Statistica Sinica **28** (2018), 721-742
doi:https://doi.org/10.5705/ss.202015.0344

NONPARAMETRIC FUNCTIONAL CALIBRATION OF COMPUTER MODELS

D. Andrew Brown and Sez Atamturktur

Clemson University

Abstract: Standard methods in computer model calibration treat the calibration parameters as constant throughout the domain of control inputs. In many applications, systematic variation may cause the best values for the calibration parameters to change across different settings. When not accounted for in the code, this variation can make the computer model inadequate. We propose a framework for modeling the calibration parameters as functions of the control inputs to account for a computer model's incomplete system representation in this regard, while simultaneously allowing for possible constraints imposed by prior expert opinion. We demonstrate how inappropriate modeling assumptions can mislead a researcher into thinking a calibrated model is in need of an empirical discrepancy term when it is only needed to allow for a functional dependence of the calibration parameters on the inputs. We apply our approach to plastic deformation of a visco-plastic self-consistent material in which the critical resolved shear stress is known to vary with temperature.

Key words and phrases: Bayesian statistics, Gaussian process, identifiability, model validation, uncertainty quantification, visco-plastic self-consistent material.

1. Introduction

Many physical phenomena studied in engineering and science disciplines are driven by complex processes that may only be partially understood. Experiments are needed to better understand these processes, but conducting them may be difficult due to economic, technical, or ethical limitations. In response to the need to study such prohibitively resource-intensive systems, the use of computer simulations as proxies for physical observations is now common practice. The design and analysis of computer experiments has become a critical tool in the advancement of numerous fields including national defense, environmental protection, medicine, and manufacturing.

The utility of any computer model is contingent upon that model's fidelity to physical reality. Determining whether or not a specific computer code is an acceptable surrogate for reality falls under the purview of model validation and

the closely related area of model calibration. The aim of computer model calibration is to find appropriate values of the parameters governing the computer code under which the code will most closely approximate physical observations according to a predefined metric. Standard methods in computer model calibration treat the calibration parameters as fixed (or averaged) values that are constant throughout the domain of control inputs (e.g., Kennedy and O'Hagan (2001); Williams et al. (2006); Bayarri et al. (2007); Higdon et al. (2008)). Computer output and physical data then are combined to obtain the posterior distribution of the calibration parameters. The posterior distribution serves as the basis for calibrating the computer code in which the calibration parameters are set to a point estimate such as the posterior mode (Kennedy and O'Hagan (2001)) or varied over the plausible range for making predictions of future responses (e.g., Reese et al. (2004); Kennedy et al. (2006); Higdon et al. (2008)).

Often in practice the best settings for the calibration parameters may change with different settings of the control inputs (Fugate et al. (2006); Atamturktur et al. (2015); Pourhabib et al. (2015); Plumlee, Joseph and Yang (2016)). This may be due to differences between manufacturing runs, raw materials, etc., or systematic variation not accounted for in the computer code due to incomplete knowledge of the system or computational difficulties. The former case was considered by Xiong et al. (2009), who used a hierarchical model to treat the calibration parameters as realizations from a common distribution with parameters estimated via maximum likelihood. The purpose of this article is to treat the latter case by modeling the calibration parameters as functions of the control inputs. To fully account for the uncertainty associated with the unknown functional form, we use a Gaussian process prior while allowing for constraints imposed by opinions of subject matter experts, as is conventionally done in computer experiments. Functional calibration is a topic of interest to many science and engineering fields. Recently, Plumlee, Joseph and Yang (2016) presented a case study in which the calibration parameters are similarly modeled with Gaussian process priors to capture functional dependence for the specific application of the ion channel models of cardiac cells. With this paper, we contribute to solving the problem in applications where available experimental data are scarce, in which case the use of expert-elicited prior constraints becomes necessary to address identifiability issues.

Our aim here is to propose a general framework for nonparametrically modeling calibration parameters as smooth functions of the control inputs. We provide guidance for implementing our so-called state-aware calibration by discussing

practical computational considerations, identifiability issues, and determining when to invoke state-aware analysis. We demonstrate the feasibility and performance of our model through an extensive simulation study as well as an application to plastic deformation of a visco-plastic self-consistent (VPSC) material in which the critical resolved shear stress varies with temperature.

The remainder of this paper is organized as follows: In Section 2 we briefly review existing approaches, including the framework of Kennedy and O'Hagan (2001), and state our proposed nonparametric functional calibration model. We explicate a special case of our model relevant to the VPSC application, including a discussion of computational considerations when implementing the model via Markov chain Monte Carlo. This is followed by a simulation study in Section 3 comparing our model under different sets of prior constraints with a model assuming a known parametric functional form of the dependence, and with a model that treats all calibration parameters as fixed throughout the experimental domain. We apply our proposed model to the VPSC problem in Section 4. We conclude with discussion of these results, suggestions for determining when functional calibration is necessary, and thoughts about future research in Section 5.

2. Methods

2.1. General formulation

A key reference for our development is Kennedy and O'Hagan (2001), but the notions of model validation and calibration appear at least as early as Berman and Nagy (1983) and Park (1991). Early Bayesian perspectives on calibration can be found in Craig et al. (2001) and Reese et al. (2004), with a maximum likelihood approach being presented in Loeppky, Bingham and Welch (2006). Methods for integrating field data and computer output for calibration and analysis appear in Higdon et al. (2004) and Williams et al. (2006). Bayarri et al. (2007) suggested a framework for the model validation process, including calibration. Computer models with high-dimensional output were calibrated using basis function representations in Higdon et al. (2008). Joseph and Melkote (2009) modified the approach of Kennedy and O'Hagan (2001) to separate estimation of calibration parameters from determination of a functional form for the model discrepancy. The determination of appropriate values of tuning parameters and calibration parameters simultaneously was done in Han, Santner and Rawlinson (2009). Calibration was extended to computer models for nonstationary spatiotemporal processes in Pratola et al. (2013). Tuo and Wu (2015, 2016) discussed

calibration based on L_2 projections and studied the estimators' asymptotic properties compared to the method of ordinary least squares. Pourhabib et al. (2015) treated the calibration parameters as latent variables and used monotone sums of splines to represent the functional relationship between the latent variables and control inputs. Nonparametric functional calibration ideas appear also in Atamturktur and Brown (2015) and Plumlee, Joseph and Yang (2016).

Suppose we have N field observations taken at experimental design settings $\mathbf{x}_1, \ldots, \mathbf{x}_N$, where $\mathbf{x}_i \in [0,1]^{d_x}$, $i = 1, \ldots, N$, $d_x \geq 1$. Denote the field data as $y_i = y(\mathbf{x}_i)$, $i = 1, \ldots, N$. Let $\eta(\mathbf{x}, \mathbf{t})$ denote the output of the approximating computer code using control input $\mathbf{x}$ and calibration parameter input $\mathbf{t}$. Here we assume that the computer code is fast-running so that a surrogate is not needed to emulate the computer output. Suppose that any discrepancy between the computer output and the field data is solely due to misspecified parameters in the computer model and measurement error. The field data then can be modeled as

$$y_i = \eta(\mathbf{x}_i, \boldsymbol{\theta}) + \varepsilon_i, \quad i = 1, \ldots, N, \tag{2.1}$$

where $\boldsymbol{\theta}$ is the vector of true parameter values under which the computer model agrees with reality. We assume that $\boldsymbol{\varepsilon} = (\varepsilon_1, \ldots, \varepsilon_N)^T \sim N_N(\mathbf{0}, \lambda_y^{-1}\mathbf{I})$, where $\mathbf{I}$ is the identity matrix and $\lambda_y > 0$.

