

Elastic Bayesian Model Calibration*

Devin Francom[†], J. Derek Tucker[‡], Gabriel Huerta[‡], Kurtis Shuler[‡], and Daniel Ries[‡]

Abstract. Functional data are ubiquitous in scientific modeling. For instance, quantities of interest are modeled as functions of time, space, energy, density, etc. Uncertainty quantification methods for computer models with functional response have resulted in tools for emulation, sensitivity analysis, and calibration that are widely used. However, many of these tools do not perform well when the computer model's parameters control both the amplitude variation of the functional output and its alignment (or phase variation). This paper introduces a framework for Bayesian model calibration when the model responses are misaligned functional data. The approach generates two types of data out of the misaligned functional responses: (1) aligned functions so that the amplitude variation is isolated and (2) warping functions that isolate the phase variation. These two types of data are created for the computer simulation data (both of which may be emulated) and the experimental data. The calibration approach uses both types so that it seeks to match both the amplitude and phase of the experimental data. The framework is careful to respect constraints that arise, especially when modeling phase variation, and is framed in a way that it can be done with readily available calibration software. We demonstrate the techniques on two simulated data examples and on two dynamic material science problems: a strength model calibration using flyer plate experiments and an equation of state model calibration using experiments performed on the Sandia National Laboratories' Z-machine.

Key words. amplitude/phase variability, Bayesian model calibration, functional data analysis, material strength calibration

MSC codes. 62P35, 62J02

DOI. 10.1137/24M1644092

1. Introduction. In domains of science and engineering where modeling is an important part of investigation and discovery, quantifying uncertainty in model inferences and predictions can be essential in order for model performance to be trusted. When a model has uncertain parameters (or inputs), model calibration is the act of tuning the parameters so that the model produces a desired response. Most often, models are calibrated to experimental or observational measurements, so that calibration seeks to make the model response reflect

*Received by the editors March 5, 2024; accepted for publication September 20, 2024; published electronically February 18, 2025.

<https://doi.org/10.1137/24M1644092>

Funding: This paper describes objective technical results and analysis. Any subjective views or opinions that might be expressed in the paper do not necessarily represent the views of the U.S. Department of Energy or the United States Government. This work was supported by the NA-22 program, the Advanced Simulation and Computing program, and Laboratory Directed Research and Development program at Los Alamos National Laboratory and Sandia National Laboratories. Sandia National Laboratories is a multimission laboratory managed and operated by National Technology and Engineering Solutions of Sandia, LLC, a wholly owned subsidiary of Honeywell International, Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525.

[†]Los Alamos National Laboratory, Los Alamos, NM 87545 USA (dfrancom@lanl.gov).

[‡]Sandia National Laboratories, Albuquerque, NM 87015 USA (jdtuck@sandia.gov, jghuert@sandia.gov, kwshule@sandia.gov, dries@sandia.gov).

reality. Model calibration (sometimes known as inversion) is often a poorly identified problem, where multiple combinations of inputs produce equally valid solutions. Kennedy and O'Hagan (2001) proposed Bayesian model calibration as a systematic approach to calibrating a model in the face of all of the sources of uncertainty so that calibration uncertainty is quantified. These sources of uncertainty are parameter uncertainty, measurement uncertainty, emulation or surrogate model uncertainty (an emulator is a fast surrogate for a more expensive model), and model form error (i.e., simulation model misspecification, discrepancy, or bias).

Often, computer model outputs are functional in nature, producing an output measured over space and/or time, for example. The majority of calibration work on these types of outputs has been done on features extracted from the outputs such as peak points or critical values (Walters et al., 2018). However, the process for extracting these features can be tedious and prone to error and is problem specific. Other calibration approaches have been developed for use with multivariate or functional response (Bayarri et al., 2007a; Francom et al., 2019; Higdon et al., 2008), although they are prone to problems when presented with functional response data that are misaligned. Our goal is to extend functional emulation and calibration methods for use when the response is misaligned, and to do so without human-intensive feature engineering.

There has been considerable effort in statistics to develop methods that can analyze functional data objects without loss of information. Such methodology is known as functional data analysis and has a rich history. An excellent introduction to this field is given in several books, including Ramsay and Silverman (2005), Horvath and Kokoszka (2012), and Srivastava and Klassen (2016). An interesting aspect of most functional data is that the underlying variability can be ascribed to two sources. These two sources are termed the amplitude (y or vertical) variability and the phase (x or horizontal or warping) variability. Capturing these two sources of variability is crucial when modeling functional data and can greatly affect the construction of statistics (e.g., averages, tolerance bounds). In this work, we refer to methods that handle both amplitude and phase variability in functional data as *elastic*. This important concept is illustrated in Figure 1 through a simulated example.

In Figure 1, we have two functions that contain both a peak and a trough. Each of the functions contains variability in the height of the peak and valley and large variability in its placement. The relative heights of the peaks can be attributed to the amplitude variability, while the different locations of the peaks constitute the phase variability.

The phase variability can be accounted for by first aligning the functions. As an example, the right panel shows time-aligned functions. The alignment involves a transformation of the horizontal axis via the warping function shown in the middle panel. The aligned function ($f_1(\gamma(t))$) captures the amplitude variability while the warping function (γ) captures the phase variability. The Bayesian model calibration method introduced in this paper considers the shape of the data by accounting for both directions of variability.

While standard calibration solutions can be applied to misaligned functions, especially when emulation is unnecessary, we demonstrate that using functional metrics that consider the misaligned nature of the data can produce more accurate calibration solutions and more practical model formulations. In our methodology, we use elastic functional data analysis methods (Marron et al., 2015; Srivastava and Klassen, 2016; Tucker et al., 2013) to construct a metric to measure the distance between functions and to specify a likelihood. We do this by

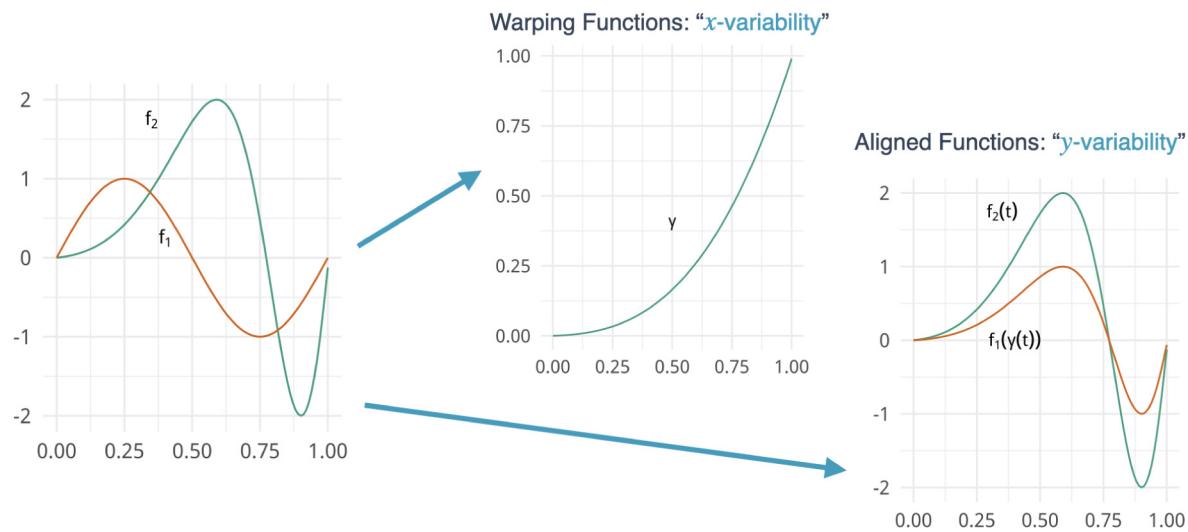


Figure 1. Demonstration of amplitude and phase variability in functional data. Left: original functions. Middle: warping function (“phase variability”). Right: aligned functions (“amplitude variability”).

