, is to increase model flexibility by letting $\eta(\boldsymbol{x}, \boldsymbol{t})$ and/or $\gamma(\eta(\boldsymbol{x}, \boldsymbol{t}), \boldsymbol{x})$ to be modeled as DGPs instead of GPs. The deep extension is particularly useful when the emulator or the real process exhibit a space-dependent behavior that is difficult to model by designing appropriate covariance functions. DGPs offer a way to learn such nonstationarities from data, so this is particularly appealing in such challenging applications. We will give illustrations of DGPs and the generalized formulation in the experiments. Thanks to the possibility to employ random feature approximations of the DGPs in the generalized model, we can obtain a scalable framework for calibration as discussed next.

3.2 Model Approximation Using Random Features

In this section, we discuss how to employ the random feature approximation to make the KOH model and its generalization suitable for variational inference. We start from the KOH model, assuming that $\eta(\boldsymbol{x}, \boldsymbol{t})$ and $\delta(\boldsymbol{x})$ are modeled as GPs with the Gaussian covariance in (2.6); we denote their anisotropy matrices by A_η and A_δ and their marginal variances σ_η^2 and σ_δ^2, respectively. By applying the random feature expansion detailed in Section 2.2 we obtain

$$\eta(\boldsymbol{x}, \boldsymbol{\theta}) = \boldsymbol{\phi}_\eta(\boldsymbol{x}, \boldsymbol{\theta})^\top \boldsymbol{w}_\eta = \sigma_\eta \, \boldsymbol{\phi}\left(\Omega_\eta \begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{\theta} \end{bmatrix}\right)^\top \boldsymbol{w}_\eta, \tag{3.2}$$

$$\delta(\boldsymbol{x}) = \boldsymbol{\phi}_\delta(\boldsymbol{x})^\top \boldsymbol{w}_\delta \;\; = \sigma_\delta \, \boldsymbol{\phi}\left(\Omega_\delta \boldsymbol{x}\right)^\top \boldsymbol{w}_\eta. \tag{3.3}$$

The feature maps $\boldsymbol{\phi}_\eta$ and $\boldsymbol{\phi}_\delta$ use the functions $\boldsymbol{\phi} : \mathbb{R}^{N_{\mathrm{RF}}} \to \mathbb{R}^{N_{\mathrm{RF}}}$ given in (2.10). The elements of $\boldsymbol{w}_\eta$ and $\boldsymbol{w}_\delta$, of size N_{RF}, have i.i.d. standard normal priors, whereas the matrices Ω_η, Ω_δ of size $N_{\mathrm{RF}} \times (d_1 + d_2)$, $N_{\mathrm{RF}} \times d_1$, have i.i.d. normal rows, with covariance dependent on the positive definite matrices A_η and A_δ; in particular, the columns of Ω_η and Ω_δ are i.i.d. $\mathcal{N}(\boldsymbol{0}, A_\eta)$ and $\mathcal{N}(\boldsymbol{0}, A_\delta)$, respectively. Figure 2 represents the model (using a neural network-like diagram) according to (2.1), (3.2), and (3.3).

It is straightforward to extend this formulation to model $\eta(\boldsymbol{x}, \boldsymbol{t})$ and $\delta(\boldsymbol{x})$ with DGPs instead of GPs, by applying the random feature expansion to each GP layer. Assuming that each DGP has L_η and L_δ layers, in this case, $\eta(\boldsymbol{x}, \boldsymbol{t})$ and $\delta(\boldsymbol{x})$ are approximated by

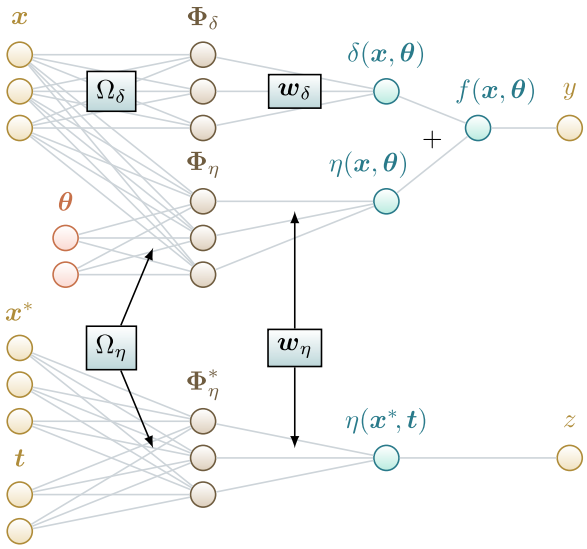

Figure 2: Neural Network representation of the proposed approximation to the KOH model. (3.2) and (3.3) formulate random feature expansions for $\eta(\boldsymbol{x},\boldsymbol{t})$ and $\delta(\boldsymbol{x})$.

a Bayesian deep neural network, where each GP layer is approximated by:

$$\begin{aligned}
\eta(\boldsymbol{x},\boldsymbol{\theta}) &= \boldsymbol{\phi}_\eta^{(L_\eta)}(\boldsymbol{a}^{(L_\eta)})^\top \boldsymbol{w}_\eta^{(L_\eta)} \qquad \boldsymbol{\phi}_\eta^{(L_\eta)}(\boldsymbol{a}^{(L_\eta)}) = \boldsymbol{\phi}(\Omega_\eta \boldsymbol{a}^{L_\eta}),\\
\boldsymbol{a}^{L_\eta} &= \boldsymbol{\phi}_\eta^{(L_\eta-1)}(\boldsymbol{a}^{(L_\eta-1)})^\top \boldsymbol{w}_\eta^{(L_\eta-1)} \qquad \boldsymbol{\phi}_\eta^{(L_\eta)}(\boldsymbol{a}^{(L_\eta-1)}) = \boldsymbol{\phi}(\Omega_\eta \boldsymbol{a}^{L_\eta-1}),\\
&\ldots = \ldots\\
\boldsymbol{a}^{(2)} &= \boldsymbol{\phi}_\eta^{(1)}(\boldsymbol{x},\boldsymbol{\theta})^\top \boldsymbol{w}_\eta^{(1)} \qquad \boldsymbol{\phi}_\eta^{(1)}(\boldsymbol{x},\boldsymbol{\theta}) = \boldsymbol{\phi}\left(\Omega_\eta^{(1)} \begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{\theta} \end{bmatrix}\right).
\end{aligned} \tag{3.4}$$

Similarly, we can construct a random feature approximation for a DGP modeling $\delta(\boldsymbol{x})$ in the KOH model, or apply the same construction for the warping function $\gamma\left(\eta(\boldsymbol{x},\boldsymbol{t}),\boldsymbol{x}\right)$ in the generalized version of the KOH model. The advantage of the proposed approximation using random features is that it enables the use of stochastic variational inference presented in Section 3.3, as discussed next.

3.3 Inference of the Approximate DGP Calibration Model

Again, we focus on the formulation of the KOH model with the discrepancy term $\delta(\boldsymbol{x})$ instead of the warping $\gamma(\eta(\boldsymbol{x},\boldsymbol{t}),\boldsymbol{x})$, but it is easy to follow the same derivation for this latter case. We make use of variational inference (VI), so we approximate the posterior distribution over all model parameters $\boldsymbol{w}_\eta$, $\boldsymbol{w}_\delta$, and $\boldsymbol{\theta}$ by introducing a variational posterior $q(\boldsymbol{w}_\eta,\boldsymbol{w}_\delta,\boldsymbol{\theta})$, see Section 2.4. We assume $q(\boldsymbol{w}_\eta,\boldsymbol{w}_\delta,\boldsymbol{\theta})$ to be Gaussian and

completely factorized across parameters, that is

$$q(\boldsymbol{w}_\eta, \boldsymbol{w}_\eta, \boldsymbol{\theta}) = \prod_{i=1}^{N_{\text{RF}}} \prod_{j=1}^{d_{\text{out}}} q(\boldsymbol{w}_{\eta,ij}) q(\boldsymbol{w}_{\delta,ij}) \prod_{\ell=1}^{d_2} q(\theta_\ell), \tag{3.5}$$

although the factorization assumption can be relaxed. With the assumption that $q(\boldsymbol{w}_\eta, \boldsymbol{w}_\eta, \boldsymbol{\theta})$ factorizes across parameters, we deal with computations and storage which are linear in the number of model parameters. Relaxing the factorization means to assume that certain groups of parameters have nonzero covariance. For example, one could assume $q(\boldsymbol{w}_{\eta,\cdot j}) = \mathcal{N}(\mu_{\eta,j}, \Sigma_{\eta,j})$ and parameterize $\Sigma_{\eta,j} = L_{\eta,j} L_{\eta,j}^\top$ to preserve positive-definiteness by optimizing $L_{\eta,j}$. It is easy to verify that this choice requires $\mathcal{O}(N_{\text{RF}}^2)$ storage and $\mathcal{O}(N_{\text{RF}}^3)$ computations.

Following the principles of VI, we need to derive a lower bound to the marginal likelihood of the model and maximize it with respect to the distribution $q(\boldsymbol{w}_\eta, \boldsymbol{w}_\delta, \boldsymbol{\theta})$. In practice, this problem turns into the optimization of the lower bound with respect to the parameters that govern $q(\boldsymbol{w}_\eta, \boldsymbol{w}_\delta, \boldsymbol{\theta})$. Taking the lower bound (2.14) and its unbiased Monte Carlo and mini-batch-based approximation (2.17), we now adapt this to our calibration model. In this adaptation we realize that there are two types of input points, namely observations X and computer runs $[X^*, T]$; note that the shapes of X, X^* and T does not allow a concatenation of these three matrices in a single matrix. One possibility is to apply mini-batch on the union of the data sets. However we do not recommend this procedure without ensuring that each category ("Observations" versus "Runs") gets sufficiently represented. For example a uniform sampling blind to the category of the input points with $n << N$, would make the number of sampled observations vary a lot from one iteration to the other and may sometimes sample none of them. A workaround is to draw two index sets, $\mathcal{I} \subset \{1, \ldots, n\}$ and $\mathcal{J} \subset \{1, \ldots, N\}$ of sizes m and M. The fitting term of the lower bound can now be approximated as

$$\mathcal{E} := \mathbb{E}_{q(\boldsymbol{w}_\eta, \boldsymbol{w}_\delta, \boldsymbol{\theta})} \left[\log(p(\boldsymbol{y}, \boldsymbol{z} | \boldsymbol{w}_\eta, \boldsymbol{w}_\delta, \boldsymbol{\theta}, U)) \right] \tag{3.6}$$

$$\approx \frac{1}{N_{\text{MC}}} \sum_{k=1}^{N_{\text{MC}}} \left[\frac{n}{m} \sum_{i \in \mathcal{I}} \log\left(y_i | \boldsymbol{\psi}, X, (\tilde{\boldsymbol{w}}_\eta)_{(k)}, (\tilde{\boldsymbol{w}}_\delta)_{(k)}, \tilde{\boldsymbol{\theta}}_{(k)} \right) + \frac{N}{M} \sum_{j \in \mathcal{J}} \log\left(z_j | \boldsymbol{\psi}, X^*, T, (\tilde{\boldsymbol{w}}_\eta)_{(k)} \right) \right], \tag{3.7}$$

with $(\tilde{\boldsymbol{w}}_\eta)_{(k)}, (\tilde{\boldsymbol{w}}_\delta)_{(k)}, \tilde{\boldsymbol{\theta}}_{(k)}$ i.i.d. samples from $q(\boldsymbol{w}_\eta, \boldsymbol{w}_\delta, \boldsymbol{\theta})$. The regularization term can be easily calculated when both priors and posteriors are Gaussian using (2.15).

3.4 Implementation Details

Considering the large number of parameters to optimize, the optimization procedure is divided into stages. We first focus on the computer model response; all parameters are fixed except the ones influencing the prediction of $\boldsymbol{z}$, i.e. σ_z, the means and variances of the components of $\boldsymbol{w}_\eta$, and the GP/DGP parameters of $\eta(\boldsymbol{x}, \boldsymbol{t})$. In the second stage

all others parameters are freed for inferring $\boldsymbol{y}$ and $\boldsymbol{\theta}$ jointly, i.e. adding the weights and hyperparameters of $\delta(\boldsymbol{x})$ or $\gamma(\eta(\boldsymbol{x},\boldsymbol{t}),\boldsymbol{x})$. Within each stage, we first optimize the means and variances of $\boldsymbol{\theta}$, and then all parameters jointly with a smaller learning rate.

For initializing the weights of the DGP (or GP) $\eta(\boldsymbol{x},\boldsymbol{t})$, we use the methodology proposed by Rossi et al. (2019), which is a scalable, layer-wise initialization strategy based on Bayesian linear regression. The idea is to perform a series of regression tasks mapping each layer to the labels, so that sensible initial parameters for the variational distribution can be obtained. Starting from the first layer, we perform Bayesian linear regression from the inputs X^* and T to the labels $\boldsymbol{z}$, so that we can estimate the posterior over the parameters at the first layer. We then freeze this distribution and compute the output of the first layer with the given input data X^* and T. This result is then used as the input to the second layer, for which the corresponding linear model is estimated by performing regression to the labels $\boldsymbol{z}$. We then proceed iteratively up to the last layer. Intuitively, this initialization promotes configurations where the first layers are already capable of obtaining sensible regression result, while the layers closer to the labels serve as refinements.