Consider a situation in which $\boldsymbol{\theta}$ depends on the particular settings of the experiment (e.g., Xiong et al. (2009); Atamturktur et al. (2015); Pourhabib et al. (2015); Plumlee, Joseph and Yang (2016)). In the case $\dim(\boldsymbol{\theta}) > 1$, we partition the calibration parameters as $\boldsymbol{\theta}(\mathbf{x}) = (\boldsymbol{\theta}_1^T(\mathbf{x}), \boldsymbol{\theta}_2^T)^T$, where $\boldsymbol{\theta}_1(\cdot) = (\theta_{11}(\cdot), \ldots, \theta_{1p}(\cdot))^T$ is the vector of state-aware calibration parameters and $\boldsymbol{\theta}_2$ contains the constant parameters. Suppose *a priori* that $\boldsymbol{\theta}_1(\cdot)$ is independent of $\boldsymbol{\theta}_2$. To accommodate the dependence of $\boldsymbol{\theta}_1$ on $\mathbf{x}$ without assuming a functional form of the dependence, we use a nonparametric model for the components of $\boldsymbol{\theta}_1(\mathbf{x})$. Specifically, we appeal to Gaussian process (GP) models (O'Hagan (1978); Neal (1998); Santner, Williams and Notz (2003)). For $\boldsymbol{\theta}_2$, we follow convention and assign the elements independent uniform priors (Higdon et al. (2008)).

Assuming independence *a priori* among all calibration parameters allows us to define the prior distribution on $\boldsymbol{\theta}$ as $\pi(\boldsymbol{\theta}(\mathbf{x})) = \pi_1(\boldsymbol{\theta}_1(\mathbf{x}))\pi_2(\boldsymbol{\theta}_2)$, where we assign Gaussian process priors independently to the elements of $\boldsymbol{\theta}_1(\cdot)$. In the absence of prior knowledge and to limit the number of parameters to be estimated, we use Gaussian processes with constant mean functions, usually sufficient for interpolating GP models (Neal (1998); Bayarri et al. (2007)). We wish to honor

any expert-elicited bounds on plausible values for functional parameters as we would under conventional calibration. Hence, we scale all of the computer code inputs to lie in the unit hypercube and connect the functional calibration parameters to the GP models through a known link function mapping the unit interval to the real line, as done in generalized linear models (GLMs; McCullagh and Nelder (1989)). Here we suppose that the functional calibration parameters vary smoothly over the control inputs, so the relationships can be well approximated by infinitely differentiable functions. Hence, we use a Gaussian correlation function. We have, for $i = 1, \ldots, p$,

$$g(\theta_{1i}(\cdot)) \overset{\text{indep.}}{\sim} \mathcal{GP}(\mu_{\theta,i},\ \lambda_{\theta,i}^{-1} R_i(\cdot,\cdot)); \quad R_i(\mathbf{x}, \mathbf{x}') = \exp\left\{-4 \sum_{k=1}^{d_x} \gamma_{\theta,i,k} |x_k - x_k'|^2\right\}, \tag{2.2}$$

where $g : (0,1) \to \mathbb{R}$ is one-to-one and differentiable, $d_x = \dim(\mathbf{x})$, $\lambda_{\theta,i}$ are the unknown precisions, and $\gamma_{\theta,i,k}$ controls the smoothness of the sample paths of $\theta_{1i}(\cdot)$ along the k^{th} dimension of $\mathbf{x}$. The mean functions $\mu_{\theta,i}$ are taken to be constant and fixed. For instance, if we take g to be the logit link, then we can center the GPs around $\log(0.5/(1-0.5)) = 0$, and likewise for other link functions. If we know *a priori* that $\theta_{1i}(\cdot)$ is bounded away from 0 and 1 with high probability, then we can take $g(\theta_{1i}) = \theta_{1i}$ as an approximation to the response function. In Section 3, we compare the performance of the logit, $g(z) = \log(z/(1-z))$, probit (inverse Gaussian distribution function), $g(z) = \Phi^{-1}(z)$, cumulative log-log (c-log-log), $g(z) = \log(-\log(z))$, and identity, $g(z) = z$, functions on a simulated example and show that they are all comparable.

When stronger plausible limits are known for the calibration parameters at certain input settings, we can modify the preceding model to

$$g(\theta_{1i}(\cdot)) \overset{\text{indep.}}{\sim} \mathcal{GP}(\mu_{\theta,i},\ \lambda_{\theta,i}^{-1} R_i(\cdot,\cdot)) \prod_{c \in C_i} I(L_c < \theta_{1i}(\mathbf{x}_c) < U_c),\ i = 1, \ldots, p, \tag{2.3}$$

for finite sets of constraints indexed by C_i with L_c, U_c being the bounds and $I(\cdot)$ the indicator function. In practice, one can use standard techniques for sampling from truncated Gaussian distributions (e.g., Robert (1995)). In some cases, it may even be feasible to draw from the unrestricted sample paths and discard those not satisfying the constraints, making implementation easy.

The model is completed by specifying priors for the hyperparameters in each Gaussian process. For hyperpriors on the parameters governing the covariance structure of the GP, it is convenient to parameterize the correlation function as $\rho_{\theta,i,k} = e^{-\gamma_{\theta,i,k}}$ and to assign $\rho_{\theta,i,k}$ independent Beta priors, $\rho_{\theta,i,k} \overset{\text{iid}}{\sim}$

Beta$(1, b_\theta)$, $i = 1, \ldots, p$; $k = 1, \ldots, d_x$ (Williams et al. (2006)). The shape parameter b_θ is chosen to place most probability mass near one to enforce the assumed smoothness *a priori*, say $b_\theta = 0.1$ or $b_\theta = 0.2$. We take $\lambda_{\theta,i} \overset{\text{iid}}{\sim} \text{Ga}(a_\theta, b_\theta)$. If g is the identity function in (2.2), then we can choose a_θ and b_θ to place the prior probability mass around one, since the calibration parameters are scaled. Otherwise, we can take, e.g., $a_\theta = 0.01$ and $b_\theta = 0.01$ so that the prior is centered at one with standard deviation $\sqrt{0.01/0.01^2} = 10$. Similarly, we take the error precision to be $\lambda_y \sim \text{Ga}(a_y, b_y)$. Since the data are standardized when calibrating the computer code, we again choose the parameters to concentrate the density near one.

A common problem in computer model calibration is that of identifiability of the calibration parameters (Bayarri et al. (2007)). Bayesian modeling can mitigate this problem through informative prior distributions (Gelfand and Sahu (1999); Gustafson (2005)). In our case, the GP induces correlation between $\boldsymbol{\theta}_1(\mathbf{x}_i)$ and $\boldsymbol{\theta}_1(\mathbf{x}_j)$, $\mathbf{x}_i \neq \mathbf{x}_j$, so that they are allowed to share information in determining plausible values in the posterior. However, GP models tend to be erratic near the boundaries of the domains over which they are studied, and this behavior can limit the Bayesian learning about $\boldsymbol{\theta}_1(\cdot)$ or $\boldsymbol{\theta}_2$ in the posterior. Another consequence of weak identifiability is the possibility of highly correlated draws in the MCMC sampling routine, potentially leading to very poor convergence properties. A possible solution is to elicit informative prior distributions from subject matter experts. If an informative prior distribution can be elicited for $\boldsymbol{\theta}_2$, or if the possible sample paths of $\boldsymbol{\theta}_1(\cdot)$ can be constrained using prior information, identifiability can be improved. We return to this point in Section 3.

A goal of computer model calibration is to facilitate reliable predictions at untested experimental settings. In the Bayesian paradigm, such predictions are based on the posterior predictive distribution. Suppose we have training data $\mathbf{y} = (y(\mathbf{x}_1), \ldots, y(\mathbf{x}_N))^T$ and we wish to make predictions for future realizations at m untested settings $\mathbf{x}_1^*, \ldots, \mathbf{x}_m^*$, $\mathbf{y}^* = (y(\mathbf{x}_1^*), \ldots, y(\mathbf{x}_m^*))^T$. Since $\mathbf{y}^*$ is determined by $\boldsymbol{\theta}_1^{(\mathbf{x}^*)} := (\boldsymbol{\theta}_1^T(\mathbf{x}_1^*), \ldots, \boldsymbol{\theta}_1^T(\mathbf{x}_m^*))^T$, $\boldsymbol{\theta}_2$, and λ_y, and *a posteriori* information about $\boldsymbol{\theta}_1^{(\mathbf{x}^*)}$ depends on $\mathbf{y}$ only through the posterior distribution of $\boldsymbol{\theta}_1^{(\mathbf{x})} = (\boldsymbol{\theta}_1^T(\mathbf{x}_1), \ldots, \boldsymbol{\theta}_1^T(\mathbf{x}_N))^T$, $\boldsymbol{\rho}_\theta = (\rho_{\theta,1}, \ldots, \rho_{\theta,p})^T$, and $\boldsymbol{\lambda}_\theta = (\lambda_{\theta,1}, \ldots, \lambda_{\theta,p})^T$, predictions at untested settings are available by drawing $\boldsymbol{\theta}_1^{(\mathbf{x})}, \boldsymbol{\rho}_\theta, \boldsymbol{\lambda}_\theta, \boldsymbol{\theta}_2$, and λ_y from the joint posterior distribution, sampling from the distribution of $\boldsymbol{\theta}_1^{(\mathbf{x}^*)} | \boldsymbol{\theta}_1^{(\mathbf{x})}, \boldsymbol{\rho}_\theta, \boldsymbol{\lambda}_\theta$, and then drawing from $\mathbf{y}^* | \boldsymbol{\theta}_1^{(\mathbf{x}^*)}, \boldsymbol{\theta}_2, \lambda_y$. Here, $\pi(\boldsymbol{\theta}_1^{(\mathbf{x}^*)} | \boldsymbol{\theta}_1^{(\mathbf{x})}, \boldsymbol{\rho}_\theta, \boldsymbol{\lambda}_\theta)$ is readily available since $(\boldsymbol{\theta}_1^{(\mathbf{x}^*)}, \boldsymbol{\theta}_1^{(\mathbf{x})}) | \boldsymbol{\rho}_\theta, \boldsymbol{\lambda}_\theta$ follows a multivariate Gaussian distribution.