decomposing our functional responses into aligned functions and warping functions, as shown in Figure 1. For expensive models, we can then build an emulator or surrogate for the aligned functional responses and, under a suitable transformation, an emulator for the warping functions. These two emulators can be used to calibrate in such a way that a proper distance is used. The strengths of this approach are that (1) emulation is likely to be more accurate when applied to aligned data instead of misaligned data (Francom et al., 2022), (2) discrepancy and measurement error modeling can be done in such a way that it is isolated to the phase or the amplitude part of the model, and (3) the alignment procedure ensures that the isometry property holds, and therefore, that the calibration is performed using a proper distance (i.e., Euclidean distance in the transformed space properly reflects distance between warping functions). Hence, this work overtly connects the two well-established fields of Bayesian model calibration and functional data analysis that have historically had limited connection. This enables better emulation, more identifiable discrepancy modeling, and hence, more accurate (likely smaller) calibration uncertainty. Furthermore, we do this in such a way that it is accessible using established tools, not requiring new and specialized or expensive approaches to posterior sampling. The misalignment problem has not been adequately addressed (or even identified) in the Bayesian model calibration literature, and our method produces significantly better results than approaches that ignore the misalignment or only partially deal with it. Related work includes Kleiber et al. (2014), which proposed a functional calibration approach using a deformation that relies on the L^2 metric in the standard function space. However, this metric can exhibit a “pinching effect” degeneracy, which the elastic methods do not have, as demonstrated in Srivastava and Klassen (2016). Recent work by Guan et al. (2019) performed Bayesian model calibration utilizing phase variability; however the emulation was done on the likelihood, limiting extension to the richer class of calibration models that we consider (e.g., including discrepancy and measurement error). We demonstrate our approach on two dynamic material science applications described in the following subsections.

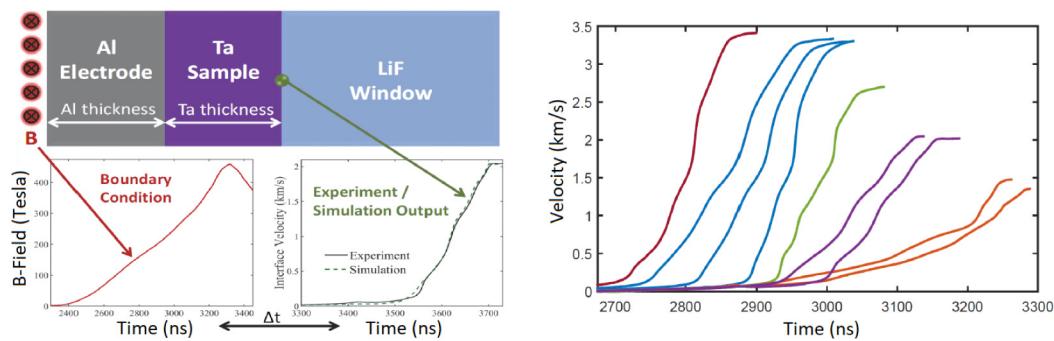


Figure 2. From Brown et al. (2014) and Brown and Hund (2018). Left: example of experiments at the Z-machine. Right: velocity response functions from nine experiments. LiF, lithium fluoride.

1.1. Equation of state exploration using Z-machine experiments. This problem is motivated by data from Brown et al. (2014), which conducted dynamic materials experiments on tantalum (Ta) generated with pulsed magnetic fields using Sandia National Laboratories' Z-machine (Savage et al., 2007). A basic version of the experiments performed at the Z-machine can be seen in Figure 2 (left). The Z-machine delivers electrical currents along an aluminum (Al) panel creating massive pressure driving an impulse into a Ta sample. Stress waves flow through the Ta, and the experiment results in measurements of velocity on the far side of the sample as a function of time. More details on the execution of the experiment can be found in Brown et al. (2014) and Lemke et al. (2005). Figure 2 (right) shows the functional response of nine experiments.

The computer model input parameters consist of calibration parameters of key interest to an equation of state (EoS) of Ta and experiment-specific parameters that are not completely known. More details about these input parameters appear in 5.1.

1.2. Material strength exploration using gas gun experiments. Boteler and Dandekar (2006) performed a series of experiments where a flyer was shot from a gas gun into a plate and the velocity of the opposite surface of the plate was measured (as a function of time) as the resultant shock wave moved through it. This experiment has similarities to the Z-machine example above, except that the shock waves that propagate through the plate are driven by the flyer's impact instead of the more continuous electrical current drive. Figure 3 shows three velocimetry curves measured during three of these flyer plate gas gun experiments. Walters et al. (2018) used these experiments to parameterize a strength model for Al. Also similar to the Z-machine computer model, flyer plate impact simulations include both inputs to be calibrated (strength model parameters) and experiment-specific parameters that are uncertain (e.g., the exact velocity of the flyer).

The structure of the rest of this paper is as follows. Section 2 gives a high-level overview of model calibration based on the Kennedy and O'Hagan (2001) framework and functional response extensions. Section 3.1 describes the types of variability for functional output, functional alignment with a proper distance metric, and how to measure the amount of variability in amplitude and phase space. Section 3.2 describes how to use the measures of variation in our elastic Bayesian model calibration method, and various modeling choices are described in section 3.3. Section 4 gives two simulated examples comparing the proposed elastic functional

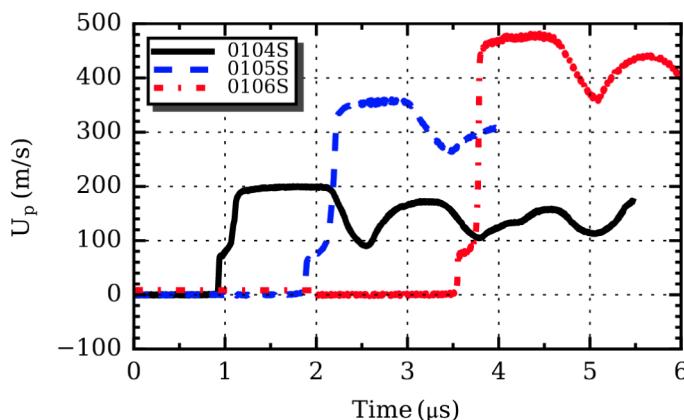


Figure 3. From [Walters et al. \(2018\)](#), the response from three gas gun experiments where both flyer and plate were Al alloys.

calibration method with functional response calibration that does not account for misalignment. Section 5 applies the proposed method to the two material model calibration problems detailed above. Lastly, section 6 provides general conclusions and considerations on future work. Supplementary material (Supplement.pdf [local/web 1.65MB]) includes an additional simulated example.