We define the routine $\boldsymbol{\mu},\Sigma \leftarrow \mathrm{LM}(\mathrm{X}_{\mathrm{LM}},\boldsymbol{y}_{\mathrm{LM}})$, which performs Bayesian linear regression on a set of input-output pairs X_{LM}, $\boldsymbol{y}_{\mathrm{LM}}$. The routine returns the mean and the covariance of the posterior distribution over the weights $\boldsymbol{w}$ in the instrumental regression $\boldsymbol{y}_{\mathrm{LM}} := \boldsymbol{z} = X_{\mathrm{LM}}\boldsymbol{w} + \sigma_\eta^2\boldsymbol{\varepsilon}$, where σ_η^2 is the variance of the likelihood function, and the prior for the components of $\boldsymbol{w},\boldsymbol{\varepsilon}$ is i.i.d. $\mathcal{N}(0,1)$. With these definitions, the initialization is reported in Algorithm 1. The posterior distribution in Bayesian linear regression has full covariance in general, whereas the assumed posterior is fully factorized. In this case, we match the optimal factorized Gaussian distribution to the actual posterior using the Kullback-Leibler divergence, which explains the assignment to $q(W^{(\ell)}_{\eta,:,i})$ in the last line of Algorithm 1. The procedure can easily exploit mini-batching, as reported in Algorithm 1, and it can operate with stochastic optimization, thus making it suitable for large-scale problems. We refer the reader to Rossi et al. (2019) for more details.

4 Experiments

In this section we validate DGP-CAL on a number of calibration problems. In each experiment, we specify whether DGP-CAL uses the additive structure in (2.1), as in the KOH formulation, or the general one in (3.1); we also specify when the model is tested with DGPs instead of GPs (the default). The experiments have the following setup. The likelihoods $p(y_i|f_i)$ and $p(z_j|\eta_j^*)$ are Gaussian with variances σ_y^2 and σ_z^2 treated as hyperparameters within $\boldsymbol{\psi}$. All covariance functions of $\eta(\boldsymbol{x},\boldsymbol{t})$ and $\delta(\boldsymbol{x})$ are Gaussian, except for the comparative experiment in Section 4.2 where a Matérn kernel is used. The variational posteriors $q(W_\eta)q(W_\delta)q(\boldsymbol{\theta})$ and the prior $p(\boldsymbol{\theta})$ are Gaussian.

Competing Methods A large literature is devoted to the practicalities of numerically challenging applications. Gramacy et al. (2015) use local approximate GP modeling and calibrate parameters by solving a derivative-free maximization of a likelihood term. Pratola and Higdon (2016) handle large problems using a Bayesian sum-of-trees regression

Input:

- DGP $\eta = g^{(L)}(\boldsymbol{g}^{(L-1)}(\ldots(\boldsymbol{g}^{(1)}(\boldsymbol{x}, \boldsymbol{t}))))$ with variational distribution of the weights $\boldsymbol{w}_\eta = \{W_\eta^{(1)}, \ldots, W_\eta^{(L)}\}$ set to the prior $\mathcal{N}(0,1)$ i.i.d. for all components.
- Computer runs X^*, T with outputs $\boldsymbol{z}$ organized in mini-batches of size M.

Result: A distribution $q(\boldsymbol{w}_\eta)$ to start the lower bound maximization with.

for *each layer index* ℓ **do**
  for *each output component* i *of* $\boldsymbol{g}^{(\ell)}$ **do**
    • $X_{\mathrm{b}}^*, T_{\mathrm{b}}, \boldsymbol{z}_{\mathrm{b}} \leftarrow$ next data batch;
    • Propagate $X_{\mathrm{b}}^*, T_{\mathrm{b}}$ through a sample path of η and save X_{LM} as:
    $X_{\mathrm{LM}} \leftarrow [\boldsymbol{g}^{(\ell-1)}(\ldots(\boldsymbol{g}(\boldsymbol{x}_{\mathrm{b},j}^*, \boldsymbol{t}_{\mathrm{b},j})))^\top]_{j=1,\ldots,M}$
    (X_{LM} is the output of the previous layer $\boldsymbol{g}^{(\ell-1)}$, uses $[X_{\mathrm{b}}^*, T_{\mathrm{b}}]$ for $\ell = 1$);
    • $\boldsymbol{y}_{\mathrm{LM}} \leftarrow \boldsymbol{z}_{\mathrm{b}}$;
    • $\boldsymbol{\mu}, \Sigma \leftarrow \mathrm{LM}(X_{\mathrm{LM}}, \boldsymbol{y}_{\mathrm{LM}})$;
    • Initialize the variational distribution of the i^{th} column of $W_\eta^{(\ell)}$
    $q(W_{\eta,:,i}^{(\ell)}) \leftarrow \mathcal{N}\left(\boldsymbol{\mu}, \mathrm{diag}\left(\frac{1}{\Sigma_{11}^{-1}}, \ldots, \frac{1}{\Sigma_{MM}^{-1}}\right)\right)$;
  end
end

Algorithm 1: Initialization for the approximate posterior distribution over η.

for modeling data from computer model and the real process jointly. More recently in Gu and Wang (2018) and Gu (2018), calibration is performed within a Bayesian framework by defining a prior distribution directly on the L_2 norm of the discrepancy. Xie and Xu (2018) sample from the posterior distribution over calibration parameters by minimizing the L_2 norm of a sample path of the discrepancy (similarly to Wong et al. 2017). These authors provided easy-to-use code in R or C++ packages. We will refer to these methods as LaGP, Sum-of-trees, Robust, and Projected, respectively

4.1 Illustrative Example

We illustrate DGP-CAL on a calibration problem with one variable and one calibration input. As a first test, the prior and hyperparameters used to generate the data set are assumed to be known, with $\theta \sim \mathcal{N}(0,1)$, $\sigma_\eta = 1$, $A_\eta = \frac{1}{2}I$, $\sigma_\delta = \frac{2}{10}$, $A_\delta = \frac{1}{20}$. We choose locations for $N = 7$ computer runs and $n = 4$ observations from the real process in a space filling manner in $[0,1] \times [-\frac{5}{2}, \frac{5}{2}]$. The output vector $\boldsymbol{z}$ of the computer model at $(\boldsymbol{x}_i^*)_{i=1,\ldots,N}$ is sampled from its prior distribution. In order to determine the real observations $\boldsymbol{y}$, we first sample $p(\theta)$ to get θ_{true} and then a sample path of $\delta(\boldsymbol{x})$. The GP priors of $\eta(\boldsymbol{x}, \boldsymbol{t})$ and $\delta(\boldsymbol{x})$ are approximated with $N_{\mathrm{RF}} = 50$ random features through (3.2) and (3.3), and the observations are computed using (2.1). The results of DGP-CAL are displayed in Figure 3. In the first and the third panels, we see that the posterior of θ obtained analytically by integrating out $\boldsymbol{w}_\eta$ and $\boldsymbol{w}_\delta$ has its mass

concentrated around the true value ≈ 0.8, where there is a (color) match between $\boldsymbol{z}$ (the dots) and $\boldsymbol{y}$ (the lines). The variational posterior (blue line) offers a reasonable approximation of the true posterior.

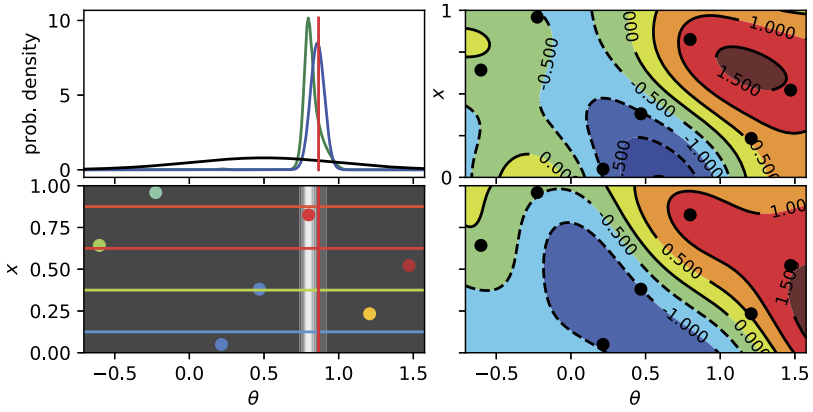

Figure 3: **Top-left:** the prior (black), the analytical posterior (green) and the variational posterior (blue) distributions of θ and the actual value used to generate $\boldsymbol{y}$ (red). **Top-right:** the true response η used to generate the data set of this example and the locations of the computer runs (dots). **Bottom-left:** shows $\boldsymbol{y}$ as horizontal lines in $\mathcal{D}_1 \times \mathcal{D}_2$. The colors of the lines correspond to the values y_i. The dots represent the computer runs $\boldsymbol{z}$. The grey levels represent the posterior distribution of θ (also displayed in the top-left panel). **Bottom-right:** the posterior mean of η.

4.2 Model Calibration in Cell Biology

We apply DGP-CAL to a biological application, which has been previously studied in Plumlee (2017) and Xie and Xu (2018). The output is the normalized current through ion channels of cardiac cells needed to maintain the membrane potential at -35 mV. The input variable x is the logarithm of the experiment time rescaled to $\mathcal{D}_1 = [0, 1]$. The calibration inputs $\boldsymbol{\theta} \in \mathcal{D}_2 = [0, 10]^3$ control a mathematical model $\eta_{\text{cell}}(\cdot, \boldsymbol{\theta})$ of the phenomenon proposed by Clancy and Rudy (1999). Here it is considered to be an expensive black box with $N = 300$ runs available, whereas the number of observations is $n = 19$. The runs are located in a space filling manner in $\mathcal{D}_1 \times \mathcal{D}_2$ (Latin hypercube sampling optimized with maximin distance criterion).

We compare DGP-CAL with additive and general discrepancy against four competitors. The method "L_2" is a simple minimization over $\boldsymbol{\theta}$ of the L_2 residual error $||\boldsymbol{y} - \hat{\eta}_{\text{cell}}(X, \boldsymbol{\theta})||$, where $\hat{\eta}_{\text{cell}}$ is a surrogate of η_{cell} given X^*, T and Z. Its minimization takes 30 seconds and the residual error is 1.31. This method is good for predicting observations from the real process, but it provides no quantification of uncertainty.

In Table 1, we report the mean squared error (MSE), $\mathbb{E}_{q(\boldsymbol{\theta})}(||\boldsymbol{y} - \eta_{\text{cell}}(X, \boldsymbol{\theta})||^2)$, where q represents the estimated posterior density of $\boldsymbol{\theta}$. All tuning parameters of the codes

Method	Time (s)	MSE
Projected	3792	2.52
dgp-cal additive	79	1.83
dgp-cal general	93	1.55
koh	3245	5.10
Robust	361	1.99

Table 1: Comparison of errors on the Cell Biology problem.

are left to default values, and all methods are run on the same machine to ensure some fairness in reporting running times (laptop with 4×2.50 GHz cores). We see that the mse values obtained by the methods Projected, dgp-cal and Robust are significantly lower than the mse of koh. The proposed dgp-cal is the fastest. The version of dgp-cal with general discrepancy performs slightly better, thanks to the relaxation of the hypothesis of purely additive discrepancy.

The posterior distributions over $\boldsymbol{\theta}$ obtained by the calibration methods we tested are reported in Figure 4. All methods yield a distribution concentrated around the L_2 minimizer (the red dot). However, the distributions are clearly not similar to each other (except for Robust and dgp-cal). This could be explained by differences in the model formulations. Although we ensured that the covariance and mean functions are the same for all competing methods (Matérn with smoothness 5/2 and constant mean), there are several differences that cannot be matched. For instance, Robust has an additional step in the hierarchy of priors concerning the L_2 norm of the discrepancy. Moreover, the definition of the calibration parameters $\boldsymbol{\theta}$ itself differs among methods. In Projected, $\boldsymbol{\theta}$ is a minimizer of a given stochastic process, while other methods follow the koh definition. Also, Robust performs a fully Bayesian inference including hyperparameters, while in the other methods, including ours, they are optimized. It would be straightforward to allow for a Bayesian treatment of the hyperparameters in dgp-cal, but we leave this for future work.

To visualize the results of the calibration process, in Figure 5 we overlay the observations from the real process with the responses of the computer model $\eta_{\text{cell}}(\cdot, \boldsymbol{\theta})$ when $\boldsymbol{\theta}$ is sampled from its posterior distribution. All the probabilistic methods present a good fit while allowing for quantification of uncertainty in the predictions, with larger uncertainty for models that account for the uncertainty in the hyperparameters.

We see how the computer model output η is warped by γ in dgp-cal with general discrepancy (3.1). In Figure 6 we display the expected derivative of the warping with respect to the computer model output, i.e., $\mathbb{E}\left[\frac{\partial\gamma(\cdot,x)}{\partial\eta}\right]$, for three values of x. As the estimated values oscillate around one for every $x \in \mathcal{D}_1$, this model confirms that an additive discrepancy is a sensible assumption. When the estimated $\gamma(\cdot, x)$ is exactly the identity, the general discrepancy boils down to an additive one. This figure also shows how the model with general discrepancy can adapt to data sets with space-dependent behavior. Indeed in this test case the values of η have a very different distribution according to x. If x is around 0.2, the distribution of the computer runs is very asymmetric, with a

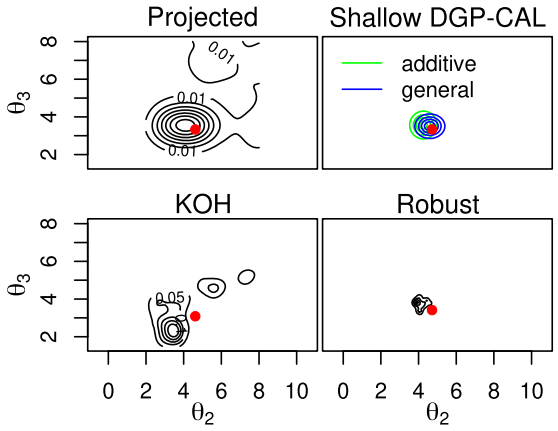

Figure 4: Posterior distributions of $\boldsymbol{\theta}$ from the calibration methods (integrated over θ_1 for visualization).

heavy tail for high values, and very light tail for low values (see the grey dots in Figure 5). On the other hand for higher values of x, say higher than 0.6, the distribution of z is more symmetric, and looks closer to a Gaussian distribution. This corresponds to the warping observed in Figure 6, where η gets its output warped and concentrated asymmetrically toward lower values for $x = 0.2$, while for $x = 0.6$ or 1, its Gaussian output is almost left untouched.