When an experimenter has more reliable prior information concerning the

functional forms of the dependencies of calibration parameters on the control settings, the nonparametric Gaussian process in (2.2) can be replaced with a parametric function, $\boldsymbol{\theta}(\mathbf{x}) = f(\mathbf{x}, \boldsymbol{\beta})$. The problem then is to assign an appropriate prior distribution to $\boldsymbol{\beta}$ and estimate plausible values from the posterior. This was done in Xiong et al. (2009) and Atamturktur et al. (2015), with a similar approach taken in Pourhabib et al. (2015). The parametric calibration problem can be expressed as a standard calibration approach, though, since the calibration parameters are still treated as constant and appear in the "augmented" computer code, $\eta(\mathbf{x}, \boldsymbol{\theta}(\mathbf{x})) = \eta(\mathbf{x}, f(\mathbf{x}, \boldsymbol{\beta})) \equiv \eta(\mathbf{x}, \boldsymbol{\beta})$.

2.2. Two parameter model with scalar control input

Our motivating example of modeling plastic deformation of viscoplastic self-consistent material involves a single control input and two calibration parameters so that $p = 1$ and $d_x = 1$ in (2.2). In light of this, it is the scenario we consider in our simulation study in Section 3. We focus on this special case and consider the specification of the model, the joint posterior distribution, and computational considerations for implementation.

Let $\mathbf{y} = (y(x_1), \ldots, y(x_N))^T$ be the vector of observed field data, $\mathbf{x} = (x_1, \ldots, x_N)^T$ the experimental settings under which the data were collected, and $\boldsymbol{\eta}(\boldsymbol{\theta}^{(\mathbf{x})}) = (\eta(x_1, \boldsymbol{\theta}(x_1)), \ldots, \eta(x_N, \boldsymbol{\theta}(x_N)))^T$ the calibrated computer output at these experimental settings. For ease of notation, we suppress the constraints in (2.3) so that any sample path restrictions are implied. Our proposed model is

$$\begin{aligned}
\mathbf{y}|\boldsymbol{\theta}^{(\mathbf{x})},\ & \lambda_y \sim N_N(\boldsymbol{\eta}(\boldsymbol{\theta}^{(\mathbf{x})}),\ \lambda_y^{-1}\mathbf{I}),\\
& \lambda_y \sim \mathrm{Ga}(a_y,\ b_y), \quad a_y, b_y > 0,\\
g(\theta_1(\cdot))|\lambda_\theta,\ & \rho_\theta \sim \mathcal{GP}(\mu_\theta,\ \lambda_\theta^{-1} R_{\rho_\theta}(\cdot,\cdot)), \quad -\infty < \mu_\theta < \infty,\\
& \theta_2 \sim \mathrm{Unif}(0, 1),\\
& \lambda_\theta \sim \mathrm{Ga}(a_\theta,\ b_\theta), \quad a_\theta, b_\theta > 0,\\
& \rho_\theta \sim \mathrm{Beta}(1,\ b_\theta), \quad b_\theta > 0,
\end{aligned} \tag{2.4}$$

where g is a known link function as in (2.2), $\boldsymbol{\theta}_1^{(\mathbf{x})} = (\theta_1(x_1), \ldots, \theta_1(x_N))^T$, and $R_{\rho_\theta}(\cdot,\cdot)$ is the correlation function given by $R_{\rho_\theta}(x,\ x') = \rho_\theta^{4(x-x')^2}$. With $\mathbf{g}(\boldsymbol{\theta}_1^{(\mathbf{x})}) = (g(\theta_1(x_1)), \ldots, g(\theta_1(x_N)))^T$, the joint posterior distribution is

$$\begin{aligned}
\pi(\boldsymbol{\theta}_1^{(\mathbf{x})}, \theta_2, \rho_\theta, \lambda_\theta, \lambda_y|\mathbf{y}) \propto \lambda_y^{N/2+a_y-1} \exp\left\{-\frac{\lambda_y}{2}(\mathbf{y}-\boldsymbol{\eta}(\boldsymbol{\theta}_1^{(\mathbf{x})}, \theta_2))^T(\mathbf{y}-\boldsymbol{\eta}(\boldsymbol{\theta}_1^{(\mathbf{x})}, \theta_2))\right\}\\
\times \exp(-b_y\lambda_y)\lambda_\theta^{N/2+a_\theta-1}|\mathbf{R}_{\rho_\theta}|^{-1/2}
\end{aligned}$$

$$\times \exp\left\{-\frac{\lambda_\theta}{2}(\mathbf{g}(\boldsymbol{\theta}_1^{(\mathbf{x})}) - \mu_\theta \mathbf{1})^T \mathbf{R}_{\rho_\theta}^{-1}(\mathbf{g}(\boldsymbol{\theta}_1^{(\mathbf{x})}) - \mu_\theta \mathbf{1})\right\}$$
$$\times \exp(-b_\theta \lambda_\theta)(1-\rho_\theta)^{b_\theta - 1},$$

where $\mathbf{R}_{\rho_\theta} = \{R_{\rho_\theta}(x_i, x_j)\}_{i,j=1}^N$ and $\mathbf{1} = (1, \ldots, 1)^T$.

We use Markov chain Monte Carlo (MCMC; Gelfand and Smith (1990)) to sample from the posterior. To eliminate the boundary constraints on θ_2 and ρ_θ, and to make our sampling algorithm less sensitive to the scale of the data, we reparameterize with $\xi = \log(-\log(\theta_2))$ and $\nu = \log(-\log(\rho_\theta))$. Here ν is equivalent to the correlation length parameterization suggested by Neal (1998) when implementing MCMC for models with GP priors. The subsequent full conditional distributions necessary for the algorithm are given in the Supplementary Material.

We use Gibbs sampling with Metropolis steps for the non-standard distributions (Metropolis et al. (1953); Geman and Geman (1984); Tierney (1994); Carlin and Louis (2009)). In drawing sample paths of $\theta_1(\cdot)$ with Metropolis proposals, we wish to take advantage of the prior smoothness assumptions. Following the suggestion of Neal (1998), we sample $\boldsymbol{\theta}_1^{(\mathbf{x})}$ using a multivariate Gaussian proposal with correlation matrix dependent upon the current value of ρ_θ. Thus, to sample from the distribution of $\boldsymbol{\theta}_1^{(\mathbf{x})}|\xi, \nu, \lambda_\theta, \lambda_y, \mathbf{y}$, on the k^{th} iteration, we find the spectral decomposition of $\mathbf{R}_\nu = \mathbf{U}\boldsymbol{\Lambda}\mathbf{U}^T$ and draw a proposal as $\boldsymbol{\theta}_1^{(\mathbf{x}),\dagger} = c\mathbf{U}\boldsymbol{\Lambda}^{1/2}\mathbf{z} + \boldsymbol{\theta}_1^{(\mathbf{x}),(k-1)}$, where $\mathbf{z} \sim N_N(\mathbf{0}, \mathbf{I})$ and c is determined adaptively during the burn-in period by monitoring the acceptance rate and adjusting periodically. We use the spectral decomposition of $\mathbf{R}_\nu$ despite the fact that it is slower to compute than the usual Cholesky decomposition, since it is more numerically stable for generating Gaussian random variables. For ξ, we use a Metropolis step with candidates $\xi^\dagger \sim N(\xi^{(k-1)}, c_\xi^2)$, where c_ξ is tuned adaptively, and similarly for ν.