2. Bayesian model calibration.

2.1. Univariate response. The traditional approach to Bayesian model calibration, introduced by [Kennedy and O'Hagan \(2001\)](#) and expanded in [Higdon et al. \(2004\)](#), seeks to calibrate parameters of a model using observations. Let $y(\mathbf{x}, \mathbf{u})$ denote the model where \mathbf{x} denotes conditions that are certain, often fixed conditions of an experiment and \mathbf{u} denotes uncertain parameters in need of calibration. Let $z(\mathbf{x})$ denote an observation. In these approaches, \mathbf{x} could include functional variables such as space or time and/or conditions at which an experiment was performed. Let n be the number of observations or experimental conditions measured so that our observed data are $z(\mathbf{x}_1), \dots, z(\mathbf{x}_n)$. Then, the calibration model is

$$(2.1) \quad z(\mathbf{x}_i) = y(\mathbf{x}_i, \boldsymbol{\theta}) + \delta(\mathbf{x}_i) + \epsilon(\mathbf{x}_i), \quad \epsilon(\mathbf{x}_i) \sim \mathcal{N}(0, \sigma_\epsilon^2),$$

where $\boldsymbol{\theta}$ denotes the best set of calibration parameters for \mathbf{u} , δ denotes (latent) error in the form of the model (often called model discrepancy), and ϵ denotes measurement or observation error in z . The Gaussian likelihood specified here assumes that measurement errors are independent and identically distributed, but other measurement error structures can be used. The unknowns in this model are the calibration parameters $\boldsymbol{\theta}$, the measurement error variance σ_ϵ^2 , and the form of the discrepancy function. In order for this model to be identifiable, priors for each of the unknowns need to be chosen carefully. If n is small, the prior for σ_ϵ^2 will be influential. Additionally, there is a natural tradeoff between the calibration parameters and the model discrepancy, so at least one of these needs to be well constrained by the prior. For instance, we could use a Gaussian process prior for the discrepancy function δ in such a way that we prefer small-valued smooth functions ([Higdon et al., 2004](#)), or we could try to specify priors that prefer positive values or monotone functions ([Brynjarsdóttir and O'Hagan, 2014](#)).

In many realistic scenarios, the model y is expensive to evaluate, requiring powerful computing and nontrivial amounts of time. This slows down the evaluation of the likelihood and makes inference impractical. In these scenarios, we build a surrogate model (or emulator) to use in place of y , which is trained using a (relatively) small number of model evaluations, $\{y(\mathbf{x}_j, \mathbf{u}_j)\}_{j=1}^{N_{sim}}$ (Sacks et al., 1989). The full Bayesian approach to inference then seeks to learn all the unknowns (calibration parameters, model discrepancy, measurement error, and emulator parameters) conditional on all of the data (n observations and N_{sim} model runs). Higdon et al. (2004) use this approach with a Gaussian process emulator, while Kennedy and O'Hagan (2001) fix some of the Gaussian process emulator and/or discrepancy parameters in advance for computational and identifiability reasons. More generally, Liu et al. (2009) describe cases where modularization (or *cutting feedback* from parts of the model that lead to identifiability issues or poor Markov chain Monte Carlo (MCMC) mixing) will result in better posterior inference. Many modern practitioners use modularized approaches either for philosophical or computational reasons.

2.2. Functional response. Various research efforts have extended the Bayesian model calibration approach above to be more explicitly suited to functional response. This can vastly increase the training data size, but Gaussian processes with Kronecker covariance structures can help produce scalable models (Williams et al., 2006). A somewhat different approach is used by Gu and Berger (2016), which involves fitting a Gaussian process for each output while sharing some parameters across processes and, with a suitable discrepancy model, obtaining the calibration. Perhaps the most common approach is to project the functional response onto basis functions and build functional models for calibration in the reduced-dimension basis coefficient space (Bayarri et al., 2007a; Francom et al., 2019; Higdon et al., 2008). Let t denote the functional variable (e.g., time), and let $z(t, \mathbf{x}_i)$ denote an experimental measurement from the i th experiment at functional variable t . Similarly, let $y(t, \mathbf{x}_i, \mathbf{u})$ denote a simulation of the i th experiment at the functional variable t with input parameters \mathbf{u} . Then, an approach to Bayesian model calibration with functional response specifies that

$$(2.2) \quad z(t, \mathbf{x}_i) = y(t, \mathbf{x}_i, \boldsymbol{\theta}) + \delta(t, \mathbf{x}_i) + \epsilon(t, \mathbf{x}_i), \quad \epsilon(t, \mathbf{x}_i) \sim \mathcal{N}(0, \sigma_\epsilon^2).$$

The typical approach to inference is to discretize t onto a grid t_1, \dots, t_{N_T} , which generates N_T -dimensional vectors $\mathbf{z}(\mathbf{x}_i)$, $\mathbf{y}(\mathbf{x}_i, \boldsymbol{\theta})$, and $\boldsymbol{\delta}(\mathbf{x}_i)$ of the respective functions evaluated on the discretized grid, which simplifies the model to a multivariate representation:

$$(2.3) \quad \mathbf{z}(\mathbf{x}_i) = \mathbf{y}(\mathbf{x}_i, \boldsymbol{\theta}) + \boldsymbol{\delta}(\mathbf{x}_i) + \boldsymbol{\epsilon}(\mathbf{x}_i), \quad \boldsymbol{\epsilon}(\mathbf{x}_i) \sim N(\mathbf{0}, \sigma_\epsilon^2 \mathbf{I}).$$

As in the univariate case, when $y(t, \mathbf{x}, \boldsymbol{\theta})$ is expensive to evaluate, we will require an emulator or surrogate model in order to evaluate the likelihood function quickly. Higdon et al. (2008) project the model runs onto functional principal components and project the discrepancy onto a separate flexible basis, and inference is carried out in the resulting low-dimensional space. Bayarri et al. (2007b) project the model runs and discrepancy onto a wavelet basis and carry out the inference in the low-dimensional space. Francom et al. (2019) project the model runs onto functional principal components, allow the discrepancy to be full-dimensional, and fit the emulator in a modular fashion by keeping aspects of the emulator separate from the full Bayesian analysis.

All of these calibration approaches can be applied to misaligned functional data, but all will have difficulty with emulation and the specification of model discrepancy because they will only deal explicitly with amplitude variation. Additionally, distances computed between the different functions will not be proper distances since they would not account for the phase variability.

3. Elastic Bayesian model calibration. In this section, we will review elastic functional data analysis, introduce our elastic approach to calibration, and suggest various modeling choices and practical considerations.

3.1. Elastic functional data analysis.

3.1.1. Types of variability in functional outputs. When model outputs are functional, we must consider two types of variability in the functional outputs: amplitude and phase variability. Amplitude variability is variability in the output for a fixed time (t), or, more simply, y -axis variation. Phase variability is variability in time, or, more simply, x -axis variability.