4.3 Model Calibration for Complex Response

We deal now with a case-study with locally nonsmooth/nonstationary response of the computer model, for which a stationary GP is generally inadequate. The computer model is a simulator of the effects of underground nuclear tests on radionuclide diffusion into aquifers at the Yucca Flats in the United States (Fenelon, 2005). We take the same data set as generated by a script in the supplementary material of Pratola and Higdon (2016), which is available online, with $d_1 = 2$, $d_2 = 6$, $n = 10$ and $N = 17600$.

In Pratola and Higdon (2016), the size of the dataset as well as nonstationary modeling is handled with a sum-of-trees regression. We carry out calibration using DGP-CAL with a two-layer DGP emulator for the computer model to showcase the ability of a more complex emulator to capture the nonstationarity that characterizes this problem. We therefore compare DGP-CAL with a shallow GP emulator. Details about the initialization can be found in Table 2. Furthermore, we compare against the modularized method with Local Approximate GPs (LAGP) of Gramacy et al. (2015).

In Figure 7, we display the posterior over the function $f(\cdot, \boldsymbol{\theta})$ modeling the real observations. We observe that only the deep variational calibration and the sum-of-trees approach manage to reproduce the nonstationary nature of the data set by capturing the spike characterizing one observation.

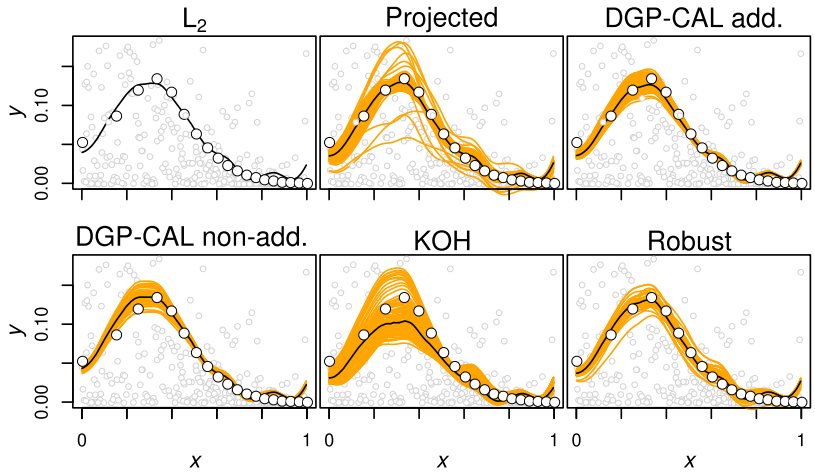

Figure 5: Samples of $\eta_{\text{cell}}(\cdot, \boldsymbol{\theta})$, with $\boldsymbol{\theta}$ drawn from its posterior. The grey dots represent the computer runs and the white dots the real observations.

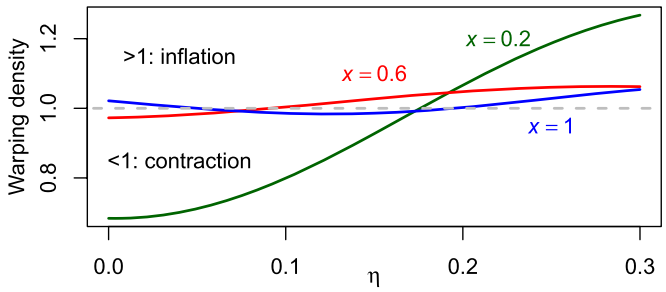

Figure 6: Derivative $\frac{\partial \gamma(\cdot, x)}{\partial \eta}$ for three values of x.

4.4 Data Set Size Scalability

We now showcase the scalability of DGP-CAL to a large calibration problem in 8 dimensions with one million computer runs, and 100,000 real observations. We use the borehole function $\eta_{\text{bh}}(\boldsymbol{x}, \boldsymbol{t})$, which is a widely used function in the literature of computer experiments (see, e.g., Gramacy et al. 2015). For all $\boldsymbol{x} \in [0,1]^5$ and $\boldsymbol{t} \in [0,1]^3$, we have

$$\eta_{\text{bh}}(\boldsymbol{x}, \boldsymbol{t}) = \frac{2\pi T_u (H_u - H_l)}{\ln(r/r_w)\left(1 + \frac{2LT_u}{\ln(r/r_w) r_w^2 K_w} + \frac{T_u}{T_l}\right)}, \qquad \delta_{\text{bh}}(\boldsymbol{x}) = \frac{2(10x_1^2 + 4x_2^2)}{50x_1x_2 + 10}, \tag{4.1}$$

with $T_u = x_1(115600 - 63070) + 63070$, $H_u = x_2(1110 - 990) + 990$, $H_l = x_3(820 - 700) + 700$, $L = x_4(1680 - 1120) + 1120$, $K_w = x_5(12045 - 9855) + 9855$, $r_w = t_1(0.15 - 0.05) + 0.05$, $r = t_2(50000 - 100) + 100$, $T_l = t_3(116 - 63.1) + 63.1$.

PARAM.	SHALLOW	DEEP
$\mathbb{E}_{q(\boldsymbol{\theta})}(\boldsymbol{\theta})$	$\frac{1}{2}[1,1,1]^\top$	$\frac{1}{2}[1,1,1]^\top$
$\mathrm{var}_{q(\boldsymbol{\theta})}(\boldsymbol{\theta})$	$\frac{1}{4}[1,1,1]^\top$	$\frac{1}{4}[1,1,1]^\top$
σ_y	10^{-2}	10^{-2}
σ_z	10^{-3}	10^{-3}
A_η, A_δ	$20I$	$\emptyset$, $20I$
σ_δ	$\frac{1}{10}$	$\frac{1}{10}$
σ_η	1	$\emptyset$
$A_{g^{(1)}}$, $A_{g^{(2)}}$	$\emptyset$	$2I_{d_1+d_2}$
$\sigma_{g^{(1)}}$, $\sigma_{g^{(2)}}$	$\emptyset$	1

Table 2: Radionuclide Model: Initial Values for the DGP-CAL models (in Deep setting, $\eta(\cdot) = g^{(2)}(\boldsymbol{g}^{(1)}(\cdot))$).

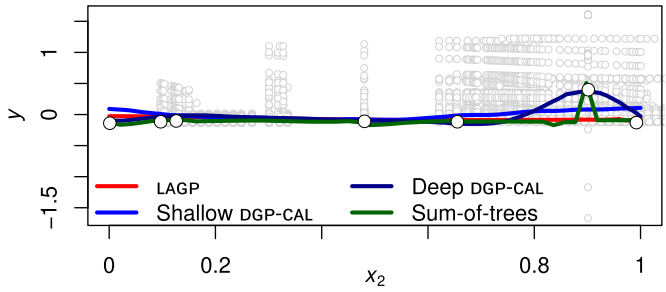

Figure 7: Mean of the posterior over the function $f(\cdot, \boldsymbol{\theta})$.

We are interested in retrieving a randomly chosen true value $\boldsymbol{\theta} = [0.089, 0.308, 0.372]^\top$. The locations X, X^* and T are generated with Latin hypercube sampling. To generate $\boldsymbol{y}$, a white Gaussian noise $\boldsymbol{\varepsilon}$ of standard deviation $\sigma_{\text{bh}} = 5 \times 10^{-3}$ is added: $y_i = \eta(\boldsymbol{x}_i, \boldsymbol{\theta}) + \delta(\boldsymbol{x}_i) + \varepsilon_i$.

We build a shallow DGP-CAL model with additive discrepancy as described by (2.1), (3.2), and (3.3). Indeed, as the response surface of the borehole functions is smooth, we consider "shallow" GPs for $\eta(\boldsymbol{x}, \boldsymbol{\theta})$ and $\delta(\boldsymbol{x})$, with Gaussian covariance approximated by 100 random features.

Concerning the sum-of-trees calibration, a sensible budget would be to set 2000 posterior samples plus 10000 for burn-in, with 1000 tree cut-points. However, this corresponds to one month of computation on our computers, so we divided the sampling budget by 5, and set 100 cut-points, keeping all other default parameters untouched.

We did not compare with the modularized calibration using LAGP, as the current implementation in `R` does not support large amount of real observations. This does not question the relevance of the method, which could be modified by using a scalable GP for the discrepancy.

Calibration	Time (h)	MSE
None (unif. samples on $\mathcal{D}_2$)	0	0.32
dgp-cal	1.4	0.03
Sum-of-trees	132.4	0.14

Table 3: Results of calibration on a large data set.

We evaluate the performance by comparing the posterior over $\boldsymbol{\theta}$ with the truth (Figure 8) and by evaluating the mse error between the computer model and observation from the real process $\mathbb{E}_{q(\boldsymbol{\theta})}(||\boldsymbol{y} - \eta_{\text{bh}}(X, \boldsymbol{\theta})||^2)$ (Table 3). dgp-cal provides the best performance both in retrieving $\boldsymbol{\theta}$ and mse, and it is the fastest by far.

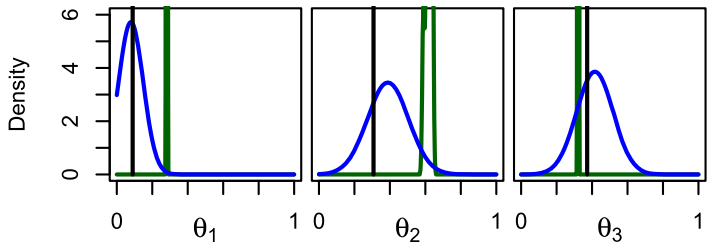

Figure 8: Posterior distribution of $\boldsymbol{\theta}$ on a large data set (black: truth, blue: dgp-cal, green: sum-of-trees).

5 Discussion

The koh model and inference in Kennedy and O'Hagan (2001) offers a classical framework to tackle calibration problems where quantification of uncertainty is of primary interest. In this paper we proposed dgp-cal, which offers a number of improvements over the koh calibration. From the modeling perspective, we cast the koh calibration model as a special case of a more general dgp model, where the latent process modeling the real observations is a warped version of the emulator of the computer model; we showed that this general calibration model retains the possibility to reason about uncertainty in the discrepancy between the computer model and the real process.

Furthermore, the proposed approximation of gps and dgps with random features and approximate inference through variational techniques give dgp-cal a number of advantages, such as simplicity of implementation in development environments featuring automatic differentiation and the possibility to exploit GPU-type hardware. The experiments showed that the approximations introduced to recover tractability do not affect the ability of dgp-cal to effectively calibrate parameters of complex computer models, while enjoying scalability to large number of observations and/or computer runs, demonstrating that dgp-cal is a powerful alternative to the state-of-the-art. We are currently investigating the application of dgp-cal to other large-scale calibration problems in

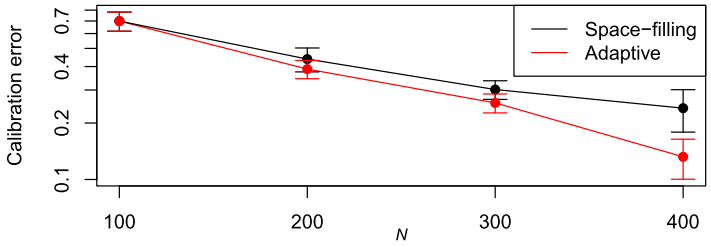

Figure 9: Calibration error as a function of the number of runs.

environmental sciences, where the KOH model and related calibration methodologies are usually not the preferred choice due to limited scalability.

We conclude the paper by discussing two important aspects of this work, namely adaptive experimental design and identifiability.

Adaptive Experimental Design Building up the design sequentially may allows smaller training sets, limiting the evaluation of the simulator code (see e.g. Sauer et al. 2021). It is possible to extend DGP-CAL to handle cases where the uncertainty in the model can be used to guide the incremental design of the experiment. For simplicity, we assume $\eta(\boldsymbol{x}, \boldsymbol{t})$ in the KOH model to be modeled as a GP. The goal is to improve calibration by sequentially optimizing the locations of the inputs of the computer model (X^* and T). More precisely, instead of determining the N locations from the outset, N_{add} points are added at each of n_{it} iterations to an initial design of $N_0 < N$ experiments. At each iteration, the model is inferred on the data integrating the new batch of N_{new} inputs and outputs. The batch of points $[X^*_{\text{ca}}, T_{\text{ca}}]$ with corresponding labels $\boldsymbol{z}_{\text{ca}}$ added to the design is determined as a solution to the optimization of a criterion. We can specify a criterion such that the evaluation of η at the best candidate points maximally reduces the variance $\boldsymbol{\theta}$. Many sampling criteria can be imagined and tested; for example, one can choose the sum of the partial derivatives of the lower bound with respect to the (logarithm of the) variance of the components of $\boldsymbol{\theta}$:

$$[X^*_{\text{add}}, T_{\text{add}}] = \underset{X^*_{\text{ca}} \in \mathcal{D}_1, T_{\text{ca}} \in \mathcal{D}_2}{\operatorname{argmax}} - \sum_i^{d_2} \frac{\partial \mathcal{L}_{\text{ca}}(X^*_{\text{ca}}, T_{\text{ca}})}{\partial \log \xi_i}, \tag{5.1}$$

where ξ_i is the parameter of the of variational distribution controlling the variance of θ_i. The function $\mathcal{L}_{\text{ca}}$ returns the lower bound in (2.14) after updating the variational distribution with candidate input points.