When the observed design points are close together, the columns of the correlation matrix $\mathbf{R}_\nu$ are nearly linearly dependent so that $\mathbf{R}_\nu^{-1}$ is numerically unstable. While the spectral decomposition mitigates the problem when simulating multivariate Gaussian draws, this technique is not helpful in solving the matrix or finding its determinant. To address this, we add a nugget δ to obtain $\mathbf{R}_{\nu,\delta} := \mathbf{R}_\nu + \delta\mathbf{I}$. Ranjan, Haynes and Karsten (2011) proposed determining the nugget with $\delta = \max\{\lambda_N(\kappa(\mathbf{R}_\nu) - e^a)(\kappa(\mathbf{R}_\nu))^{-1}(e^a - 1)^{-1},\ 0\}$, where λ_N is the largest eigenvalue of $\mathbf{R}_\nu$, $\kappa(\mathbf{R}_\nu)$ is the condition number, and e^a is the threshold on $\kappa(\mathbf{R}_\nu)$ for the matrix to be well-conditioned. We use the

Cholesky factorization of the modified matrix, $\mathbf{R}_{\nu,\delta} = \mathbf{L}_\delta \mathbf{L}_\delta^T$, to approximate $\log(|\mathbf{R}_\nu|^{-1/2}) \approx -\sum_{i=1}^N \log(l_{ii})$, where l_{ii} is the i^{th} diagonal element of $\mathbf{L}_\delta$.

We obtain acceptable results using the above Metropolis-within-Gibbs sampling scheme, but we still find the smoothness hyperparameter ρ_θ difficult to estimate via posterior inference. Approaches to this problem suggested in the literature include Hamiltonian Monte Carlo (Neal (1998, 2011)), or substituting an empirical Bayes estimator such as the posterior mode (Qian and Wu (2008)). The identifiability problem and the dependence it can induce among calibration parameters in MCMC simulations have motivated useful advances such as delayed rejection adaptive Metropolis (Haario et al. (2006)). There is no doubt more research to be done in this area.

3. Simulation Study

To illustrate our proposed method, we simulated field data y_i, $i = 1, \ldots, N$, by supposing that $y(x_i) = c_1(x_i) + c_2 x_i^2 + \varepsilon_i$, where $\varepsilon_i \overset{\text{iid}}{\sim} N(0,\ 0.05^2)$, $i = 1, \ldots, N$. The computer model was $\eta(x, t_1, t_2) = t_1 + t_2 x^2$ so that both calibration parameters t_1 and t_2 were assumed constant in the computer code. We supposed that, in reality, $c_2 = 2.5$ is constant across the domain and that $c_1(\cdot)$ is determined by $c_1(x) = 2\sqrt{x}$. The field data were generated at $\mathbf{x} = (0.00, 0.05, 0.10, \ldots, 0.90, 0.95)^T$. The responses $\mathbf{y}^* = (y(0.45), \ldots, y(0.65))^T$ were held out as a validation set, leaving the remaining 15 observations as a training dataset. We used the logit link, $g(\theta(x)) = \log(\theta(x)/(1 - \theta(x)))$, and took $a_\theta = b_\theta = 0.01$ in the prior on λ_θ. We set $a_y = b_y = 5$, encouraging λ_y to be close to one since the data are standardized. For ρ_θ, we concentrated the prior near one with $b_\theta = 0.2$. The field data were standardized and the calibration parameters were scaled to lie in the unit hypercube prior to calibrating the computer model, $\theta_i = (c_i - c_{\min,i})(c_{\max,i} - c_{\min,i})^{-1}$, $i = 1, 2$, in (2.4).

To illustrate what is at stake, Figure 1 plots the simulated data along with the computer model predictions obtained when using the posterior means as the calibrated estimates inside the code. The left panel plots the estimated posterior means using both a constant assumption on θ_1 (with a uniform prior) as well as the functional assumption with the logit link. When treating both calibration parameters as constant, the calibrated code yields strong disagreement between the predictions and field data. In practice, this disagreement would likely be absorbed by adding an extra term to (2.1) to represent model discrepancy. Such an approach would conceal the true nature of the system, illustrated in the left

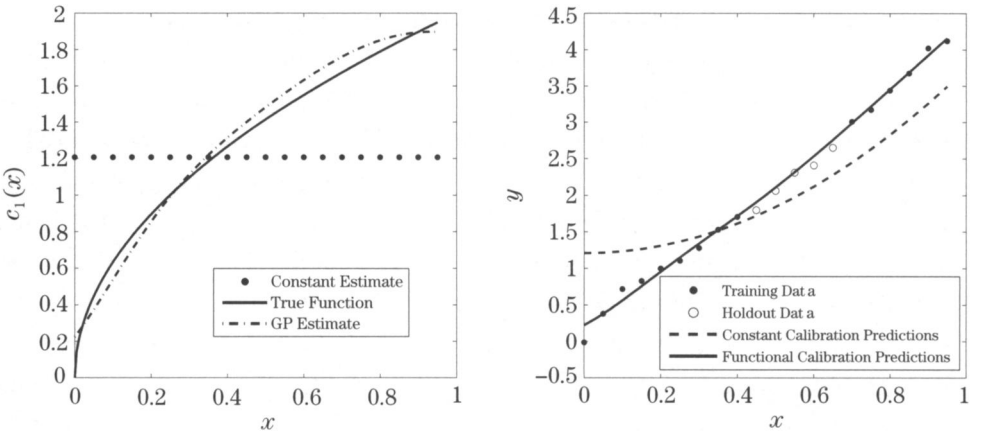

Figure 1. Simulated data used for calibration under the logit link along with posterior means. All values are plotted on the original scale. The right panel compares the corresponding mean predictions.

panel of Figure 1. By allowing $\theta_1(\cdot)$ to change over the experimental settings, reality is represented in a manner more consistent with experiments without resorting to a purely empirical discrepancy term. Our approach thus allows what would be a previously unknown functional form to emerge. We discuss these results further below. Notably, we show that treating θ_1 as constant still results in posterior concentration about the "average" value, so that a researcher could gain a false sense of security in their assumptions.

We simulated draws from the posterior via MCMC as described in Section 2. For each of the scenarios considered, we ran three chains in parallel using different starting values to assess convergence. Each chain used a burn-in period of 5,000 iterations, after which the chains were run for an additional 4,000 sampling iterations. Each chain was thinned to reduce autocorrelation. Trace plots were examined to assess convergence, after which the draws for the three chains were combined.

We *a priori* enforced the constraints $-0.075 \leq c_1(x_1) \leq 0.075$ and $1.85 \leq c_1(x_{20}) \leq 2.05$ so that the range $c_1(x_1)$ had a width of 3 error standard deviations and the range of $c_1(x_{20})$ had a width of 4 error standard deviations. By contrast, we took the prior on c_2 to be uniform between $c_{2,\min} = 1$ and $c_{2,\max} = 3$, so that it measured 40 error standard deviations in width. Figure 2 illustrates the results. In the left panel we see a strong contrast between the prior and posterior densities of c_2, demonstrating the considerable Bayesian learning about this parameter that occurred. Further, the posterior is correctly concentrating about $c_2 = 2.5$.