For computer model applications, input variables can induce both phase and amplitude variability in the computer model outputs, resulting in fundamentally different shapes over the range of plausible inputs. Additionally, model discrepancy can result in an imperfect match between the computer model prediction and observed data at the correct value of the model inputs. With misaligned functional data, we must consider model discrepancy in both phase and amplitude variability to accurately represent discrepancy-induced shape distortions in the functional predictions.

When functional response variability is not solely driven by amplitude variability, pointwise (nonelastic) calibration (as in section 2.2) will produce estimates of the calibration parameters that may not agree with the true physical values, when these exist. Even when phase variability is taken into account, there exists a problem with the \mathbb{L}^2 metric known as the *pinching problem* (Ramsay and Li, 1998). Specifically, if we have two functions, f_1 and f_2 and the range(f_1) is entirely above the range(f_2), the \mathbb{L}^2 metric becomes degenerate and pinches the warped function. To address this problem, Srivastava et al. (2011) introduced a mapping for functional data called the *square-root velocity function*, or SRVF, that improves functional alignment and provides fundamental mathematical equalities that lead to the formal development of this topic. Moreover, the metric used in the alignment is a proper distance and avoids the pinching effects of the standard \mathbb{L}^2 metric in function space without the use of a penalty. We propose functional calibration metrics that account for both phase and amplitude variation while properly measuring the distance between functions.

3.1.2. Functional alignment. To explain metrics for comparing functional data in a calibration setting, we consider the comparison of two functions of t : $z(t, \mathbf{x}_i)$ and $y(t, \mathbf{x}_i, \boldsymbol{\theta})$. Varying $\boldsymbol{\theta}$ will change the shape of $y(t, \mathbf{x}_i, \boldsymbol{\theta})$, so we seek to find $\boldsymbol{\theta}$ such that $z(t, \mathbf{x}_i)$ and $y(t, \mathbf{x}_i, \boldsymbol{\theta})$ are *optimally matched*, where the optimality criterion considers the distance between the functions in both amplitude and phase. For notational convenience and to emphasize that we are comparing these functions for a fixed \mathbf{x}_i and $\boldsymbol{\theta}$, we rewrite $z(t, \mathbf{x}_i)$ and $y(t, \mathbf{x}_i, \boldsymbol{\theta})$ as $z(t)$ and $y(t)$ for the rest of this subsection. To measure the distance between $z(t)$ and $y(t)$, we use elastic functional data analysis (EFDA) (Srivastava and Klassen, 2016). The main premise behind EFDA is to construct a proper distance metric between the computational prediction $y(t)$ and the experimental data $z(t)$. To construct this metric, a continuous

mapping $\gamma_{y \rightarrow z}(t) : [0, 1] \rightarrow [0, 1]$ between $y(t)$ and the experimental data $z(t)$ is constructed such that γ is a diffeomorphism. The function $\gamma_{y \rightarrow z}(t)$ is referred to as a warping function, as it measures phase distortions in $y(t)$ such that $y \circ \gamma_{y \rightarrow z}(t) = y(\gamma_{y \rightarrow z}(t))$ aligns with $z(t)$. Srivastava et al. (2011) and Tucker et al. (2013) show that, by applying a specific transformation to the original functions $z(t)$ and $y(t)$, there exist simple expressions for the amplitude and phase distance between functions. We now describe how to construct this transformation and how to define the distance metrics on the transformed data.

The functions $z(t)$ and $y(t)$ are transformed to their SRVFs. That is, we define the SRVF of $f(t)$ as

$$(3.1) \quad q(t) = \text{sgn}(\dot{f}(t))\sqrt{|\dot{f}(t)|},$$

where \dot{f} denotes the time derivative of f . The SRVF is a bijective mapping up to a translation; that is, $f(t)$ can be uniquely determined from $q(t)$ and a single point on the curve $f(t)$.

Let $q_z(t)$ and $q_y(t)$ denote the SRVFs of $z(t)$ and $y(t)$, respectively. The warping function that aligns y to z , denoted $\gamma_{y \rightarrow z}$, can be estimated solving the optimization problem via dynamic programming (Tucker et al., 2013),

$$(3.2) \quad \gamma_{y \rightarrow z} = \arg \inf_{\gamma \in \Gamma} \|q_z - (q_y \circ \gamma)\sqrt{\dot{\gamma}}\|^2,$$

or using a Bayesian approach (Cheng et al., 2016; Lu et al., 2017). We utilize the same group structure as (Tucker et al., 2013), where Γ is the set of orientation-preserving diffeomorphisms of the unit interval $[0, 1]$: $\Gamma = \{\gamma : [0, 1] \rightarrow [0, 1] | \gamma(0) = 0, \gamma(1) = 1, \gamma \text{ is a diffeomorphism}\}$. The properties of the diffeomorphisms are what allow the bending and stretching described above and why we use the term *elastic*, and the norm is the standard \mathbb{L}^2 norm on the space of SRVFs.

An advantage of this approach to warping function estimation is that the analyst does not have to specify landmarks for function alignment; the estimation of γ is achieved by using the group structure of Γ . When we use the optimization in (3.2) in later sections, we will refer to this as a decomposition (specifically, the *warping decomposition*) because it decomposes a misaligned function into an aligned function and a warping function.

Given a warping function $\gamma(t)$ and aligned SRVF $(q_y \circ \gamma \sqrt{\dot{\gamma}})(t)$, we can construct measures of phase and amplitude variability. Specifically, amplitude variability is measured as

$$(3.3) \quad d_a(q_z, q_y) = \|q_z - (q_y \circ \gamma_{y \rightarrow z})\sqrt{\dot{\gamma}_{y \rightarrow z}}\|^2.$$

Srivastava et al. (2011) show that this \mathbb{L}^2 distance on the transformed and aligned SRVF functions is a proper distance metric for amplitude.

3.1.3. Measuring phase distance. Defining a measure of phase variability is more difficult than for amplitude variability because the space of warping functions, Γ , is an infinite-dimensional nonlinear manifold and therefore cannot be treated as a standard Hilbert space. Since we would like to exploit Riemannian-geometric structure when making inferences about the warping functions, we again apply a specific transformation to the warping functions such that we can use a standard distance metric (the \mathbb{L}^2 norm) to measure distance. Specifically,

we represent an element $\gamma \in \Gamma$ by the square root of its derivative $\psi = \sqrt{\gamma}$. Note that this is the same as the SRVF defined earlier and takes this simpler form because $\dot{\gamma} > 0$. The identity $\gamma_{z \rightarrow z}$ maps to a constant function with value $\psi_{z \rightarrow z}(t) = 1$, which corresponds to no warping. Since $\gamma(0) = 0$, the mapping from γ to ψ is a bijection, and one can reconstruct γ from ψ using $\gamma(t) = \int_0^t \psi(s)^2 ds$. An important advantage of this transformation is that, since $\|\psi\|^2 = \int_0^1 \psi(t)^2 dt = \int_0^1 \dot{\gamma}(t) dt = \gamma(1) - \gamma(0) = 1$, the set of all such ψ 's is the positive orthant of the Hilbert sphere $\Psi = \mathbb{S}_\infty^+$ (i.e., a unit sphere in the Hilbert space \mathbb{L}^2). In other words, the square-root representation simplifies the complicated geometry of Γ to a unit sphere. The distance between any two warping functions (i.e., the phase distance) is exactly the arc-length between their corresponding SRVFs on the unit sphere \mathbb{S}_∞ :

$$d_p(\gamma_1, \gamma_2) = d_\psi(\psi_1, \psi_2) \equiv \cos^{-1} \left(\int_0^1 \psi_1(t) \psi_2(t) dt \right).$$