In the case of GPs with random features, the update of the variational distribution can be computed analytically (see, e.g., Section 2.3.3 of Bishop (2006) for a derivation), obtaining updated mean and covariance as

$$\begin{aligned} \boldsymbol{\mu}_{\text{ca}} &= \Sigma_{\text{ca}} \left(\Sigma_0^{-1} \boldsymbol{\mu}_0 + \sigma_z^2 \Phi_{\eta,\text{ca}}^\top \boldsymbol{z}_{\text{ca}} \right), \\ \Sigma_{\text{ca}} &= \sigma_z^2 \left(\Phi_{\eta,\text{ca}}^\top \Phi_{\eta,\text{ca}} + \sigma_z^2 \Sigma_0^{-1} \right)^{-1}, \end{aligned} \tag{5.2}$$

with $\Phi_{\eta,\text{ca}} \in \mathbb{R}^{N_\text{ca} \times N_\text{RF}}$ the row by row evaluation of $\boldsymbol{\phi}_\eta(\boldsymbol{x}, \boldsymbol{t})^\top$ on X^*_ca and T_ca, $\boldsymbol{\mu}_0$ and Σ_0 are the mean and the covariance of the variational distribution before selecting the candidate points, and $\sigma_z^2 > 0$ corresponds to the noise term of the likelihood function. The criterion forecasts the effect of evaluating η at $[X^*_\text{ca}, T_\text{ca}]$ using the predictive mean $\boldsymbol{z}_\text{ca} = \mathbb{E}(\eta(X^*_\text{ca}, T_\text{ca})|U, \boldsymbol{\psi}, \Omega)$. Therefore, the adaptive sampling chooses new input locations which maximally reduce the variance of the approximate posterior over $\boldsymbol{\theta}$.

As an illustration of this approach, we perform adaptive sampling on the borehole function in (4.1), starting from $N_0 = 100$ points and adding batches of $N_\text{add} = 20$ until we reach $N = 400$. The initial design is obtained by a latin hypercube sampling with optimized maximin distance. The sampling criterion in (5.1) is optimized with gradient descent using automatic differentiation with respect to X^*_ca and T_ca. We compare against latin hypercube sampling with optimization of the maximin distance, which is a classic space-filling design, for $N = 100$, 200, 300 and 400. The experiment is reproduced five times with different initial and space filling designs, and we report calibration errors $||\mathbb{E}_{q(\theta)}(\boldsymbol{\theta}) - \boldsymbol{\theta}||$ with standard deviations in Figure 9. The designs produced by the adaptive method lead to consistently lower calibration error due to the optimized locations of the computer runs.

Identifiability The issue of identifiability that we discussed for the KOH model affects our formulation too. In particular, for the generalized formulation where the discrepancy is not modeled through an additive component but through a composition with an unknown function $\gamma(\eta(\boldsymbol{x}, \boldsymbol{t}), \boldsymbol{x})$, our approach yields a posterior over $\boldsymbol{\theta}$ and $\gamma(\eta(\boldsymbol{x}, \boldsymbol{t}), \boldsymbol{x})$ which concentrates on the manifold

$$\mathcal{M} = \{(\boldsymbol{t}, \gamma(\eta(\boldsymbol{x}, \boldsymbol{t}), \boldsymbol{x})) \,|\, \gamma(\eta(\boldsymbol{x}, \boldsymbol{t}), \boldsymbol{x}) = f(\boldsymbol{x}, \boldsymbol{t}) \quad \text{with} \quad \boldsymbol{x} \in X \cup X^*\},$$

as the number of observations increases. Ignoring the discrepancy by removing the composition of the warping function with $\eta(\boldsymbol{x}, \boldsymbol{\theta})$ would not constitute a problem from the implementation perspective, and our approach would result in a fast and flexible calibration model where $\boldsymbol{\theta}$ is identifiable. Similarly to the KOH model, however, ignoring the discrepancy results in a model that could potentially yield a biased estimation of the parameters $\boldsymbol{\theta}$, depending on how accurate $\eta(\boldsymbol{x}, \boldsymbol{t})$ is in modeling the physical process.

The composition with a discrepancy function is therefore needed to avoid bias in the estimate of $\boldsymbol{\theta}$. As a result, the importance of priors over $\boldsymbol{\theta}$ and model discrepancy to mitigate the issue of lack of identifiability is as important as in the KOH model. This can dictate the choice of GP versus DGP priors. For instance, in the radionuclide diffusion test case (Section 4.3), a preview of the hectic observations hints in favor of a DGP model. On the contrary, in Section 4.4 knowing that the borehole function is smooth, the computer model and the discrepancy were given GP priors. Beyond these general considerations, there exist works allowing to impose constraints on DGPs priors, such as positivity, monotonicity, convexity, or being solutions to differential equations. In Lorenzi and Filippone (2018), this is done within the same framework proposed here involving random feature approximations and variational inference, so it would be rather straightforward to incorporate these in our model and inference.

Other ideas proposed in the literature on how to deal with the lack of identifiability could easily be adapted to our framework. For example, when multiple responses are available and they are mutually dependent on the same set of calibration parameters $\boldsymbol{\theta}$, this improves identifiability (Arendt et al., 2012), and accommodating for multiple responses in our framework is straightforward. Furthermore, it would be possible to apply similar ideas to the ones proposed in Tuo and Wu (2016); Wong et al. (2017) where GP priors are replaced by DGPs.

Another exciting line of research, which we could benefit from in order to investigate the identifiability properties of DGP-CAL, follows recent results on so-called *disentanglement* (Locatello et al., 2019). This literature studies the properties of deep models in the context of density estimation, through the use of variational autoencoders (Kingma and Welling, 2014). Autoencoders are models composed of two elements: an encoder and a decoder. The encoder maps a set of observations into a low-dimensional latent representation, whereas the decoder maps the latent variables into the observations. In variational autoencoders, one is interested in obtaining a distribution over the latent variables by means of a variational formulation. Recently, a lot of interest has been devoted to disentangled representations, which are those where the latent variables are independently controlling different generative factors, and their number is sufficient to capture the diversity observed in the observations. A popular example is the one where observations are images of a ball of different colors and sizes, and with varying color of the background; the three generating factors are color and size of the ball, and color of the background, and one hopes to be able to infer this without any prior knowledge and simply by analyzing the given images. Locatello et al. (2019) made a theoretical breakthrough showing that this is impossible unless some extra information on these generating factors is available. Following this negative result, Khemakhem et al. (2020) provided theoretical guarantees on how to recover identifiability of these latent generating factors by means of a factorized prior distribution over the latent variables that is conditioned on some additional observed variables. While the setup is different to calibration, there are many similarities with our model in that they both use deep models and variational inference. Our calibration model can be seen as the decoder of such variational autoencoders, and $\boldsymbol{\theta}$ in our model can be seen as the latent generative factors without encoder. We believe that this is an interesting direction to establish novel results and insights on the identifiability of the proposed calibration model.

References

Álvarez, M. A. and Lawrence, N. D. (2011). "Computationally efficient convolved multiple output Gaussian processes." *Journal of Machine Learning Research*, 12(41): 1459–1500. MR2813145. 1306

Arendt, P. D., Apley, D. W., Chen, W., Lamb, D., and Gorsich, D. (2012). "Improving identifiability in model calibration using multiple responses." *Journal of Mechanical Design*, 134(10): 100909. 1305, 1324

Arhonditsis, G. B., Qian, S. S., Stow, C. A., Lamon, E. C., and Reckhow, K. H. (2007).

"Eutrophication risk assessment using Bayesian calibration of process-based models: Application to a mesotrophic lake." *Ecological Modelling*, 208(2): 215–229. 1301

Bishop, C. M. (2006). *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Berlin, Heidelberg: Springer-Verlag. MR2247587. doi: https://doi.org/10.1007/978-0-387-45528-0. 1323

Blei, D. M., Kucukelbir, A., and McAuliffe, J. D. (2017). "Variational inference: A review for statisticians." *Journal of the American statistical Association*, 112(518): 859–877. MR3671776. doi: https://doi.org/10.1080/01621459.2017.1285773. 1309

Brynjarsdóttir, J. and O'Hagan, A. (2014). "Learning about physical parameters: The importance of model discrepancy." *Inverse Problems*, 30(11): 114007. MR3274591. doi: https://doi.org/10.1088/0266-5611/30/11/114007. 1301, 1302, 1305

Bui, T. D., Hernández-Lobato, D., Hernández-Lobato, J. M., Li, Y., and Turner, R. E. (2016). "Deep Gaussian Processes for Regression using Approximate Expectation Propagation." In Balcan, M.-F. and Weinberger, K. Q. (eds.), *Proceedings of the 33nd International Conference on Machine Learning, ICML 2016, New York City, NY, USA, June 19–24, 2016*, volume 48, 1472–1481. JMLR.org. 1308

Cho, Y. and Saul, L. (2009). "Kernel Methods for Deep Learning." In Bengio, Y., Schuurmans, D., Lafferty, J., Williams, C., and Culotta, A. (eds.), *Advances in Neural Information Processing Systems*, volume 22, 342–350. Curran Associates, Inc. 1307

Clancy, C. E. and Rudy, Y. (1999). "Linking a genetic defect to its cellular phenotype in a cardiac arrhythmia." *Nature*, 400(6744): 566. 1317

Cutajar, K., Bonilla, E. V., Michiardi, P., and Filippone, M. (2017). "Random Feature Expansions for Deep Gaussian Processes." In Precup, D. and Teh, Y. W. (eds.), *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, 884–893. International Convention Centre, Sydney, Australia: PMLR. 1302, 1308, 1310, 1311

Damianou, A. and Lawrence, N. D. (2013). "Deep Gaussian Processes." In Carvalho, C. M. and Ravikumar, P. (eds.), *Proceedings of the Sixteenth International Conference on Artificial Intelligence and Statistics*, volume 31 of *Proceedings of Machine Learning Research*, 207–215. Scottsdale, Arizona, USA: PMLR. 1302, 1308

Duvenaud, D. K., Rippel, O., Adams, R. P., and Ghahramani, Z. (2014). "Avoiding pathologies in very deep networks." In *Proceedings of the Seventeenth International Conference on Artificial Intelligence and Statistics, AISTATS 2014, Reykjavik, Iceland, April 22–25, 2014*, volume 33 of *JMLR Workshop and Conference Proceedings*, 202–210. JMLR.org. 1308

Fenelon, J. M. (2005). *Analysis of Ground-Water Levels and Associated Trends in Yucca Flat, Nevada Test Site, Nye County, Nevada, 1951–2003*. US Department ot the Interior, US Geological Survey. 1319

Filippone, M. and Girolami, M. (2014). "Pseudo-marginal Bayesian inference for Gaussian processes." *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(11): 2214–2226. MR3577381. doi: https://doi.org/10.1214/16-BA1038. 1302

Filippone, M., Zhong, M., and Girolami, M. (2013). "A comparative evaluation of stochastic-based inference methods for Gaussian process models." *Machine Learning*, 93(1): 93–114. MR3093120. doi: https://doi.org/10.1007/s10994-013-5388-x. 1302

Gal, Y. and Ghahramani, Z. (2016). "Dropout as a Bayesian Approximation: Representing Model Uncertainty in Deep Learning." In Balcan, M. F. and Weinberger, K. Q. (eds.), *Proceedings of The 33rd International Conference on Machine Learning*, volume 48 of *Proceedings of Machine Learning Research*, 1050–1059. New York, New York, USA: PMLR. 1308

Gramacy, R. B., Bingham, D., Holloway, J. P., Grosskopf, M. J., Kuranz, C. C., Rutter, E., Trantham, M., and Drake, R. P. (2015). "Calibrating a large computer experiment simulating radiative shock hydrodynamics." *Annals of Applied Statistics 2015, Vol. 9, No. 3, 1141-1168*. MR3418718. doi: https://doi.org/10.1214/15-AOAS850. 1316, 1320, 1321

Graves, A. (2011). "Practical Variational Inference for Neural Networks." In Shawe-Taylor, J., Zemel, R. S., Bartlett, P. L., Pereira, F., and Weinberger, K. Q. (eds.), *Advances in Neural Information Processing Systems 24*, 2348–2356. Curran Associates, Inc. 1309, 1310, 1311

Gu, M. (2018). *RobustCalibration: Robust Calibration of Imperfect Mathematical Models*. R package version 0.5.1. 1316

Gu, M. and Wang, L. (2018). "Scaled Gaussian stochastic process for computer model calibration and prediction." *SIAM/ASA Journal on Uncertainty Quantification*, 6(4): 1555–1583. MR3875809. doi: https://doi.org/10.1137/17M1159890. 1316

Henderson, D. A., Boys, R. J., Krishnan, K. J., Lawless, C., and Wilkinson, D. J. (2009). "Bayesian emulation and calibration of a stochastic computer model of mitochondrial DNA deletions in substantia nigra neurons." *Journal of the American Statistical Association*, 104(485): 76–87. MR2663034. doi: https://doi.org/10.1198/jasa.2009.0005. 1301

Hensman, J. and Lawrence, N. D. (2014). "Nested Variational Compression in Deep Gaussian Processes." *arXiv preprint arXiv:1412.1370*. 1308