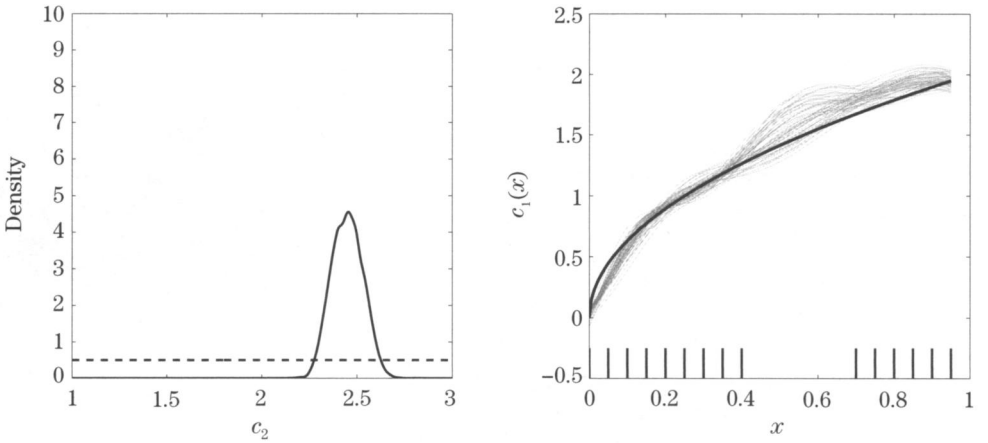

Figure 2. Posteriors of $c_1(\cdot)$ and c_2 under the logit link with constraints on the boundary values of $c_1(\cdot)$. The dashed and solid curves in the left panel are the prior and posterior densities of c_2, respectively. The thick line in the right panel is the true function $c_1(\cdot)$. The heavy tick marks at the bottom indicate the x values of the training data.

The right panel plots sample paths from the distribution of $c_1(\cdot)|\mathbf{y}$. Here we see our model's ability to recover the true functional dependence on x, despite c_1 being treated as constant inside the computer code. The posterior draws tend to closely agree with the truth at each of the holdout points.

For the second scenario, we placed more informative prior bounds on c_2 so that it was uniform between $c_{2,\min} = 2.35$ and $c_{2,\max} = 2.65$. We removed the constraints on the values of $c_1(x)$ at any x so that the possible realizations were unrestricted. Figure 3 illustrates the prior and posterior of c_2 and posterior sample paths of $c_1(\cdot)$. We see the weak Bayesian learning about c_2 that has occurred in this case. This reflects the fact that, given the bounds we have already imposed on the possible values for c_2, the data contain little additional information concerning plausible values. We again see posterior concentration of $c_1(\cdot)$ about the true parameter path at both the observed design points as well as at the untested design settings. Despite allowing for unconstrained functional paths, we were able to recover the functional form.

Suppose we know that $c_1(\cdot)$ can be approximated with $c_1(x) = \beta_0^U + \beta_1^U\sqrt{x}$, where β_0^U and β_1^U are unknown. In this case, we altered Model (2.4) by writing $\boldsymbol{\theta}_1^{(\mathbf{x})} = (\beta_0 + \beta_1\sqrt{x_1}, \ldots, \beta_0 + \beta_1\sqrt{x_N})^T$ and assigning prior distributions to β_0 and β_1, where we drop the superscripts to indicate the rescaling. Calibration then involved determining probable values of $\boldsymbol{\beta} = (\beta_0, \beta_1)^T$. To give the data as much freedom as possible in determining appropriate values, we used a flat

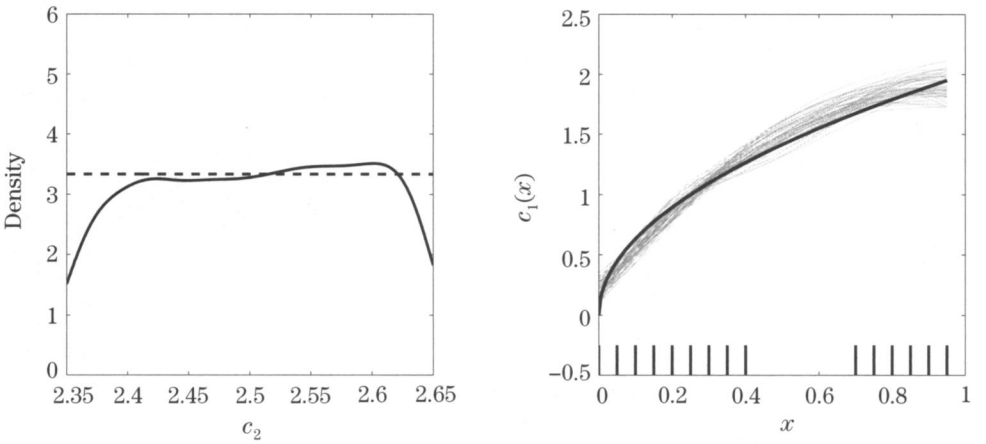

Figure 3. Posteriors of $c_1(\cdot)$ and c_2 with logit link and tight prior bounds on c_2. The dashed and solid curves in the left panel are the prior and posterior densities of c_2, respectively. The thick line in the right panel is the true function $c_1(x)$. The heavy tick marks at the bottom indicate the x values of the training data.

prior, $\pi(\boldsymbol{\beta}) \propto 1$. We used the same prior distribution for λ_y as in the previous simulations and again took c_2 to be *a priori* uniform between $c_{2,\min} = 2.35$ and $c_{2,\max} = 2.65$. We simulated the posterior distribution with the same burn-in period and the same number of chains as with the GP model.

Supplementary Figure 1 displays the smoothed approximate posterior densities of c_2, β_0^U, and β_1^U. We see that $\beta_0^U = 0$ and $\beta_1^U = 2$ are contained in the high density regions of their respective posteriors. Thus, we recover the true functional relationship $c_2(x) = 2\sqrt{x}$ with high probability, as evident in the far right panel of the Figure.

Another situation we considered is the conventional approach in which all calibration parameters were assigned flat prior distributions over ranges determined from, e.g., expert opinion. In this case, we took $\pi(c_2) \propto I(2.35 < c_2 < 2.65)$ and $\pi(c_1) \propto I(-0.5 < c_1 < 2.5)$. For the MCMC implementation with the rescaled calibration parameters, we reparameterized the joint posterior in terms of $\log(-\log(\theta_1))$ just as we did with θ_2 to eliminate boundary constraints and facilitate Gaussian proposals for Metropolis sampling.

Figure 4 presents the smoothed approximate posterior densities for c_1 and c_2 resulting from treating both as constant throughout the domain of applicability. Similar to the previous models with informative bounds on c_2, we see little additional Bayesian learning about c_2. In spite of the simplistic treatment of c_1, we see considerable posterior concentration around 1.25. This belies the fact that c_1

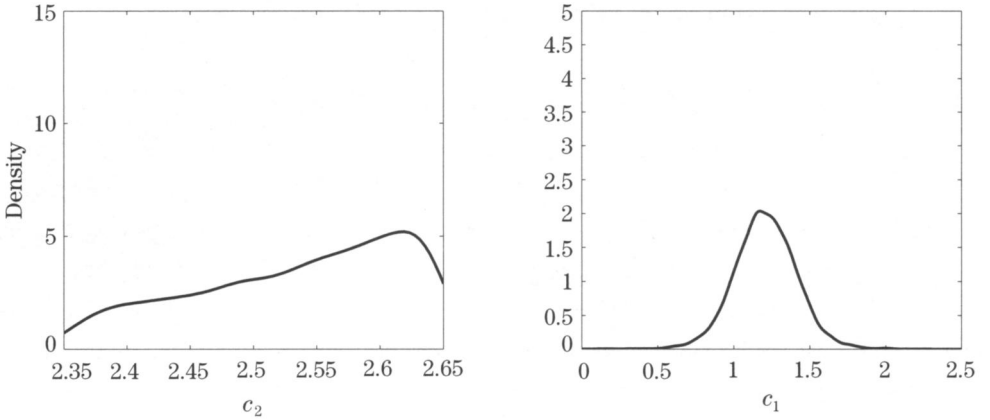

Figure 4. Smoothed approximate posterior distributions of c_2 (left panel) and c_1 (right panel) when replacing the GP prior on $\theta_1(\cdot)$ with $\theta_1 \sim$ Uniform in Model (2.4).

is truly state-dependent. Hence, strongly identified parameters are no guarantee that the assumed model is the best a researcher can do in describing the system of interest. This could be misleading to the practitioner, who might instead rely on an empirical model discrepancy term to correct the prediction errors seen in Figure 1.