While the geometry of $\Psi \subset \mathbb{S}_\infty$ is more tractable, it is still a nonlinear manifold, and computing distances remains difficult. Instead, we use a tangent (vector) space at a certain fixed point for further analysis. The tangent space at any point $\psi \in \Psi$ is given by $T_\psi(\Psi) = \{v \in \mathbb{L}^2 \mid \int_0^1 v(t) \psi(t) dt = 0\}$. To map between the representation space Ψ and tangent spaces, one requires the exponential and inverse-exponential mappings. The exponential map at a point $\psi \in \Psi$, denoted by $\exp_\psi : T_\psi(\Psi) \mapsto \Psi$, is defined as

$$(3.4) \quad \exp_\psi(v) = \cos(\|v\|)\psi + \sin(\|v\|) \frac{v}{\|v\|},$$

where $v \in T_\psi(\Psi)$. Thus, $\exp_\psi(v)$ maps points from the tangent space at ψ to the representation space Ψ . Similarly, the inverse-exponential map, denoted by $\exp_\psi^{-1} : \Psi \mapsto T_\psi(\Psi)$, is defined as

$$(3.5) \quad \exp_\psi^{-1}(\psi_1) = \frac{\kappa}{\sin(\kappa)} (\psi_1 - \cos(\kappa)\psi),$$

where $\kappa = d_p(\gamma_1, \gamma)$. This mapping takes points from the representation space to the tangent space at ψ .

The tangent space representation v is sometimes referred to as a *shooting vector*. As discussed previously, it can be sensible to warp simulations to the observations in this calibration framework, which corresponds to defining the tangent space relative to the observations z , which we denote as $\psi_{z \rightarrow z}$. The identity warping is defined as $\gamma_{z \rightarrow z}(t) = t$, which results in $\psi_{z \rightarrow z}(t) = 1$ and $\exp_{\psi_{z \rightarrow z}}^{-1}(\psi_{z \rightarrow z}(t)) = 0$. In this case, deviations in v from 0 represent deviations in phase from the experimental data.

In practice, there are situations when warping to the experiment presents difficulties, which we discuss in a later section. In these cases, we warp model runs and experiments to a common “template” function (e.g., one of the model runs that has all the peaks/valleys/characteristics of interest, possibly selected by subscreening the model runs visually). We then define distance between a model run and the experiment by considering their distances to the template. This is effectively changing the origin in the distance calculation. Just as changing the origin when calculating Euclidean distances does not change the distance between two points because the metric defined above is a proper distance metric, mathematical distance between misaligned

curves is unaffected by the choice of template. In practice, there can be differences, which leads us to introduce elastic Bayesian model calibration in the next section using an arbitrary template y^* , typically taken to be the experiment or one of the model runs. Hence, discussion of warping in the next section will involve warping experiment and simulations to the template to get quantities like $\gamma_{y \rightarrow y^*}$ and $\gamma_{z \rightarrow y^*}$ instead of the quantities used above, $\gamma_{y \rightarrow z}$ and $\gamma_{z \rightarrow z}$.

3.2. Using EFDA within Bayesian model calibration. With the metrics for distance defined in the previous section, we are ready to detail how to do Bayesian model calibration with misaligned functional responses, which we call elastic Bayesian model calibration.

We decompose the observations from experiment i into aligned functions and warping functions (each specific to experiment i) so that

$$(3.6) \quad z(t, \mathbf{x}_i) = \tilde{z}(t, \mathbf{x}_i) \circ_t \gamma_{z \rightarrow y^*}(t, \mathbf{x}_i),$$

where \circ_t emphasizes that the composition is only in t such that $f(a, b) \circ_a g(a, b) = f(g(a, b), b)$. If the template y^* is taken to be the observations z , this is the identity warping, and nothing happens in this step. We similarly decompose each simulation (e.g., simulation j) with

$$(3.7) \quad y(t, \mathbf{x}_j, \mathbf{u}_j) = \tilde{y}(t, \mathbf{x}_j, \mathbf{u}_j) \circ_t \gamma_{y \rightarrow y^*}(t, \mathbf{x}_j, \mathbf{u}_j)$$

using the warping decomposition of (3.2). To facilitate modeling with proper distance metrics, we transform the warping functions into shooting vector space with

$$(3.8) \quad \mathbf{v}_{z \rightarrow y^*}(\mathbf{x}_i) = \exp_{\psi_{z \rightarrow y^*}}^{-1} \left(\sqrt{\dot{\gamma}_{z \rightarrow y^*}(\mathbf{x}_i)} \right),$$

$$(3.9) \quad \mathbf{v}_{y \rightarrow y^*}(\mathbf{x}_j, \mathbf{u}_j) = \exp_{\psi_{y \rightarrow y^*}}^{-1} \left(\sqrt{\dot{\gamma}_{y \rightarrow y^*}(\mathbf{x}_j, \mathbf{u}_j)} \right).$$

We can then use these aligned model runs and associated shooting vectors for emulator building, if desired.

3.2.1. Calibration when emulation is unnecessary. Our calibration model using the aligned data and shooting vectors is specified with

$$(3.10) \quad \tilde{z}(t, \mathbf{x}_i) = \tilde{y}(t, \mathbf{x}_i, \boldsymbol{\theta}) + \delta_{\tilde{y}}(t, \mathbf{x}_i) + \epsilon_{\tilde{z}}(t, \mathbf{x}_i), \quad \epsilon_{\tilde{z}}(t, \mathbf{x}_i) \sim \mathcal{N}(0, \sigma_{\tilde{z}}^2),$$

$$(3.11) \quad \mathbf{v}_{z \rightarrow y^*}(\mathbf{x}_i) = \mathbf{v}_{y \rightarrow y^*}(\mathbf{x}_i, \boldsymbol{\theta}) + \boldsymbol{\delta}_v(\mathbf{x}_i) + \boldsymbol{\epsilon}_v(\mathbf{x}_i), \quad \boldsymbol{\epsilon}_v(\mathbf{x}_i) \sim \mathcal{N}(0, \sigma_v^2 \mathbf{I}).$$

Notice the similarity of (3.10) and (2.2). Equation (3.10) is merely doing the functional response Bayesian model calibration of (2.2) with aligned functions. Equation (3.11) is a standard multivariate response calibration of the shooting vectors. Doing one or the other of these calibrations will either calibrate to the shapes of curves or their timing, while using both ensures we match the shape and timing. This framework also allows us to separately specify the discrepancy in the aligned functional responses or their warping functions, which is a key benefit. For instance, if the discrepancy is a timing shift, that can be modeled directly through $\boldsymbol{\delta}_v(\mathbf{x}_i)$. A discrepancy that would add a shape characteristic to the curves could be modeled directly through $\delta_{\tilde{y}}(t, \mathbf{x}_i)$. Furthermore, these two types of discrepancy could be used at the same time, making for a very practical discrepancy modeling framework.