Jordan, M. I., Ghahramani, Z., Jaakkola, T. S., and Saul, L. K. (1999). "An introduction to variational methods for graphical models." *Machine Learning*, 37(2): 183–233. 1309

Kennedy, M. C. and O'Hagan, A. (2001). "Bayesian calibration of computer models." *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 63(3): 425–464. MR1858398. doi: https://doi.org/10.1111/1467-9868.00294. 1301, 1302, 1304, 1305, 1311, 1322

Khemakhem, I., Kingma, D., Monti, R., and Hyvarinen, A. (2020). "Variational Autoencoders and Nonlinear ICA: A Unifying Framework." In Chiappa, S. and Calandra, R. (eds.), *Proceedings of the Twenty Third International Conference on Artificial In-*

telligence and Statistics, volume 108 of *Proceedings of Machine Learning Research*, 2207–2217. PMLR. 1325

Kingma, D. P., Salimans, T., and Welling, M. (2015). "Variational Dropout and the Local Reparameterization Trick." In Cortes, C., Lawrence, N. D., Lee, D. D., Sugiyama, M., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 28*, volume 28, 2575–2583. Curran Associates, Inc. 1310

Kingma, D. P. and Welling, M. (2014). "Auto-Encoding Variational Bayes." In *Proceedings of the Second International Conference on Learning Representations (ICLR 2014)*. 1310, 1325

Larssen, T., Huseby, R. B., Cosby, B. J., Høst, G., Høgåsen, T., and Aldrin, M. (2006). "Forecasting acidification effects using a Bayesian calibration and uncertainty propagation approach." *Environmental Science & Technology*, 40(24): 7841–7847. PMID: 17256536. 1301

Lázaro-Gredilla, M., Quinonero-Candela, J., Rasmussen, C. E., and Figueiras-Vidal, A. R. (2010). "Sparse spectrum Gaussian process regression." *Journal of Machine Learning Research*, 11: 1865–1881. MR2660655. 1306

Liu, Q. and Wang, D. (2016). "Stein Variational Gradient Descent: A General Purpose Bayesian Inference Algorithm." In Lee, D., Sugiyama, M., Luxburg, U., Guyon, I., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc. 1309

Locatello, F., Bauer, S., Lucic, M., Raetsch, G., Gelly, S., Schölkopf, B., and Bachem, O. (2019). "Challenging Common Assumptions in the Unsupervised Learning of Disentangled Representations." In Chaudhuri, K. and Salakhutdinov, R. (eds.), *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, 4114–4124. PMLR. 1325

Lorenzi, M. and Filippone, M. (2018). "Constraining the Dynamics of Deep Probabilistic Models." In Dy, J. and Krause, A. (eds.), *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, 3227–3236. PMLR. 1324

Matthews, A., Hron, J., Rowland, M., Turner, R. E., and Ghahramani, Z. (2018). "Gaussian Process Behaviour in Wide Deep Neural Networks." In *International Conference on Learning Representations*. 1308

Neal, R. M. (1993). "Probabilistic Inference using Markov chain Monte Carlo Methods." Technical Report CRG-TR-93-1, Dept. of Computer Science, University of Toronto. 1302

Neal, R. M. (1996). *Bayesian Learning for Neural Networks (Lecture Notes in Statistics)*. Springer, 1 edition. 1306, 1308

Paciorek, C. J. and Schervish, M. J. (2003). "Nonstationary Covariance Functions for Gaussian Process Regression." In *Proceedings of the 16th International Conference on Neural Information Processing Systems*, NIPS'03, 273–280. Cambridge, MA, USA: MIT Press. 1302

Plumlee, M. (2017). "Bayesian calibration of inexact computer models." *Journal of the American Statistical Association*, 112(519): 1274–1285. MR3735376. doi: https://doi.org/10.1080/01621459.2016.1211016. 1305, 1316

Pratola, M. T. and Higdon, D. M. (2016). "Bayesian additive regression tree calibration of complex high-dimensional computer models." *Technometrics*, 58(2): 166–179. MR3488296. doi: https://doi.org/10.1080/00401706.2015.1049749. 1316, 1319, 1320

Qian, P. Z. G. and Wu, C. F. J. (2008). "Bayesian hierarchical modeling for integrating low-accuracy and high-accuracy experiments." *Technometrics*, 50(2): 192–204. MR2439878. doi: https://doi.org/10.1198/004017008000000082. 1311

Rahimi, A. and Recht, B. (2008). "Random Features for Large-Scale Kernel Machines." In Platt, J. C., Koller, D., Singer, Y., and Roweis, S. T. (eds.), *Advances in Neural Information Processing Systems 20*, 1177–1184. Curran Associates, Inc. 1306, 1307

Rasmussen, C. E. and Williams, C. (2006). *Gaussian Processes for Machine Learning*. MIT Press. MR2514435. 1302, 1305, 1306

Rezende, D. and Mohamed, S. (2015). "Variational Inference with Normalizing Flows." In Bach, F. and Blei, D. (eds.), *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, 1530–1538. Lille, France: PMLR. 1309

Robbins, H. and Monro, S. (1951). "A stochastic approximation method." *The Annals of Mathematical Statistics*, 22: 400–407. MR0042668. doi: https://doi.org/10.1214/aoms/1177729586. 1310

Rossi, S., Michiardi, P., and Filippone, M. (2019). "Good Initializations of Variational Bayes for Deep Models." In *Proceedings of the 36th International Conference on Machine Learning – Vol. 97*, ICML'19. JMLR.org. MR3862432. 1314, 1315

Sacks, J., Welch, W. J., Mitchell, T. J., and Wynn, H. P. (1989). "Design and analysis of computer experiments." *Statistical Science*, 4(4): 409–423. MR1041765. 1301, 1302

Salimbeni, H. and Deisenroth, M. (2017). "Doubly Stochastic Variational Inference for Deep Gaussian Processes." In Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 30*, 4588–4599. Curran Associates, Inc. 1308

Salter, J. M., Williamson, D. B., Scinocca, J., and Kharin, V. (2018). "Uncertainty Quantification for Spatio-Temporal Computer Models with Calibration-Optimal Bases." *arXiv preprint arXiv:1801.08184*. MR4047301. doi: https://doi.org/10.1080/01621459.2018.1514306. 1301

Sansó, B., Forest, C. E., and Zantedeschi, D. (2008). "Inferring climate system properties using a computer model." *Bayesian Analysis*, 3(1): 1–37. MR2383247. doi: https://doi.org/10.1214/08-BA301. 1301

Sauer, A., Gramacy, R. B., and Higdon, D. (2021). "Active learning for deep Gaus-

sian process surrogates." *Technometrics*. doi: https://doi.org/10.1080/00401706.2021.2008505. 1323

Tuo, R. and Wu, C. F. J. (2016). "A theoretical framework for calibration in computer models: Parametrization, estimation and convergence properties." *SIAM/ASA Journal on Uncertainty Quantification*, 4(1): 767–795. MR3523087. doi: https://doi.org/10.1137/151005841. 1305, 1325

Williams, B., Higdon, D., Gattiker, J., Moore, L., McKay, M., and Keller-McNulty, S. (2006). "Combining experimental data and computer simulations, with an application to flyer plate experiments." *Bayesian Analysis*, 1(4): 765–792. MR2282206. doi: https://doi.org/10.1214/06-BA125. 1301

Wong, R. K., Storlie, C. B., and Lee, T. C. (2017). "A frequentist approach to computer model calibration." *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 79(2): 635–648. MR3611763. doi: https://doi.org/10.1111/rssb.12182. 1305, 1316, 1325

Xie, F. and Xu, Y. (2018). "Bayesian Projected Calibration of Computer Models." *arXiv preprint arXiv:1803.01231*. MR4353726. doi: https://doi.org/10.1080/01621459.2020.1753519. 1316

Invited Discussion

Anthony O'Hagan*

Reading this paper was truly delightful. Having played my part in starting this line of research by explicitly recognising the importance of model discrepancy in computer models, I am thrilled to see how it has moved on in the last twenty years. I will confess right away that I have not kept up with the technical and computational developments employed in this paper; I was aware of deep Gaussian processes and variational inference, but had no real appreciation of their mathematics and what they could do. It seems that they offer enormous advances in this field of computer model calibration.

Having said that, there remains the problem of identifiability. As the authors point out clearly in Section 5, we cannot hope to learn about meaningful, physical parameters without additional prior information. And while that may come in part from prior information about the parameters themselves, the enterprise has hardly achieved much if we only learn about them *a posteriori* by having good knowledge of them *a priori*. The crucial step is to incorporate prior knowledge of the model discrepancy, and it seems to me that while the developments in this paper resolve the principal computational challenges in the original Kenndy-O'Hagan formulation, they may do so at the expense of making it harder to see and construct the prior information. I strongly encourage the authors and other researchers in the field to demonstrate, in real applications, how genuine prior knowledge can be incorporated into a deep Gaussian process model for the discrepancy.

*University of Sheffield, a.ohagan@sheffield.ac.uk

Invited Discussion

Annie Sauer* and Robert B. Gramacy†

Let us begin by congratulating Marmin and Filippone (2022) for a timely and important paper. Their methodological advancements touch on several themes that are important to our own research – computer surrogate modeling at scale and at high fidelity, uncertainty quantification, and active learning for sequential design – and about which we shall offer some discussion below. Broadly, as advances in predictive modeling, primarily in machine learning (ML), work their way back to their statistical origins (e.g., deep Gaussian processes [DGPs] originated in the geo-spatial statistical literature (Sampson and Guttorp, 1992; Schmidt and O'Hagan, 2003)), one may naturally worry at what risk, or what has been lost? Marmin and Filippone do a fantastic job, in our opinion, of borrowing from ML for computer model calibration. In so doing, they demonstrate that it is possible to enjoy advances in modern analytics in the context of an endeavor demanding numerically cumbersome posterior inference and full uncertainty quantification (UQ): something dear to statisticians but often ignored in ML. Here we offer some thoughts on how (we hope) things can continue to move, productively, in that direction while at the same time shamelessly promoting some of our own work.

Diverging goals of calibration

In their seminal paper on Bayesian computer model calibration, Kennedy and O'Hagan (2001, KOH, hereafter) acknowledged that their framework would excel at synthesis (of computer model runs and field data), but struggle at identification of the calibration parameter (the inverse problem): $\boldsymbol{\theta}$. Without getting too technical, this is because the modeling apparatus – coupling two GPs, one for the computer model and one for the bias – together with $\boldsymbol{\theta}$ is "too flexible". Since then, and especially recently, the literature has diverged. Some, beginning with Higdon et al. (2004) and Liu et al. (2009), and more recently with Brynjarsdóttir and O'Hagan (2014); Plumlee (2017); Gu and Wang (2018); Gu (2019); Tuo and Wu (2015, 2016); Wong et al. (2017); Plumlee (2019) focus on identification of $\boldsymbol{\theta}$ in the inversion problem, mostly by reigning in that flexibility. Others, like this discussion paper, but also including Konomi et al. (2017); Karagiannis et al. (2019); Cheng et al. (2021) and some of our own work (Gramacy et al., 2015; Huang et al., 2020; Huang and Gramacy, 2021) focus instead on modeling computer simulations at scale, enhancements in fidelity of the surrogate, and on synthesis of data sources (simulation and physical observation) for prediction and UQ. In other words, emphasizing more flexibility because an ordinary GP is insufficient when physical processes are nonstationary or exhibit discontinuity, for example.

This divide, between identification and modeling fidelity, reminds us of a similar tussle in econometrics and clinical trials: that between causal inference/treatment effects, and that of modeling for the purpose of reproducing behavior, and for accurately

*Department of Statistics, Virginia Tech, anniees@vt.edu
†Department of Statistics, Virginia Tech

predicting the outcome (with UQ) for an experiment in novel circumstances. The two would seem to be fundamentally at odds with one another. Identification requires more things be "bolted down", anathema to modern ML, whereas predictive accuracy and UQ require a delicate balance between fidelity and regularization – allowing the model to exhibit and learn complex dynamics when the data support it.

Our opinion is that in the context of computer surrogate modeling, you have to have the first (fidelity) *before* you have the second (identification). We think KOH would agree. If you can't handle the scale of the simulation experiment (tens of thousands to millions of runs), and (in spite of such large data) you can't capture the nuances of the (e.g., physical) dynamics in play, then it doesn't matter if the theory says you can identify $\boldsymbol{\theta}$ or not. Ideally you would have both, but at this time – and perhaps indefinitely if econometrics is any guide – that would seem to be illusive. We therefore congratulate Marmin and Filippone for making important advancements in modeling fidelity by deploying DGPs where ordinary (shallow) GPs are not enough. We wish they, and other authors before them, didn't have to apologize for not accommodating identification (awkward paragraphs in Sections 2.1 and 5), and we hope that their paper will provide inspiration to continue with advances along modeling fidelity lines.

Chapter 8.1 of *Surrogates* (Gramacy, 2020) discusses a ball-drop experiment where the computer model incorporates acceleration due to gravity, but omits atmospheric resistance among other factors. By simplifying the KOH apparatus you might believe that you can identify $\boldsymbol{\theta}$, the acceleration due to gravity. But in fact you can't, not with any amount of data, not with any of the theoretically sound papers cited above. This is a common situation. You can only "invert $\boldsymbol{\theta}$" modulo the fictional scenario that your computer model includes all dynamics exhibited in field. In practice your model will idealize some of those dynamics, possibly over-simplifying important ones (because all models are wrong). However, it *is* possible to accurately predict the ball drop time, accounting for both forces and anything else omitted from the computer model, if you use the full KOH apparatus because it does a great job of synthesizing information sources. The only limitation here is whether or not your class of models for the surrogate, and the discrepancy term, is rich enough to capture dynamics which the data (perhaps a large amount of it) reveal. This is where ML tools are most valuable, and where Marmin and Filippone make an important advance on the state-of-the-art.