Supplementary Figure 2 plots posterior predictions and approximate 95% error bars about the holdout design settings for each of the models considered above. While each model is capturing the true responses within its prediction tolerance, an obvious difference between them is in the associated uncertainties. As expected, the model assuming the correct functional form for $\theta_1(x)$ results in the best predictions. We see, however, that the more flexible GP model still yields competitive predictions. Little is lost by relaxing the assumption of a specific parametric function. Note the loss of predictive certainty from treating θ_1 as constant. The root mean squared predictive errors (RMSPE) are displayed in Table 1, which indicate that all three models assuming functional dependence vastly outperform the model treating both calibration parameters as constant.

The most common link functions for unit interval-valued data are the logit, probit, and cumulative log-log functions. When the values can be assumed to be away from the boundaries with high probability, the identity link also can be used. Supplementary Figure 3 compares posterior sample paths obtained from our simulated data using each of these link functions with unconstrained sample paths and informative prior bounds on c_2. We see that all of them are competitive in terms of recovering the true functional relationship. The differences arise from

Table 1. Root mean squared predictive error (RMSPE) of the posterior predictions at the holdout settings.

Model	Parametric $\theta_1(\cdot)$	Constrained $\theta_1(x_1)$, $\theta_1(x_N)$	Informative $\pi(\theta_2)$	Constant θ_1
RMSPE	0.0538	0.1185	0.0902	0.2783

each link function's effect on the Bayesian learning in the posterior and hence the convergence of the MCMC algorithm. When the Gaussian approximation is justified, faithful posterior estimates of the function are obtained. This approximation also results in the smallest out of sample prediction error, as evident in Supplementary Table 1, which shows the RMSPE for each of the considered link functions.

To demonstrate what can go wrong, suppose in our example that both c_2 and $c_1(\cdot)$ are given vague priors. The danger here is that weak identifiability might result in highly correlated parameters in the sampling algorithm, leading to convergence difficulties. Supplementary Figures 4 and 5 display trace plots of the sampled values of c_2, $c_1(x_{10})$, and $c_1(x_{15})$ from three different chains using different initial values along with sample paths of $c_1(\cdot)$ obtained from these chains when using vague priors. The chains do not mix well so that posterior inference is unreliable. Weak identifiability is a concern when resource-intensive collection of field data limits the available sample size. In this case, using available prior information is crucial.

Simulation results demonstrate that our proposed model can calibrate computer codes and adequately capture unknown functional behavior of the calibration parameters. We see that eliciting such prior information about the parameters can mitigate identifiability problems that are ubiquitous in model validation. Our results suggest the counterintuitive fact that allowing the calibration parameter θ_1 to vary across the experimental domain results in much less uncertainty about future predictions, in spite of the strong Bayesian learning that occurs when treating θ_1 as constant. This behavior is particularly appealing since the reduction in uncertainty occurred regardless of whether we imposed the correct functional form or assigned $\theta_1(\cdot)$ a nonparametric Gaussian process prior. We illustrate further that similar results can be obtained under a variety of link functions. We emphasize, however, that our experience suggests that the identity link approximation is best when the true values can be safely assumed to be far from the boundaries.

4. Application to VPSC Material Plastic Deformation

As an application, we consider a viscoplastic self-consistent material (VPSC) model for the plastic deformation of polycrystals. This model, developed by Lebensohn and Tomeé (1993) and studied by Atamturktur et al. (2015), treats a polycrystal as a set of single crystals with a texture represented by crystollographic orientations that evolve during plastic deformation. Relationships between deviatoric stress and strain-rate tensors are used to model this viscoplastic deformation. The VPSC formulation imposes a strain-rate during each incremental deformation step, resulting in stress-strain curves as part of the output of the model. The so-called glide-only version of the VPSC model allows dislocations of single crystals to move within the slip plane and hence describes simple shear deformations on this plane. The strain rate at the level of a single crystal, $\dot{\varepsilon}$, is approximated by

$$\dot{\varepsilon} = \dot{\gamma}_0 \sum_{s=1}^{N_s} m^s \left(\frac{|m^s : \sigma|}{\tau_0} \right)^{n_g} \operatorname{sign}(m^s : \sigma), \tag{4.1}$$

where σ is the applied stress, m^s is the Schmid tensor, τ_0 is the critical resolved shear stress associated with glide, n_g is the inverse rate sensitivity for the glide activity, N_s is the total number of active slip systems, $\dot{\gamma}_0$ is a normalizing constant, and : denotes the tensor product.

Stout et al. (1998a,b) reported experiments concerning the plastic deformation of 5182 aluminum to which the glide VPSC model is applicable. Two inputs, temperature and strain-rate, were varied in the experiments and stress-strain curves subsequently measured. The experiments were performed until each specimen attained a strain of 0.6, at which time the corresponding stress of the specimen was recorded. Eleven experiments were originally conducted at temperature settings between 200 and 550 °C and strain-rate equal to 10^{-3} and 1. In the VPSC computer code for implementing (4.1), the glide stress exponent n_g and the critical resolved shear stress τ_0 are to be calibrated against the experimental data. Previous empirical work suggests that τ_0 is a function of temperature. We thus incorporated this functional dependence into calibrating Model (4.1). A parametric functional form was used for $\tau_0(\cdot)$ in Atamturktur et al. (2015). This model is purely empirical in the absence of any existing theory. Hence, we relaxed the parametric assumption and used a Gaussian process model for $\tau_0(\cdot)$. We used as our field data the experiments conducted at strain-rate equal to 10^{-3} while varying temperature. The experimental data are given in Table 2.

We relied on expert opinion and previous empirical work to determine ranges

Table 2. Experimental results of plastic deformation of 5182 aluminum (Stout et al. (1998a,b)).

Experiment	A	B	C	D	E	F
Temperature (°C)	200	300	350	400	500	550
Maximum Stress (MPa)	226.2	91.4	50.0	30.6	14.9	7.0

Table 3. Bounds on control and calibration parameters for the VPSC application.

Parameter	Temperature (°C)	n_g	τ_0 (MPa)	$\tau_0(x_1)$	$\tau_0(x_N)$
Range	[180.00, 570.00]	[2.50, 4.50]	[1.20, 1343.40]	[519.03, 693.07]	[7.78, 42.15]

for n_g and τ_0. We also had available the extrema for the control input, temperature. These bounds, displayed in Table 3, are used to scale the parameters to lie in the unit hypercube prior to calibration. Atamturktur et al. (2015) used nonlinear constrained optimization to obtain optimal values for τ_0 at different temperature settings for use in estimating a parametric function for $\tau_0(\cdot)$. We used this information to refine the constraints on $\tau_0(\cdot)$ at the boundaries of the experimental domain. These values are given in Table 3, as well.

Figure 5 displays the prior and smoothed approximate posterior distributions of n_g along with sample paths drawn from the approximate posterior distribution of $\tau_0(\cdot)$ using the identity link approximation. Superimposed on the sample paths are the pointwise mean curve and the constraints on the boundary values of the paths. For reference, experimental temperature settings used in the calibration are denoted with the large tick marks along the x-axis. The density about n_g has updated to become slightly more concentrated about 3.5, in agreement with previous empirical work. The boundary constraints on $\tau_0(\cdot)$ are obviously influential in determining posterior sample paths, as we would expect given the limited experimental data available.

As a check of model adequacy, we examined the distributions of selected test quantities of interest, $p(T(\mathbf{y}^*)|\mathbf{y}) = \int p(T(\mathbf{y}^*)|\boldsymbol{\theta})\pi(\boldsymbol{\theta}|\mathbf{y})d\boldsymbol{\theta}$, where $\mathbf{y}^*$ is a posterior replication of the dataset. The test quantities we used were the sample mean, $T_1(\mathbf{y}) = N^{-1}\sum_{i=1}^{N} y_i$, the sample variance, $T_2(\mathbf{y}) = (N-1)^{-1}\sum_{i=1}^{N}(y_i - \overline{y})^2$, and the sample inner product $T_3(\mathbf{y}) = \sum_{i=1}^{N} x_i y_i$. These are sufficient statistics for a linear regression of $\mathbf{y}$ on $\mathbf{x}$ and thus summarize salient features of the data. The distributions of these test quantities also enabled us to approximate the *Bayesian p-values* (Gelman et al. (2014, chap. 6)), $p_B^{(i)} = P(T_i(\mathbf{y}^*) \geq T_i(\mathbf{y})|\mathbf{y}) = \int\int I[T_i(\mathbf{y}^*) \geq T_i(\mathbf{y})]p(\mathbf{y}^*|\boldsymbol{\theta})\pi(\boldsymbol{\theta}|\mathbf{y})d\mathbf{y}^* d\boldsymbol{\theta}$, $i = 1, 2, 3$. The Bayesian p-value is a simple measure of discrepancy between a hypothesized model and observed data,