In order to produce inferences with this model, we discretize t so that (3.10) can be rewritten in a vector form, $\tilde{z}(\mathbf{x}_i) = \tilde{y}(\mathbf{x}_i, \boldsymbol{\theta}) + \boldsymbol{\delta}_{\tilde{y}}(\mathbf{x}_i) + \epsilon_{\tilde{z}}(\mathbf{x}_i)$, and the joint likelihood can then be written as

$$\begin{aligned} & f(\tilde{z}(\mathbf{x}_1), \dots, \tilde{z}(\mathbf{x}_n), \mathbf{v}_{z \rightarrow y^*}(\mathbf{x}_1), \dots, \mathbf{v}_{z \rightarrow y^*}(\mathbf{x}_n) | \boldsymbol{\theta}, \sigma_{\tilde{z}}^2, \sigma_v^2, \boldsymbol{\beta}_{\tilde{y}}, \boldsymbol{\beta}_v) \\ &= \prod_{i=1}^n [\mathcal{N}(\tilde{z}(\mathbf{x}_i) | \tilde{y}(\mathbf{x}_i, \boldsymbol{\theta}) + \boldsymbol{\delta}_{\tilde{y}}(\mathbf{x}_i), \sigma_{\tilde{z}}^2 \mathbf{I}) \mathcal{N}(\mathbf{v}_{z \rightarrow y^*}(\mathbf{x}_i) | \mathbf{v}_{y \rightarrow y^*}(\mathbf{x}_i, \boldsymbol{\theta}) + \boldsymbol{\delta}_v(\mathbf{x}_i), \sigma_v^2 \mathbf{I})], \end{aligned}$$

where $\boldsymbol{\beta}_{\tilde{y}}$ and $\boldsymbol{\beta}_v$ parameterize the discrepancy terms of \tilde{y} and v , respectively. With a prior specified for $\boldsymbol{\theta}$, $\sigma_{\tilde{z}}^2$, σ_v^2 , $\boldsymbol{\beta}_{\tilde{y}}$, and $\boldsymbol{\beta}_v$, the posterior

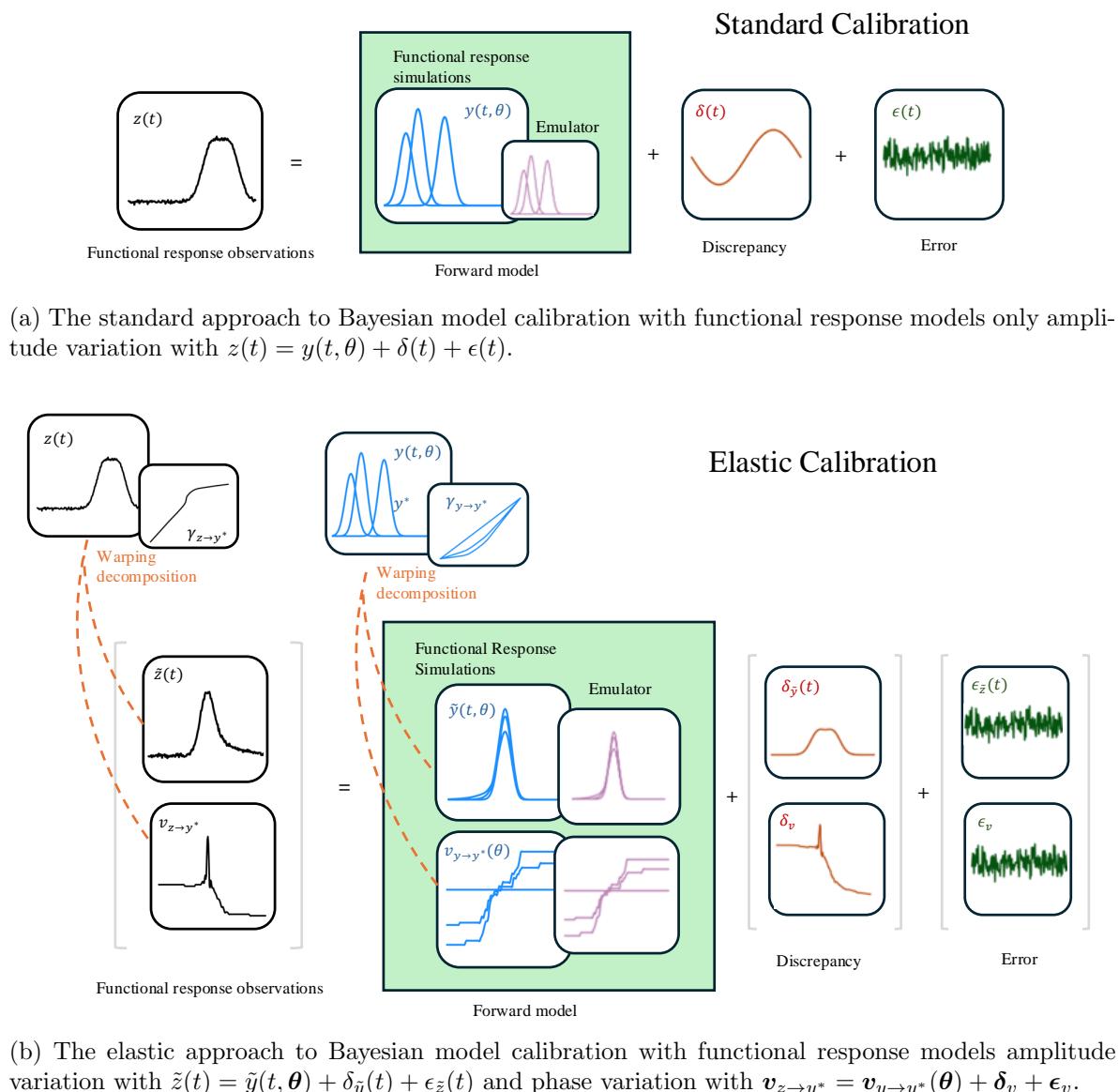
$$\pi(\boldsymbol{\theta}, \sigma_{\tilde{z}}^2, \sigma_v^2, \boldsymbol{\beta}_{\tilde{y}}, \boldsymbol{\beta}_v | \tilde{z}(\mathbf{x}_1), \dots, \tilde{z}(\mathbf{x}_n), \mathbf{v}_{z \rightarrow y^*}(\mathbf{x}_1), \dots, \mathbf{v}_{z \rightarrow y^*}(\mathbf{x}_n))$$

is proportional to likelihood multiplied by priors and can be sampled with MCMC. Note that this approach requires the warping decomposition (and computer model) to be called for each likelihood evaluation. Even though this is just the decomposition of a single curve and fairly fast, this expense can make emulation more desirable.