Surrogate modeling fidelity and UQ

But at the same time, we wonder if the variational inference (VI) and random feature expansion (RFE) approach by Marmin and Filippone is really doing the best job at prediction and UQ. The combination of these boils down to two choices: an inferential scheme to handle the intractible DGP posterior (VI) and an approximation apparatus to circumvent the cubic costs of GP inference (RFE), which are exacerbated in a DGP setting. VI, although popular in the DGP literature, offers an approximate solution by settling for a "best guess" distribution from a known target family. By nature, this approximation oversimplifies the complexities of the DGP posterior. Bayesian Markov chain Monte Carlo (MCMC) sampling approaches, while computationally heavier, offer full posterior integration and thorough UQ. See Sauer et al. (2022b) for an ellipti-

cal slice sampling (ESS) approach and Havasi et al. (2018) for a Hamiltonian Monte Carlo (HMC) implementation. Until recently, the prevailing approximation apparatus for DGPs was via inducing points (Snelson and Ghahramani, 2006). Inducing points yield disappointingly blurry predictions; RFE is perhaps a promising alternative. We have found recent success with the Vecchia approximation (Vecchia, 1988) for fully Bayesian DGPs (Sauer et al., 2022a).

The pivotal question is, which of these offer the best DGP surrogate predictions/UQ at reasonable computational cost? The answer unfortunately is masked by coding considerations; existing codes combine one of each of the above tools (VI/MCMC and inducing points/RFE/Vecchia), along with varying treatment of kernel hyperparameters, etc., making it tricky to pull apart individual components. Nevertheless, a candid comparison is warranted. To this end, we offer a simulation study of current DGP surrogate implementations on two nonstationary test cases (from Surjanovic and Bingham, 2013). The DGP comparators we considered are: doubly stochastic VI + inducing points of Salimbeni and Deisenroth (2017) via `gpflux` (Dutordoir et al., 2021), HMC + inducing points of Havasi et al. (2018), ESS + Vecchia approximation of Sauer et al. (2022a) via `deepgp` (Sauer, 2022), VI + RFE of Cutajar et al. (2017), and VI + RFE of Marmin and Filippone (2022). Prediction error (root mean squared error [RMSE]) and UQ performance (continuous rank probability score [CRPS], lower is better) for each DGP surrogate are provided in Figure 1. Code to reproduce all results is available in our public repository.[1]

Both VI + RFE implementations utilize the same approximation tools yet yield drastically different results, making it clear that coding and modeling choices can be as impactful as the choice of statistical tools. While it's impossible to fully separate inferential schemes, approximation choices, and other coding considerations, several things are still clear. Sampling schemes (HMC + ESS) tend to outperform VI. In all but the smallest settings, inducing point and RFE variations appeared to saturate and were unable to improve learning from larger training data sizes. In contrast, the Vecchia approximation consistently improved as training data size increased. The combination of full posterior integration and high-fidelity Vecchia approximation (ESS + Vecchia) outperformed across the board.

DGP surrogates are growing in popularity and impact, but the existing public implementations are a bit of a Wild West. There is great potential for off-the-shelf DGP implementations that work well in robust settings. In surrogate settings where prediction accuracy and UQ are essential, full posterior integration is clearly superior to VI, albeit at the cost of some extra computation. VI may still have its place when data sizes are enormous (full posterior sampling for DGPs has only recently become accessible for data sizes in the hundred thousands, Sauer et al. (2022a)), but when data is abundant, the benefit of a DGP surrogate's nonstationarity may be less impactful.

Active learning

As a final thought we wish to touch on Marmin and Filippone's Section 5 discussion of potential for active learning (AL) in calibration contexts, i.e., of acquiring new com-

[1] https://bitbucket.org/gramacylab/deepgp-ex/

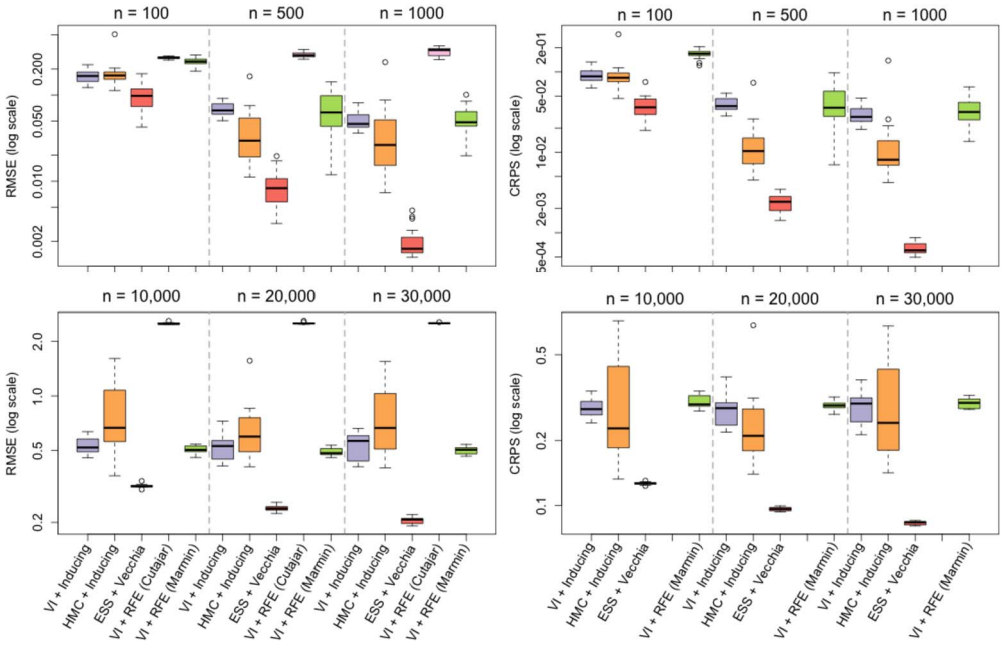

Figure 1: RMSE (left) and CRPS (right) for DGP surrogate fits to the 2d Schaffer function (20 reps) and the 6d G-function (10 reps) as training data size (n) increases. VI + RFE (Cutajar) does not provide variances, so it is omitted from the right panels.

puter model runs or field data observations to improve the fit. UQ is essential to AL enterprises, and AL is essential when training data sizes are limited. This is especially the case for DGP surrogates, as remarked by Sauer et al. (2022b). If you have limited training data, then it is important to place runs – or learn where to place runs – so that you can learn how dynamics are changing in the input space. With ordinary GPs, AL isn't much help because it invariably learns to space-fill the study region. But with DGPs the resulting designs can be highly non-uniform and, consequently, lead to superior predictive performance. We agree with Marmin and Filippone that it would be interesting to study this further in a calibration context.

References

Brynjarsdóttir, J. and O'Hagan, A. (2014). "Learning about physical parameters: The importance of model discrepancy." *Inverse Problems*, 30(11): 114007. MR3274591. doi: https://doi.org/10.1088/0266-5611/30/11/114007. 1332

Cheng, S., Konomi, B. A., Matthews, J. L., Karagiannis, G., and Kang, E. L. (2021). "Hierarchical Bayesian nearest neighbor co-kriging Gaussian process models; an application to intersatellite calibration." *Spatial Statistics*, 44: 100516. MR4270521. doi: https://doi.org/10.1016/j.spasta.2021.100516. 1332

Cutajar, K., Bonilla, E. V., Michiardi, P., and Filippone, M. (2017). "Random feature expansions for deep Gaussian processes." In *International Conference on Machine Learning*, 884–893. PMLR. 1334

Dutordoir, V., Salimbeni, H., Hambro, E., McLeod, J., Leibfried, F., Artemev, A., van der Wilk, M., Hensman, J., Deisenroth, M. P., and John, S. (2021). "GPflux: A library for deep Gaussian processes." *arXiv preprint* arXiv:2104.05674. 1334

Gramacy, R. B. (2020). *Surrogates: Gaussian Process Modeling, Design and Optimization for the Applied Sciences*. Boca Raton, Florida: Chapman Hall/CRC. http://bobby.gramacy.com/surrogates/. MR4283556. doi: https://doi.org/10.1201/9780367815493. 1333

Gramacy, R. B., Bingham, D., Holloway, J. P., Grosskopf, M. J., Kuranz, C. C., Rutter, E., Trantham, M., and Drake, R. P. (2015). "Calibrating a large computer experiment simulating radiative shock hydrodynamics." *Annals of Applied Statistics*, 9(3): 1141–1168. MR3418718. doi: https://doi.org/10.1214/15-AOAS850. 1332

Gu, M. (2019). "Jointly robust prior for Gaussian stochastic process in emulation, calibration and variable selection." *Bayesian Analysis*, 14(3): 857–885. MR3960774. doi: https://doi.org/10.1214/18-BA1133. 1332

Gu, M. and Wang, L. (2018). "Scaled Gaussian Stochastic Process for Computer Model Calibration and Prediction." *SIAM/ASA Journal on Uncertainty Quantification*, 6(4): 1555–1583. MR3875809. doi: https://doi.org/10.1137/17M1159890. 1332

Havasi, M., Hernández-Lobato, J. M., and Murillo-Fuentes, J. J. (2018). "Inference in deep Gaussian processes using stochastic gradient Hamiltonian Monte Carlo." *Advances in neural information processing systems*, 31. 1334

Higdon, D., Kennedy, M., Cavendish, J. C., Cafeo, J. A., and Ryne, R. D. (2004). "Combining field data and computer simulations for calibration and prediction." *SIAM Journal on Scientific Computing*, 26(2): 448–466. MR2116355. doi: https://doi.org/10.1137/S1064827503426693. 1332

Huang, J. and Gramacy, R. B. (2021). "Multi-output calibration of a honeycomb seal via on-site surrogates." To appear in *Technometrics*. *arXiv preprint* arXiv:2102.00391. 1332

Huang, J., Gramacy, R. B., Binois, M., and Libraschi, M. (2020). "On-site surrogates for large-scale calibration." *Applied Stochastic Models in Business and Industry*, 36(2): 283–304. *arXiv preprint* arXiv:1810.01903. MR4091796. doi: https://doi.org/10.1002/asmb.2523. 1332

Karagiannis, G., Konomi, B. A., and Lin, G. (2019). "On the Bayesian calibration of expensive computer models with input dependent parameters." *Spatial Statistics*, 34: 100258. Spatio-temporal and geostatistical analysis of hydrological events and related hazards. MR4018089. doi: https://doi.org/10.1016/j.spasta.2017.08.002. 1332

Kennedy, M. C. and O'Hagan, A. (2001). "Bayesian calibration of computer models

(with discussion)." *Journal of the Royal Statistical Society, Series B*, 63(3): 425–464. MR1858398. doi: https://doi.org/10.1111/1467-9868.00294. 1332

Konomi, B. A., Karagiannis, G., Lai, K., and Lin, G. (2017). "Bayesian Treed Calibration: An Application to Carbon Capture With AX Sorbent." *Journal of the American Statistical Association*, 112(517): 37–53. MR3646551. doi: https://doi.org/10.1080/01621459.2016.1190279. 1332

Liu, F., Bayarri, M., and Berger, J. (2009). "Modularization in Bayesian analysis, with emphasis on analysis of computer models." *Bayesian Analysis*, 4(1): 119–150. MR2486241. doi: https://doi.org/10.1214/09-BA404. 1332

Marmin, S. and Filippone, M. (2022). "Deep Gaussian Processes for Calibration of Computer Models." *Bayesian Analysis*, 1–30. doi: https://doi.org/10.1214/21-BA1293. 1332, 1333, 1334, 1335

Plumlee, M. (2017). "Bayesian calibration of inexact computer models." *Journal of the American Statistical Association*, 112(519): 1274–1285. MR3735376. doi: https://doi.org/10.1080/01621459.2016.1211016. 1332

Plumlee, M. (2019). "Computer model calibration with confidence and consistency." *Journal of the Royal Statistical Society: Series B*, 81(3): 519–545. MR3961497. 1332

Salimbeni, H. and Deisenroth, M. (2017). "Doubly stochastic variational inference for deep Gaussian processes." *arXiv preprint* arXiv:1705.08933. 1334

Sampson, P. D. and Guttorp, P. (1992). "Nonparametric estimation of nonstationary spatial covariance structure." *Journal of the American Statistical Association*, 87(417): 108–119. 1332

Sauer, A. (2022). *deepgp: Sequential Design for Deep Gaussian Processes using MCMC*. R package version 1.0.0. 1334

Sauer, A., Cooper, A., and Gramacy, R. B. (2022a). "Vecchia-approximated Deep Gaussian Processes for Computer Experiments." *arXiv preprint* arXiv:2204.02904. 1334

Sauer, A., Gramacy, R. B., and Higdon, D. (2022b). "Active learning for deep Gaussian process surrogates." *Technometrics*, 1–15. doi: https://doi.org/10.1080/00401706.2021.2008505. 1333, 1335

Schmidt, A. M. and O'Hagan, A. (2003). "Bayesian inference for non-stationary spatial covariance structure via spatial deformations." *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 65(3): 743–758. MR1998632. doi: https://doi.org/10.1111/1467-9868.00413. 1332

Snelson, E. and Ghahramani, Z. (2006). "Sparse Gaussian Processes using Pseudo-inputs." *Advances in Neural Information Processing Systems 18*, 1257–1264. URL http://www.gatsby.ucl.ac.uk/~snelson/SPGP_up.pdf. 1334

Surjanovic, S. and Bingham, D. (2013). "Virtual library of simulation experiments: test functions and datasets." http://www.sfu.ca/~ssurjano. 1334

Tuo, R. and Wu, C. F. J. (2015). "Efficient calibration for imperfect computer models."