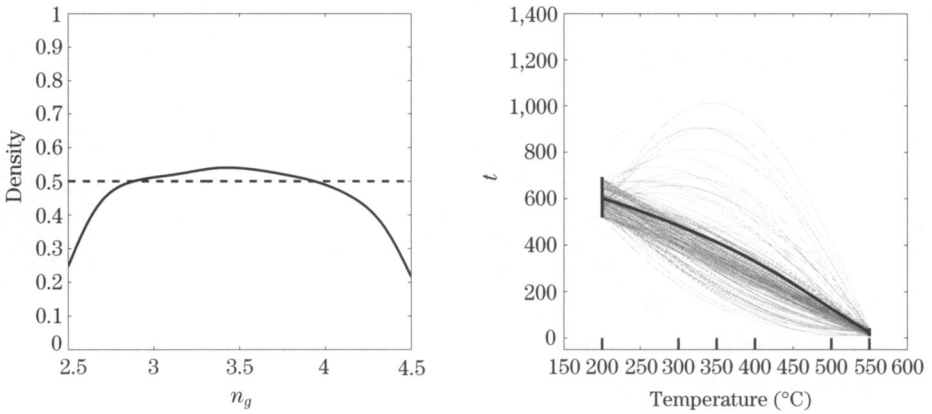

Figure 5. Smoothed prior and posterior histogram for n_g (left panel) and sample paths drawn from the posterior of $\tau_0(\cdot)$ (right panel). The dark curve in the center is the pointwise mean of the sample paths; the dark vertical lines on the boundaries indicate the prior constraints imposed on the curves. The large tick marks along the x-axis denote the experimental temperature settings used for the calibration.

with values close to zero or one indicating a model's failure to explain features of the data. Supplementary Figure 6 displays histograms of realizations of T_1, T_2, and T_3 based on 2,000 replications drawn from $p(\mathbf{y}^*|\mathbf{y})$. In each plot, the dark vertical line represents the observed value of the statistic from the experimental data. In all three cases, the observed value is well within the range of plausible values posited by our model. The Bayesian p-values for each statistic were $p_B^{(1)} = 0.831$, $p_B^{(2)} = 0.616$, and $p_B^{(3)} = 0.785$. Supplementary Figure 7 displays the posterior predictions with approximate 95% error bars at the observed temperature settings, where we see that the field data are well within the bounds predicted by our model. We can conclude that our modeling assumptions and the subsequent calibrations are consistent with the experimental data.

This application illustrates our model's ability to adapt to changes in appropriate calibration values as a function of the experimental settings while treating other calibration parameters constant. The example also illustrates how the additional uncertainty introduced by omitting the assumption of a parametric functional form is incorporated into model predictions. In the presence of this uncertainty, we still obtain calibrated model predictions that are consistent with field data.

5. Discussion

Standard practice in computer model calibration is to use elicited prior in-

formation to construct relatively simple prior distributions on the calibration parameters and treat them as constant throughout the domain of applicability. While this methodology has proven to be effective, the situation can be improved by acknowledging the fact that the calibrated values might vary as a function of the control inputs and modeling this phenomenon appropriately. Indeed, when models are simplified, the dependence of parameters on the state of the system can be lost. The proposed nonparametric functional model presented here makes the calibration "state-aware" through a Gaussian process on the parameters thought to change over the domain.

Through simulation and application, we show that the posterior distribution of our proposed model effectively incorporates prior information and fully accounts for the remaining uncertainty in the presence of small sample sizes while still yielding predictions consistent with experimental observations. We demonstrate that knowing the correct functional form *a priori* yields the best predictions with the most precision. However, we are able to obtain competitive predictive performance even after relaxing the parametric assumption in favor of a nonparametric model.

Our results suggest that the constant parameter assumption could be misleading in that the posterior distribution may still concentrate around particular calibration parameter values despite this assumption being incorrect. In this case, a researcher might opt for a purely empirical model discrepancy term to account for the differences between the calibrated predictions and the field data. Such an approach works well when prediction is the only goal of the calibration procedure. Often, however, inferences about the calibration parameters are desired in addition to reliable prediction of future outcomes. In this case, the presence of a discrepancy term exacerbates identifiability problems that are already present (Bayarri et al. (2007)). Our proposed approach can reveal when a discrepancy term is unnecessary, facilitating stronger inferences while increasing a researcher's confidence in using their model for extrapolation.

Small sample sizes are the norm rather than the exception in computer model calibration, so identifiability is of utmost concern. This paper illustrates that unconstrained functional calibration with vague priors limits posterior inference. We demonstrate the utility of incorporating prior information which is often available from subject matter experts. There has been considerable discussion, though, about interpreting the calibration parameters in certain applications and whether such interpretations even admit the possibility of expert-elicited prior distributions (Kennedy and O'Hagan (2001)). However, uncertainties from sources other

than existing expert opinion can be used to construct prior distributions, as well (Bayarri et al. (2007)). Thus nonparametric functional calibration is still feasible. We demonstrate how it produces reliable inference and predictions while fully accounting for the uncertainty about the functional form. In addition to boundary constraints, it also might be possible to incorporate prior information such as known monotonicity to further improve identifiability (Golchi et al. (2015)).

Our proposed model assumes fast-running computer code, circumventing the need for a surrogate model. It is common in practice, though, for the computer code to be computationally expensive. Indeed, while we are able to obtain the results in Section 4 without an emulator for the VPSC model, the code does in fact take a couple of seconds to execute a single run, making the MCMC routine slow. A natural extension that will be explored in future work is the replacement of the actual computer code in (2.4) with a surrogate model. As the dimension of the parameter space increases in a computer model, however, the sensitivities and parameter correlations are much easier to understand when a GP emulator is avoided (Hemez and Atamturktur (2011)). We thus recommend using the computer model directly if at all feasible, but acknowledge that the extension of our proposed method to include an emulator is needed.

There remains the question of deciding when to invoke our so-called state-aware calibration, as it may not always be obvious which parameters to treat as functional and which to treat as constant. We suggest beginning with the conventional calibration approach in which all the calibration parameters are treated as constant. The presence of systematic model bias can point to the need for incorporating functional relationships into the calibration. If it is not obvious which parameters might follow a functional relationship, then a sensitivity analysis can be performed, after which the most influential parameters would naturally be the first ones assigned a functional model. Through this approach, a researcher may gain an idea of which parameters to treat as functionally related to the control inputs, but might not know the functional form. At this point, a nonparametric Gaussian process model can be fit to the functional calibration parameters, which then may suggest a specific parametric functional form. Both the parametric and nonparametric versions of the model can be fit and compared using a model assessment tool such as the deviance information criterion (DIC; Carlin and Louis (2009)). If it is found suitable, the parametric model is to be preferred, since it can improve extrapolation and, more importantly, suggest missing physics in the system. State-aware calibration, then, can be a valuable tool for determining when to expand on a currently accepted physics model by

revealing previously unknown functional relationships. When found to be consistent with experimentation, suitable parametric functions suggested by the initial nonparametric model will help researchers fill gaps in scientific knowledge.

Supplementary Materials

The online Supplementary Material includes the full conditional distributions necessary for Gibbs sampling, additional figures referred to in the text, and our `MATLAB` implementation of the MCMC algorithm discussed in this paper.

Acknowledgment

The authors are grateful to Garrison Stevens for her assistance with the `MATLAB` code, to Brian Williams and Cetin Unal of Los Alamos National Laboratory for helpful comments, and to Ricardo Lebensohn of Los Alamos National Laboratory for access to the VPSC code. We thank the editors and referees for their suggestions toward improving this paper, especially for bringing to our attention the recent work of Plumlee, Joseph and Yang (2016).

References

Atamturktur, S. and Brown, D. A. (2015). State-aware calibration for inferring systematic bias in computer models of complex systems. *NAFEMS World Congress Proceedings*, San Diego, CA.

Atamturktur, S., Hegenderfer, J., Williams, B., Egeberg, M., Lebensohn, R. A. and Unal, C. (2015). A resource allocation framework for experiment-based validation of numerical models. *Mech. Adv. Mater. Struc.* **22**, 641-654.