3.2.2. Calibration when emulation is necessary. Assume that the set of N_{sim} model runs is decomposed into aligned functions and warping functions. We can then use separate or joint emulators with training inputs $\{\mathbf{x}_j, \mathbf{u}_j\}_{j=1}^{N_{sim}}$ and outputs $\{\tilde{y}(\mathbf{x}_j, \mathbf{u}_j), v_{y \rightarrow y^*}(\mathbf{x}_j, \mathbf{u}_j)\}_{j=1}^{N_{sim}}$. A full Bayesian approach to emulation and calibration results in a joint likelihood of the observation data and model runs, while a modular Bayesian approach fits the emulator first and then uses the emulator to perform the calibration. Because these models result in likelihoods and posteriors that are not vastly different from the likelihood without an emulator, we omit the explicit likelihood here. For Gaussian process emulators, the likelihood under the full Bayesian functional calibration model (in basis space) can be seen in Higdon et al. (2008). In fact, we have purposefully designed this model so that the inference could be carried out using practical frameworks such as the ones in Bayarri et al. (2007b), Francom et al. (2019), Gu and Wang (2018), Higdon et al. (2008), and Williams et al. (2006). Each of these approaches can be applied by merely replacing their use of the original functional responses z and y with aligned responses (\tilde{z} and \tilde{y}) and shooting vectors ($\mathbf{v}_{z \rightarrow y^*}$ and $\mathbf{v}_{y \rightarrow y^*}$). That being said, there are a number of modeling choices and practical considerations that can make this approach successful that we discuss in the next section.

Figure 4 shows a schematic of what the different parts of the elastic Bayesian calibration model are (Figure 4(b)) compared to the standard approach to calibration with functional response (Figure 4(a)). The warping decomposition is a preprocessing step that results in two datasets, after which standard functional response calibration can be applied using the two datasets. This is a simple approach, but we will demonstrate that it can result in calibration that is significantly more accurate.

3.3. Modeling choices and practical considerations. To this point, we have discussed how using standard functional response calibration tools with misaligned functional responses is possible but prone to problems. We also mentioned that emulation techniques for functional response do not perform well when applied to misaligned functional responses, as demonstrated



(b) The elastic approach to Bayesian model calibration with functional response models amplitude variation with $\tilde{z}(t) = \tilde{y}(t, \theta) + \delta_{\tilde{y}}(t) + \epsilon_{\tilde{z}}(t)$ and phase variation with $v_{z \rightarrow y^*} = v_{y \rightarrow y^*}(\theta) + \delta_v + \epsilon_v$.

Figure 4. The standard approach to Bayesian model calibration with functional response (a) compared to the elastic approach (b). The elastic approach first isolates phase and amplitude variation as a preprocessing step, and then uses the same techniques used to do inference under the standard approach.

in Francom et al. (2022). We then showed how EFDA methods use proper distance metrics by aligning functions in the SRVF space and using (1) the distance between aligned functions and (2) the distance between shooting vectors (transformed warping functions). Then, we introduced how to frame functional response computer model calibration such that it uses these proper elastic metrics. Simply put, we use the warping decomposition to separate our misaligned functional responses into aligned functional responses and shooting vectors, and we use these two new datasets instead of the original data when calibrating.

Below, we discuss additional modeling choices and practical considerations for using this methodology.

3.3.1. Warping decomposition. *Uncertainty*: Because we are using an optimization technique to obtain the warping functions, we are fixing a part of the model that could be considered uncertain. This is a modeling choice, similar to the choice of [Kennedy and O'Hagan \(2001\)](#) to fix some emulator parameters at maximum likelihood values or to the choice of [Higdon et al. \(2008\)](#) to not allow for uncertainty in the basis representation of the functional response (i.e., the functional principal components are fixed). As with these other modeling choices, a potential consequence is that uncertainty is underestimated. If the warping functions were to be inferred jointly with all of the other unknowns in a Bayesian framework, this would lead to a much greater computational burden and would require much more specialized calibration software. Similar to emulation, a possible shortcut could be to use modular Bayes techniques to propagate uncertainty from the warping decomposition to calibration uncertainty while cutting the feedback from the calibration to the warping decomposition ([Liu et al., 2009; Plummer, 2015](#)).

Regularization: Another practical consideration is for the family of warping functions allowed in the warping decomposition, Γ . If these functions are not smooth or regularized, their variation can be difficult to predict using the parameters, resulting in emulators with large residual variance. However, if they are overregularized, they may not align the functional responses enough. In case one wants to control the amount of warping or *elasticity*, this can be done as described in [Wu and Srivastava \(2011\)](#) using a penalty on (3.2). In the examples and applications below, we arrive at regularization choices by trial and error, where we are satisfied with a level of regularization when it results in nicely aligned functional responses with relatively smooth warping functions.

Choice of alignment reference: This decomposition can occur by either aligning the model to the experiment or by aligning to some other common element (e.g., one of the model runs). Alignment to the experiment is natural in many cases, but there are a few reasons why aligning to one of the model runs could be useful. First, the model runs have no discrepancy or noise, which means that warping to them can be more stable. Second, exploration of the computer model is often of interest even if there is no calibration data, in which case emulation is frequently used and can be performed more accurately when accounting for misalignment.

Modeling transformed aligned curves: Recall that we use the SRVF transformation of the original curves to obtain the warping decomposition. However, we opt not to build the model for the aligned data in the SRVF space (i.e., we model z and y instead of q_z and q_y). We do this because modeling in derivative space means that, when models are transformed back to the native space (e.g., to make predictions), errors are integrated and unrealistic heteroskedasticity arises. Using \mathbb{L}^2 distance between the aligned curves (rather than their SRVFs) still results in a proper metric, as long as the SRVFs were used for the alignment. Note that we do not experience heteroskedasticity issues when transforming from shooting vectors to warping functions because of the highly constrained shape of the warping functions.

3.3.2. Emulation. [Francom et al. \(2022\)](#) showed that taking alignment into account can improve emulator accuracy and efficiency, although their approach relied on a model for landmarks (rather than shooting vectors) to build warping functions. They found that, in

many cases, training separate emulators for aligned functions and warping functions is more desirable than training a single emulator for both. This is because the warping function model will often have larger unexplained variation (i.e., larger residual variance than the aligned data model, even under appropriate standardization) due to the latent nature of the warping functions. Subtle variations in the warping function regularization level can result in fairly significant variations in the shooting vectors, and while this variation is less pronounced when transforming back into warping function space, the larger variation can “corrupt” (Liu et al., 2009) the modeling of the aligned data unless careful precautions are taken. The easiest precaution is to merely create the two emulators independently. This works well in practice.