Annals of Statistics, 43(6): 2331–2352. MR3405596. doi: https://doi.org/10.1214/15-AOS1314. 1332

Tuo, R. and Wu, C. F. J. (2016). "A Theoretical Framework for Calibration in Computer Models: Parameterization, Estimation and Convergence Properties." *Journal of Uncertainty Quantification*, 4: 767–795. MR3523087. doi: https://doi.org/10.1137/151005841. 1332

Vecchia, A. V. (1988). "Estimation and model identification for continuous spatial processes." *Journal of the Royal Statistical Society: Series B (Methodological)*, 50(2): 297–312. MR0964183. 1334

Wong, R. K. W., Storlie, C. B., and Lee, T. C. M. (2017). "A Frequentist Approach to Computer Model Calibration." *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 79(2): 635–648. MR3611763. doi: https://doi.org/10.1111/rssb.12182. 1332

Invited Discussion*

Dave Higdon† and Stephen Walsh‡

Thanks to the authors for this enjoyable and interesting paper. From a narrow perspective, this paper adds some welcome advances to the field of Bayesian computer model calibration. From a broader perspective, this paper gives

- a compelling example of embedding deep Gaussian Processes (DGPs) within an encompassing hierarchical modeling framework, and
- an engineered inference scheme that thoughtfully combines (and trades off) statistical modeling, distributional approximation, computation, and the quantification of uncertainty.

Here we discuss these broad themes, as well as specifics regarding the computer model calibration problem.

This paper makes it clear that the idea of embedding DGPs within larger modeling frameworks is an important one and will likely be so for the foreseeable future. Embedding the model discrepancy within a DGP to account for the difference between computer model and system measurements is particularly innovative.

As algorithms and code for implementing DGP's become more available, it will be more common to see DGPs as part of a larger modeling framework – as it is in this paper. For example, we see them in multifidelity models (Perdikaris et al., 2017; Cutajar et al., 2019); they also show up implicitly in non-stationary spatial models (Schmidt and O'Hagan, 2003) as well as in models for coupling computer models (Kyzyurova et al., 2018). We currently are working on hierarchical models where one of the layers of a DGP links related processes. As a community, I'm sure we'll figure out the settings in which these more complex models can best aid in statistical inference.

This paper/framework is also an excellent example of thoughtfully combining probabilistic modeling, computation and approximation in the service of statistical inference. This work does a great job in leveraging variational approximations and random feature representations so that this framework can handle very high volumes of observation and simulation output.

We note that the statistician's modeling and specification choices can influence even a seemingly fixed model formulation. For example, with the Kennedy and O'Hagan (2001) formulation one has the choice of how to standardize data, represent computer

*This work was supported by the U.S. Department of Energy, Office of Science, Office of Advanced Scientific Computing Research and Office of High Energy Physics, Scientific Discovery through Advanced Computing (SciDAC) program under Award Number 0000231018.

†Department of Statistics, Virginia Tech, dhigdon@vt.edu

‡Department of Statistics, Virginia Tech, walsh124@vt.edu

model (and observation) output, priors that control the GP's, priors for the model parameters, priors for the observation error, etc. This new DGP formulation puts even more choices and controls in the hands of the statistician. We think this is a good thing. But it puts more responsibility on the modeler and makes details of comparisons a bit tricky.

We suspect that a DGP emulator may be a good fit for the contaminant transport problem of Section 4.3. Some of the outputs are concentrations at different times. Here some model parameters control the speed of contaminant transport, moving the spike in concentration to later or earlier times (Figure 7). Such shifts over the support x are not easily captured with standard GPs or additive tree models. However, a DGP that allows one of the latent layers to warp time as model parameters vary could be very well suited to the emulation task in this example.

With very large numbers of computer model runs, uncertainty in emulation often reduces to the point of being negligible in comparison to uncertainty in other features/parameters of the model. In such settings, a number of machine learning-based emulators can be quite advantageous – even when their uncertainty is only characterized approximately.

However, as mentioned in this paper and many others, uncertainty regarding model parameters and model discrepancy does not necessarily decrease with additional model runs and physical measurements. It has been our preference to embrace this joint uncertainty and incorporate it into the posterior inference. This indeterminacy is not a conceptual problem under the Bayesian paradigm (but some approximations may have difficulty accurately describing this joint parameter-discrepancy distribution). In new, more extrapolative settings this wider parameter uncertainty can play an important role in capturing prediction uncertainty. For example, some parameters (often governing material behavior) are not well constrained by initial experiments – especially while allowing for model discrepancy. But new experiments may explore more extreme temperature, pressure, or strain-rate regimes where such parameters are now influential. Admitting the full range of uncertainty allowed by previous experiments is crucial for producing useful prediction uncertainties. This is also crucial for planning future experimental campaigns. With all of this in mind, interpreting comparisons such as those in Fig 4 may not be straightforward. That said, we don't have any better ideas at this stage.

Appropriately capturing uncertainty also highlights the importance of specifying the prior discrepancy distribution. In the additive discrepancy setting, we have some experience and examples to lean on. Defining a prior discrepancy model that's embedded within a DGP framework strikes us as something that's a bit trickier. Do the authors have any thoughts or guidance in this regard?

Once again, our thanks to the authors for advancing the field: producing a self-contained package for handling computer model calibration based on a new DGP model formulation that's flexible and capable of handling much larger datasets.

References

Cutajar, K., Pullin, M., Damianou, A., Lawrence, N., and González, J. (2019). "Deep gaussian processes for multi-fidelity modeling." *arXiv preprint* arXiv:1903.07320. 1339

Kennedy, M. C. and O'Hagan, A. (2001). "Bayesian calibration of computer models." *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 63(3): 425–464. MR1858398. doi: https://doi.org/10.1111/1467-9868.00294. 1339

Kyzyurova, K. N., Berger, J. O., and Wolpert, R. L. (2018). "Coupling computer models through linking their statistical emulators." *SIAM/ASA Journal on Uncertainty Quantification*, 6(3): 1151–1171. MR3845282. doi: https://doi.org/10.1137/17M1157702. 1339

Perdikaris, P., Raissi, M., Damianou, A., Lawrence, N. D., and Karniadakis, G. E. (2017). "Nonlinear information fusion algorithms for data-efficient multi-fidelity modelling." *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 473(2198): 20160751. 1339

Schmidt, A. M. and O'Hagan, A. (2003). "Bayesian inference for non-stationary spatial covariance structure via spatial deformations." *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 65(3): 743–758. MR1998632. doi: https://doi.org/10.1111/1467-9868.00413. 1339

Contributed Discussion*

D. Andrew Brown†

I would first like to offer my congratulations to the authors for a particularly useful step forward in the area of computer model calibration. I echo the Invited Discussants in believing that wedding deep Gaussian processes (DGPs) with the Kennedy-O'Hagan formulation offers a particularly novel way of dealing with model discrepancy, surrogate modeling, and calibration. This approach will likely open new and exciting avenues of research in this area. My brief comments consist of one (mild) concern, and one thought about a different type of calibration to which the authors' proposed method might be suitable.

Quality of the VI Approximation As already discussed and illustrated by the Invited Discussants, I wonder about the quality of the variational approximation to the posterior distribution. One of the selling points of variational inference (VI) is that it can produce measures of uncertainty around the estimators with much less computational effort than full-on Markov chain Monte Carlo. However, most implementations, including the one proposed by Marmin and Filippone here, use a mean-field family of densities, assuming that the posterior can be well approximated by a collection of Gaussian factors. As pointed out by Blei et al. (2017), at times this assumption can be overly restrictive, resulting in a very poor approximation to the true posterior. Even when the approximate posterior concentrates on the same $\boldsymbol{\theta}$ as the true posterior, the associated uncertainty can be quite different from that in the true posterior distribution, possibly resulting in misleading measures of uncertainty — measures which are of course important in the field of UQ. It seems the only way to really verify if one's VI approximation is adequate is to run MCMC extensively under the same model and compare the results, which of course defeats the purpose of VI.

Extension to Functional Calibration Relatively recently, engineers and other practitioners have recognized that in the computer model calibration problem, it is sometimes preferable to allow the calibration parameters to vary as a function of the control inputs; i.e., $\boldsymbol{\theta} \equiv \boldsymbol{\theta}(\boldsymbol{x})$, resulting in so-called functional calibration (e.g., Plumlee et al., 2016; Brown and Atamturktur, 2018; Tuo et al., 2021). This phenomenon has been observed in plastic deformation models (Brown and Atamturktur, 2018), burning of nuclear fuels (Unal et al., 2013), resistance spot welding (Ezzat et al., 2018), buckypaper manufacturing (Pourhabib et al., 2015), and elsewhere. The framework proposed by Marmin and Filippone seems as though it could extend naturally to functional calibration; i.e., by modifying the graph in Figure 2 so that $\boldsymbol{x}$ feeds into the $\boldsymbol{\theta}$ nodes. Depending on one's goals, this may or may not introduce additional challenges. For instance, as alluded to by Marmin and Filippone and the Invited Discussants, there is sometimes an

*This work is partially supported by National Science Foundation Grant DMS-2210686.

†School of Mathematical and Statistical Sciences, Clemson University, Clemson, SC 29634-0975, ab7@clemson.edu

interest in recovering a true, physically-meaningful $\boldsymbol{\theta}$. In the framework of functional calibration, for instance, so-called partitioned analysis can treat $\boldsymbol{\theta}(\boldsymbol{x})$ as an unobservable constituent physical system in which "calibration" amounts to empirically estimating the constituent system so that the computer code may be partitioned into smaller, modular codes (Stevens et al., 2018; Flynn et al., 2019). While there is typically little interest in what is going on in the intermediate layers of a neural network, it seems that (meaningfully) estimating the calibration functions would, in fact, put one's interest on at least the first layer of the DGP (the $\boldsymbol{\theta}$-layer in the modified graph). Do the authors have thoughts as to whether or not this would even be feasible? Or does the ubiquitous identifiability problem render the situation hopeless?

References

Blei, D. M., Kucukelbir, A., and McAuliffe, J. D. (2017). "Variational inference: A review for statisticians." *Journal of the American Statistical Association*, 112(518): 859–877. MR3671776. doi: https://doi.org/10.1080/01621459.2017.1285773. 1342

Brown, D. A. and Atamturktur, S. (2018). "Nonparametric functional calibration of computer models." *Statistica Sinica*, 28: 721–742. MR3791085. 1342

Ezzat, A. A., Pourhabib, A., and Ding, Y. (2018). "Sequential design for functional calibration of computer models." *Technometrics*, 60(3): 286–296. MR3847166. doi: https://doi.org/10.1080/00401706.2017.1377638. 1342

Flynn, G. S., Chodora, E., Atamturktur, S., and Brown, D. A. (2019). "A Bayesian inference-based approach to empirical training of strongly coupled constituent models." *ASME Journal of Verification, Validation, and Uncertainty Quantification*, 4: 021005. 1343

Plumlee, M., Joseph, V. R., and Yang, H. (2016). "Calibrating functional parameters in the ion channel models of cardiac cells." *Journal of the American Statistical Association*, 111(514): 500–509. MR3538682. doi: https://doi.org/10.1080/01621459.2015.1119695. 1342

Pourhabib, A., Huang, J. Z., Wang, K., Zhang, C., Wang, B., and Ding, Y. (2015). "Modulus prediction of buckypaper based on multi-fidelity analysis involving latent variables." *IIE Transactions*, 47: 141–152. 1342

Stevens, G., Atamturktur, S., Brown, D. A., Williams, B. J., and Unal, C. (2018). "Statistical inference of empirical constituents in partitioned analysis from integral-effect experiments: An application in thermo-mechanical coupling." *Engineering Computations*, 35(2): 672–691. 1343

Tuo, R., He, S., Pourhabib, A., Ding, Y., and Huang, J. (2021). "A reproducing kernel Hilbert space approach to functional calibration of computer models." *Journal of the American Statistical Association*. 1342

Unal, C., Stull, C. J., and Williams, B. J. (2013). "Parametric uncertainty in the thermal conductivity model of uranium oxide light water nuclear reactor fuel." *Review of Applied Physics*, 2(3): 39–48. 1342

Rejoinder

Sébastien Marmin* and Maurizio Filippone†

1 Introduction

We feel privileged to be able to reflect back on our work and comment on a number of aspects of our paper which we could improve on. For this unique opportunity, we would like to express our gratitude to the Editorial Board and the Editor-in-Chief Professor Steel for selecting our work to be a suitable one to be discussed. In addition, we think it is important to thank the reviewers who helped us tremendously to improve our original submission, and the discussants for bringing up a number of important points which we have the opportunity to elaborate on.

In the following we will refer to authors of the four discussions as follows: S&G for Mrs Sauer and Professor Gramacy, H&W for Professor Higdon and Mr Walsh, O'H for Professor O'Hagan, and Br for Professor Brown. We structured our rejoinder in two main parts. The first part addresses comments on uncertainty in model parameters, the choice of DGP approximations, the choice of inference approaches, and some considerations on the landscape (or the "Wild West" quoting S&G) of software packages which researchers and practitioners can rely on. The second part is dedicated to a concern shared by all discussants on how to impose sensible priors on the discrepancy when such beliefs are expressed by DGPs.