Bayarri, M. J., Berger, J. O., Paulo, R., Sacks, J., Cafeo, J. A., Cavendish, J., Lin, C.-H. and Tu, J. (2007). A framework for validation of computer models. *Technometrics* **49**, 138-154.

Berman, A. and Nagy, E. J. (1983). Improvement of a large analytical model using test data. *AIAA J.* **21**, 1168-1173.

Carlin, B. P. and Louis, T. A. (2009). *Bayesian Methods for Data Analysis.* 3rd ed. Chapman & Hall/CRC, Boca Raton.

Craig, P. S., Goldstein, M., Rougier, J. C. and Seheult, A. H. (2001). Bayesian forecasting for complex systems using computer simulators. *J. Am. Stat. Assoc.* **96**, 717-729.

Fugate, M., Williams, B., Higdon, D., Hanson, K. M., Gattiker, J., Chen, S.-R. and Unal, C. (2006). Hierarchical Bayesian analysis and the Preston-Tonks-Wallace model. Technical Report LA-UR-06-5205, Los Alamos National Laboratory.

Gelfand, A. E. and Sahu, S. K. (1999). Identifiability, improper priors, and Gibbs sampling for generalized linear models. *J. Am. Stat. Assoc.* **94**, 247-253.

Gelfand, A. E. and Smith, A. F. M. (1990). Sampling-based approaches to calculating marginal densities. *J. Am. Stat. Assoc.* **85**, 398-409.

Gelman, A., Carlin, J. B., Stern, H. S., Dunson, D. B., Vehtari, A. and Rubin, D. B. (2014).

Bayesian Data Analysis. 3rd ed. Chapman & Hall/CRC, Boca Raton.

Geman, S. and Geman, D. (1984). Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. *IEEE T. Pattern Anal.*, **6**, 721-741.

Golchi, S., Bingham, D. R., Chipman, H. and Campbell, D. A. (2015). Monotone emulation of computer experiments. *SIAM/ASA J. Uncertainty Quantification* **3**, 370-392.

Gustafson, P. (2005). On model expansion, model contraction, identifiability, and prior information: Two illustrative scenarios involving mismeasured variables. *Stat. Sci.* **20**, 111-140.

Haario, H., Laine, M., Mira, A. and Saksman, E. (2006). DRAM: Efficient adaptive MCMC. *Stat. Comput.* **16**, 339-354.

Han, G., Santner, T. J. and Rawlinson, J. J. (2009). Simultaneous determination of tuning and calibration parameters for computer experiments. *Technometrics* **51**, 464-474.

Hemez, F. M. and Atamturktur, S. (2011). The dangers of sparse sampling for the quantification of margin and uncertainty. *Reliab. Eng. Syst. Safe.* **96**, 1220-1231.

Higdon, D., Gattiker, J., Williams, B. and Rightley, M. (2008). Computer model calibration using high-dimensional output. *J. Am. Stat. Assoc.* **103**, 570-583.

Higdon, D., Kennedy, M., Cavendish, J. C., Cafeo, J. and Ryne, R. D. (2004). Combining field data and computer simulations for calibration and prediction. *SIAM J. Sci. Comput.* **26**, 448-466.

Joseph, V. R. and Melkote, S. N. (2009). Statistical adjustments to engineering models. *J. Qual. Technol.* **41**, 362-375.

Kennedy, M. C., Anderson, C. W., Conti, S. and O'Hagan, A. (2006). Case studies in Gaussian process modeling of computer codes. *Reliab. Eng. Syst. Safe.* **91**, 1301-1309.

Kennedy, M. C. and O'Hagan, A. (2001). Bayesian calibration of computer models. *J. R. Stat. Soc. B* **63**, 425-464.

Lebensohn, R. A. and Tomé, C. N. (1993). A self-consistent anisotropic approach for the simulation of plastic deformation and texture development of polycrystals: Application to zirconium alloys. *Acta Metall. Mater.* **41**, 2611-2623.

Loeppky, J. L., Bingham, D. and Welch, W. J. (2006). Computer model calibration or tuning in practice. Unpublished manuscript.

McCullagh, P. and Nelder, J. A. (1989). *Generalized Linear Models.* 2nd ed. Chapman & Hall, London.

Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H. and Teller, E. (1953). Equation of state calculations by fast computing machines. *J. Chem. Phys.* **21**, 1087-1091.

Neal, R. M. (1998). Regression and classification using Gaussian process priors. In *Bayesian Statistics 6*, eds. Bernardo, J. M., Berger, J. O., Dawid, A. P., and Smith, A. F. M. Oxford University Press, New York.

Neal, R. M. (2011). MCMC using Hamiltonian dynamics. In *Handbook of Markov Chain Monte Carlo*, eds. Brooks, S., Gelman, A., Jones, G. L., and Meng, X.-L. Chapman & Hall/CRC, Boca Raton.

O'Hagan, A. (1978). Curve fitting and optimal design for prediction. *J. R. Stat. Soc. B* **40**, 1-42.

Park, J. S. (1991). Tuning complex computer codes to data and optimal designs. PhD Thesis, University of Illinois, Champaign/Urbana, IL, USA.

Plumlee, M., Joseph, V. R. and Yang, H. (2016). Calibrating functional parameters in the ion channel models of cardiac cells. *J. Am. Stat. Assoc.* To appear.

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 Trans.* **47**, 141-152.

Pratola, M. T., Sain, S. R., Bingham, D., Wiltberger, M. and Rigler, E. J. (2013). Fast sequential computer model calibration of large nonstationary spatial-temporal processes. *Technometrics* **55**, 232-242.

Qian, P. Z. G. and Wu, C. F. J. (2008). Bayesian hierarchical modeling for integrating low-accuracy and high-accuracy experiments. *Technometrics* **50**, 192-204.

Ranjan, P., Haynes, R. and Karsten, R. (2011). A computationally stable approach to Gaussian process interpolation of deterministic computer simulation data. *Technometrics* **53**, 366-378.

Reese, C. S., Wilson, A. G., Hamada, M., Martz, H. F. and Ryan, K. J. (2004). Integrated analysis of computer and physical experiments. *Technometrics* **46**, 153-164.

Robert, C. P. (1995). Simulation of truncated normal random variables. *Stat. Comput.* **5**, 121-125.

Santner, T. J., Williams, B. J. and Notz, W. I. (2003). *The Design and Analysis of Computer Experiments.* Springer-Verlag, New York.

Stout, M., Chen, S. R., Kocks, U. F., Schwartz, A. J., MacEwan, S. R. and Beaudoin, A. J. (1998a). Constitutive modeling of 5182 aluminum as a function of strain rate and temperature. In *Hot Deformation of Aluminum Alloys II*, eds. Bieler, T., Lalli, T. A., and MacEwan, L. A. TMS, Warrendale.

Stout, M., Chen, S. R., Kocks, U. F., Schwartz, A. J., MacEwan, S. R. and Beaudoin, A. J. (1998b). Mechanisms responsible for texture development in a 5182 aluminum alloy deformed at elevated temperature. In *Hot Deformation of Aluminum Alloys II*, eds. Bieler, T., Lalli, T. A., and MacEwan, L. A. TMS, Warrendale.

Tierney, L. (1994). Markov chains for exploring posterior distributions. *Ann. Stat.* **22**, 1701-1762.

Tuo, R. and Wu, C. F. J. (2015). Efficient calibration for imperfect computer models. *Ann. Stat.* **43**, 2331-2352.

Tuo, R. and Wu, C. F. J. (2016). A theoretical framework for calibration in computer models: Parameterization, estimation and convergence properties. *SIAM/ASA J. Uncertainty Quantification.* To appear.

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**, 765-792.

Xiong, Y., Chen, W., Tsui, K.-L. and Apley, D. W. (2009). A better understanding of model updating strategies in validating engineering models. *Comput. Method. Appl. M.* **198**, 1327-1337.

Department of Mathematical Sciences, Clemson University, Clemson, SC 29634, U.S.A.

E-mail: ab7@clemson.edu

Glenn Department of Civil Engineering, Clemson University, Clemson, SC 29634, U.S.A.

E-mail: sez@clemson.edu

(Received October 2015; accepted May 2016)