2 Uncertainty in Model Parameters

We fully agree with H&W about the importance of being able to produce useful uncertainties, and for this reason we believe it is important to question our choice to approximate GPs with random features and to employ VI, as pointed out by S&G. A first draft of this work was completed in October 2018, and in the last four years there have been several advances in the GP literature and MCMC sampling, which suggest that there are definitely better alternatives. In our group, right after we completed this work, we realized that VI can be problematic not only for DGPs approximated with random features but also for other classes of over-parameterized models. The reason is that the variational objective (Eq. 2.15 in the main paper) contains one term capturing how well the approximation models the observations, which scales with the number of observations, and another which acts as a regularization term, which scales with the number of parameters. When the number of parameters is much larger than the number of observations, VI tends to return an approximate posterior which is equal to the prior, meaning that the likelihood term is completely disregarded. This problem is so acute

*French National Metrology Institute, 1 Rue Gaston Boissier, 75724 PARIS Cedex 15, France, sebastien.marmin@lne.fr

†EURECOM, Campus SophiaTech, 450 Route des Chappes, 06410 Biot, France, maurizio.filippone@eurecom.fr

that, to the best of our knowledge, there hasn't been any significant application of VI to carry out approximate inference for popular models such as Convolutional Neural Networks (CNNs) before Gal and Ghahramani (2016), who proposed an interpretation of dropout as VI. We believe that this is remarkable in the context of the deep learning revolution, because it shows how much harder it is to treat such complex models in a Bayesian way instead of optimizing their parameters.

We carried out some work to address this problem by improving the initialization of VI (Rossi et al., 2019) and by reducing the parameterization of intermediate layers (Rossi et al., 2020), both with some degree of success. We implemented the improved initialization in this work, and we noticed that we could obtain a consistently better optimization compared to random initializations. Another possibility to improve VI would be to use normalizing flows (Rezende and Mohamed, 2015), which allow for a more flexible class of variational approximations, or a better variational objective (Burda et al., 2016; Salimbeni et al., 2019). We believe that these innovations could significantly improve the quality of uncertainties in the framework proposed in our work.

2.1 Variational Inference vs MCMC

BR rightfully points out the VI with a factorized approximation scales well with the number of parameters but it tends to yield poor uncertainties. This is something to be fully aware of, as H&W highlight in their discussion too. Due to this consideration and the difficulty in working with VI in over-parameterized models, in the years following the completion of a draft of this work, we increased our focus on scalable MCMC. In particular, we adopted Stochastic Gradient Hamiltonian Monte Carlo (Chen et al., 2014, SG-HMC) for DGPs (Rossi et al., 2021) and Bayesian DNNs (Tran et al., 2021, 2022), with excellent results, especially when adopting the cyclical step-size approach proposed by Zhang et al. (2020), which makes tuning easier.

Because SG-HMC operates on mini-batches like stochastic VI, the two solutions are comparable in asymptotic complexity, but in some cases the quality of the uncertainties obtained by the MCMC approach are vastly superior. Arguably, the number of SG-HMC iterations needed to test and to achieve convergence is very large, making VI an attractive alternative, but the optimization problem in VI may be surprising slow to converge too in some cases. For applications where uncertainty is a primary requirement, and where running the computer model is expensive, today we do not see any reason to rule out the use SG-HMC. To convince skeptical readers, in some recent works we have carried out experiments involving millions of data points and rather complicated models, including DGPs, Convolutional DGPs, Bayesian DNNs and CNNs (see, e.g., Rossi et al. (2021); Tran et al. (2021, 2022)).

One reason to prefer VI over SG-HMC could be that VI lends itself to problems where observations are collected sequentially (e.g., adaptive experimental design scenarios) given that the approximate posterior can easily be used as a prior for new observations. Also, VI could be a useful tool in an exploration phase to obtain some preliminary insights on a calibration problem before running SG-HMC. We agree with S&G that it is important to understand the cost of employing approximations and what we can expect in terms of quality of uncertainties in the emulation.

H&W and S&G are right that flexibility is a good thing, and that with more flexibility comes more responsibility. We believe that the Bayesian approach offers the right tools to answer questions related to model selection. VI offers a proxy for the model evidence; however, this needs to be interpreted and used with care and recent studies show how this can be improved (Chen et al., 2018). It is well known that estimating the model evidence with MCMC poses some challenges, but it is possible (e.g., via thermodynamic integration (Friel and Pettitt, 2008) or Annealed Importance Sampling (Neal, 2001)), and as far as we know, this has not been studied in detail for DGPs and other Bayesian deep models. Model selection is expensive, but definitely something to consider if one can afford to do so.

2.2 Code and Software Packages

It is great to see the comparison carried out by S&G in their discussion despite the "Wild West" of GP/DGP packages out there. We know from experience how painful it can be to carry out comprehensive and fair comparisons. A rigorous comparison across software packages and GP/DGP implementations is difficult to do due to, e.g., issues with software environments/dependencies, tuning of optimization (hyper-)parameters, design choices for model architectures, limited choices of approximations/inference methods, or unmaintained code. Producing software which addresses all these limitations while keeping a general framework for inference in GPs/DGPs is difficult unless code is constantly revised and updated; as a good example, it is worth mentioning `GPflow` (Matthews et al., 2017), which is maintained and updated by a long list of researchers and engineers, and this arguably makes it one of the most complete and up-to-date packages for scalable GPs/DGPs (which however does not feature random feature approximations!). We refer the reader to the following url[1] for a comprehensive list of software packages implementing GPs and DGPs with various approximations. It is great to see that S&G collected their work in a much needed software package in `R`. On our side, we regret that we haven't had the bandwidth to turn our code, which is available to reproduce all the experiments in the paper, into a complete software package. However, the discussion encourages us to remedy this, and we hope to release a software package featuring DGP calibration soon.

3 Incorporating Prior Knowledge into DGPs

We dedicate this section to the issue of incorporating prior knowledge into DGPs, which we believe to be very important in applications involving the calibration of computer models. We totally agree with H&W and O'H that defining a prior over the discrepancy through a DGP can be tricky. GP priors offer a natural way to specify distributions over functions with some properties of interest, and this can be done by choosing an appropriate covariance function. This makes it relatively easy to encode information about, e.g., smoothness, marginal variance/amplitude, periodicity, and length-scale. By composing GPs into DGPs we trade this "interpretable" way to specify priors for increased

[1] https://en.wikipedia.org/wiki/Comparison_of_Gaussian_process_software

flexibility. A worrying consequence is that blindly choosing the covariance of the GP layers may yield a DGP prior distribution representing mostly pathological functions, such as constant functions (Duvenaud et al., 2014; Neal, 1996). This is undesirable from a Bayesian perspective because this can potentially put a lot of prior mass on functions which are not interesting *a priori*.

It is great that the discussions raised this important point so that we can elaborate on this issue, which we had the opportunity to work on over the last couple of years. We recently studied this problem in detail by looking at ways to impose sensible priors over the functions that can be modeled by deep models, including DGPs, Bayesian DNNs and Bayesian CNNs. For these models, we need to specify a prior over the parameters of all the layers. However, the composition operation which characterizes deep models makes it extremely difficult to determine how the choice of priors on the parameters of all the layers affects the distribution over functions at the output of these models.

In Tran et al. (2022) we propose a novel way to optimize the prior over the parameters of deep modes so as to obtain distributions over functions with some desirable properties. In particular, we consider the scenario where GPs are used to encode such desirable properties, so we focus on GPs as target functional priors. The optimization uses an objective based on the Wasserstein distance and it is fully sample-based, meaning that we can target any functional prior as long as we are able to draw samples from it, and so we are not restricted to GPs. In Tran et al. (2022) we showcase a large number of successful applications of this idea to Bayesian DNNs and Bayesian CNNs, and the extension to DGPs is straightforward; instead of optimizing the prior over parameters of each layer, we just need to optimize the covariance of the GPs at each layer. We believe that this approach can guide practitioners to choose sensible priors for their emulator and discrepancy in applications involving the calibration of computer models.

4 Conclusions

We conclude by providing some final comments on a couple of other points raised in the discussions. BR discusses an interesting scenario where parameters $\boldsymbol{\theta}$ depend on the inputs $\mathbf{x}$. We could impose a GP/DGP prior on $\boldsymbol{\theta}(\mathbf{x})$, and because this is an input of the emulator, which is itself a GP/DGP, it should be possible to handle the resulting model with the framework proposed in our work. The identifiability issues of the original DGP calibration framework should then apply to this more general variant too.

Finally, an interesting point raised by S&G is whether in the regime of large number of simulations/observations it makes sense to go through the trouble of employing DGPs, as GPs may suffice. We think that it is difficult to offer general guidelines on how much data is enough to be able to safely rely on GPs, especially in large dimensional settings. We believe that combining flexible models with sensible functional priors is beneficial in many applications, and recent advances in these domains make this a concrete possibility.

References

Burda, Y., Grosse, R. B., and Salakhutdinov, R. (2016). "Importance Weighted Autoencoders." In *4th International Conference on Learning Representations, ICLR 2016, San Juan, Puerto Rico, May 2-4, 2016, Conference Track Proceedings*. 1346

Chen, L., Tao, C., Zhang, R., Henao, R., and Duke, L. C. (2018). "Variational Inference and Model Selection with Generalized Evidence Bounds." In Dy, J. and Krause, A. (eds.), *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, 893–902. PMLR. 1347

Chen, T., Fox, E., and Guestrin, C. (2014). "Stochastic Gradient Hamiltonian Monte Carlo." In Xing, E. P. and Jebara, T. (eds.), *Proceedings of the 31st International Conference on Machine Learning*, volume 32 of *Proceedings of Machine Learning Research*, 1683–1691. Bejing, China: PMLR. 1346

Duvenaud, D. K., Rippel, O., Adams, R. P., and Ghahramani, Z. (2014). "Avoiding pathologies in very deep networks." In *Proceedings of the Seventeenth International Conference on Artificial Intelligence and Statistics, AISTATS 2014, Reykjavik, Iceland, April 22-25, 2014*, volume 33 of *JMLR Workshop and Conference Proceedings*, 202–210. JMLR.org. 1348

Friel, N. and Pettitt, A. N. (2008). "Marginal likelihood estimation via power posteriors." *Journal of the Royal Statistical Society Series B*, 70(3): 589–607. MR2420416. doi: https://doi.org/10.1111/j.1467-9868.2007.00650.x. 1347

Gal, Y. and Ghahramani, Z. (2016). "Dropout as a Bayesian Approximation: Representing Model Uncertainty in Deep Learning." In Balcan, M. F. and Weinberger, K. Q. (eds.), *Proceedings of The 33rd International Conference on Machine Learning*, volume 48 of *Proceedings of Machine Learning Research*, 1050–1059. New York, New York, USA: PMLR. 1346

Matthews, A. G. d. G., van der Wilk, M., Nickson, T., Fujii, K., Boukouvalas, A., León-Villagrá, P., Ghahramani, Z., and Hensman, J. (2017). "GPflow: A Gaussian process library using TensorFlow." *Journal of Machine Learning Research*, 18(40): 1–6. MR3646635. 1347

Neal, R. M. (1996). *Bayesian Learning for Neural Networks (Lecture Notes in Statistics)*. Springer, 1 edition. 1348

Neal, R. M. (2001). "Annealed importance sampling." *Statistics and Computing*, 11(2): 125–139. MR1837132. doi: https://doi.org/10.1023/A:1008923215028. 1347

Rezende, D. and Mohamed, S. (2015). "Variational Inference with Normalizing Flows." In Bach, F. and Blei, D. (eds.), *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, 1530–1538. Lille, France: PMLR. 1346

Rossi, S., Heinonen, M., Bonilla, E., Shen, Z., and Filippone, M. (2021). "Sparse Gaussian Processes Revisited: Bayesian Approaches to Inducing-Variable Approximations." In Banerjee, A. and Fukumizu, K. (eds.), *Proceedings of The 24th Interna-*

tional Conference on Artificial Intelligence and Statistics, volume 130 of *Proceedings of Machine Learning Research*, 1837–1845. PMLR. 1346

Rossi, S., Marmin, S., and Filippone, M. (2020). "Walsh-Hadamard Variational Inference for Bayesian Deep Learning." In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., and Lin, H. (eds.), *Advances in Neural Information Processing Systems*, volume 33, 9674–9686. Curran Associates, Inc. 1346

Rossi, S., Michiardi, P., and Filippone, M. (2019). "Good Initializations of Variational Bayes for Deep Models." In Chaudhuri, K. and Salakhutdinov, R. (eds.), *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, 5487–5497. PMLR. 1346

Salimbeni, H., Dutordoir, V., Hensman, J., and Deisenroth, M. (2019). "Deep Gaussian Processes with Importance-Weighted Variational Inference." In Chaudhuri, K. and Salakhutdinov, R. (eds.), *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, 5589–5598. PMLR. 1346

Tran, B.-H., Rossi, S., Milios, D., and Filippone, M. (2022). "All You Need is a Good Functional Prior for Bayesian Deep Learning." *Journal of Machine Learning Research*, 23(74): 1–56. 1346, 1348

Tran, B.-H., Rossi, S., Milios, D., Michiardi, P., Bonilla, E. V., and Filippone, M. (2021). "Model Selection for Bayesian Autoencoders." In Ranzato, M., Beygelzimer, A., Dauphin, Y., Liang, P., and Vaughan, J. W. (eds.), *Advances in Neural Information Processing Systems*, volume 34, 19730–19742. Curran Associates, Inc. 1346

Zhang, R., Li, C., Zhang, J., Chen, C., and Wilson, A. G. (2020). "Cyclical Stochastic Gradient MCMC for Bayesian Deep Learning." In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. 1346