3.3.3. Discrepancy models. The decomposition of misaligned data into aligned data and shooting vectors not only facilitates better emulation and calibration likelihood via proper distance metrics, it also facilitates more realistic discrepancy modeling. For instance, if the discrepancy is a time shift, that can be expressed through the shooting vector discrepancy model δ_v . If the discrepancy is a change to one feature of the curves, that can be handled directly through the aligned data discrepancy model $\delta_{\tilde{y}}$. The specification of $\delta_{\tilde{y}}$ can be reasoned about in a natural way (compared to trying to specify discrepancy in the misaligned space); for instance, a set of basis functions could be used as in Higdon et al. (2008). However, the specification of δ_v is more nuanced because it is in a transformed space. For instance, adding a constant in that space (e.g., $\delta_v(\mathbf{x}_i) = \mathbf{1}$) has no effect because the shooting vectors are scaled by their norm in the exponential map that transforms them back to an SRVF on the unit sphere. Hence, a constant time shift discrepancy is not achieved by adding a constant to the shooting vectors. In the simulated example with discrepancy in section 4.2, we demonstrate modeling $\delta_{\tilde{y}}$ and δ_v with B-spline basis functions and compare to the case where the discrepancy is only defined in the standard way. In the flyer plane analysis below, we use a piecewise linear basis for δ_v to capture time shifts. In all basis expansions for δ_v , we exclude any intercept term because constant shifts in shooting vectors are not useful.

3.3.4. Residual error models. We will typically assume independence between ϵ_z and ϵ_v , but this can be relaxed. Of greater interest is the assumption of independence within ϵ_z and ϵ_v . We note here that adding a correlation structure to the residual model still results in appropriate distance metrics in the likelihood calculation. This can be done explicitly to avoid the problems identified in Brown and Hund (2018).

3.4. Identifiability. There are multiple places where identifiability is important to consider in Bayesian model calibration. In a full Bayesian approach, there are identifiability concerns between estimating the parameters, discrepancy, emulator, and measurement error. Modularization can improve parts of this; for example, modularizing the emulator removes it as a contributor to the lack of identifiability (Liu et al., 2009). By separating amplitude and phase variability, allowing us to separately emulate them and separately include discrepancy and measurement error models, we improve identifiability. For example, if the discrepancy is a time shift (which is common), the only way to specify that in the standard calibration approach is to allow flexible discrepancy across amplitude space. Of course, that does not put the discrepancy model explicitly in the time domain, which will lead to a less constrained (and hence less identifiable) discrepancy in the amplitude domain. Our approach allows for

specifying discrepancy limited to the time domain (or amplitude domain, or both), which improves identifiability. It should be noted that decomposing a curve into phase and amplitude variation is not necessarily identifiable (and is separate from all the sources of identifiability mentioned above). However, regularizing the warping functions makes the system identifiable. When used appropriately, any decrease in identifiability because of the warping transformation is easily offset by the better identified calibration model it enables.

3.4.1. Using standard functional calibration tools for elastic calibration. Our formulation results in a Gaussian-likelihood pointwise calibration using the aligned responses and the shooting vectors instead of the original misaligned functional responses. The likelihood function for the elastic calibration is formed by combining vector versions of (3.10) and (3.11). This means that the same tools that were discussed in section 2.2 can be used for elastic calibration, making the application of our proposed methods very practical. For instance, SEPIA (Simulation-Enabled Prediction, Inference, and Analysis) (Gattiker et al., 2023) implements the model of Higdon et al. (2008), and Francom et al. (2023) implements a similar model but allows for modularization of the emulator as in Francom et al. (2019) with an MCMC algorithm that handles posterior multimodality via tempering. Both of these tools can be used to do elastic calibration, and the warping decomposition can be achieved using the *fdasrsf* R-package or equivalent python package *fdasrsf* (Tucker, 2023).

Further, there are several choices available to emulate computer model runs with functional response as described in Francom et al. (2022), Hutchings et al. (2023), and Collins et al. (2024). In the examples that follow, we use a Bayesian multivariate adaptive regression spline (BMARS) (Francom et al., 2018) emulator for both the aligned simulated curves and their corresponding shooting vectors, paired with the calibration approach in Francom et al. (2023). We chose the BMARS due to its performance and the ability to scale to a large number of training points, which we have in our EoS and material strength data examples, although the methods are agnostic to emulator choice.

4. Simulated examples.

4.1. Example with comparison. To illustrate the intuition behind this method, we simulate an example problem with three parameters to calibrate and misaligned functional response and no discrepancy (a similar example with discrepancy is included in the supplemental material (Supplement.pdf [local/web 1.65MB])). The example is constructed using the following model to generate misaligned functional responses, where each function is a Gaussian pdf and the parameters control the location and height of the peak:

$$y(t, \mathbf{u}) = \frac{u_1}{0.05\sqrt{2\pi}} \exp\left(-\frac{1}{2} \left(\frac{t - (\sin(2\pi u_0^2)/4 - u_0/10 + 0.5)}{0.05} \right)^2\right) + 0u_2.$$

The calibration parameters are $\mathbf{u} = [u_0, u_1, u_2]$, although the functions only depend on the first two of these parameters. There are no experimental parameters, so we omit \mathbf{x} from the model specification used in the previous sections. We generated a set of 100 functions (model runs) using $\mathbf{u}_1, \dots, \mathbf{u}_{100}$, where each \mathbf{u}_j was sampled uniformly within the unit cube. The ‘experimental data’ $z(t)$ were generated using the parameter values $\boldsymbol{\theta} = [0.1028, 0.5930, 0]$ and adding Gaussian noise with standard deviation 0.05, but without systematic bias (discrepancy). Figure 5 presents the simulated model runs in gray and the experimental data shown in black.

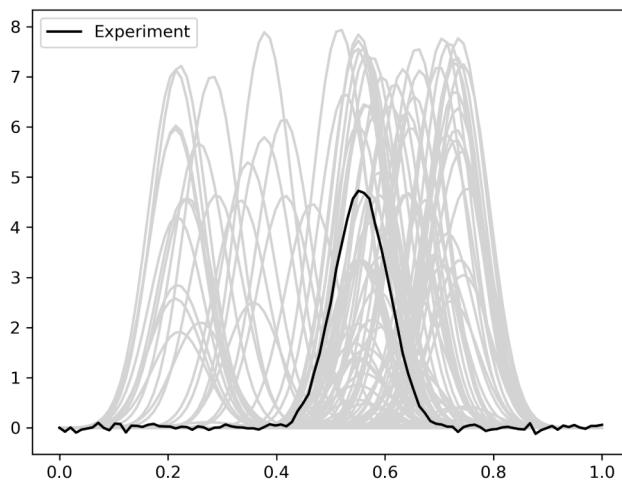


Figure 5. Simulated curves $y(t, \mathbf{u}_1), \dots, y(t, \mathbf{u}_{100})$ and experiment data $z(t)$ for the simulated example.

We separated the phase and amplitude by aligning the simulated curves to one of the model runs utilizing the elastic methodology from section 3.1. Figure 6(a) shows the aligned model runs (gray) and the experimental data (black), while Figure 6(b) presents the corresponding warping functions of both the simulated curves ($\gamma_{y \rightarrow z}(t, \mathbf{u}_j)$ for $j = 1, \dots, 100$) and the experimental data ($\gamma_{z \rightarrow z}(t) = t$). Figure 6(c) presents the corresponding shooting vectors for the simulation ($\mathbf{v}_{y \rightarrow z}(\mathbf{u}_j)$) and the experimental data ($\mathbf{v}_{z \rightarrow z}$).

We built separate BMARS emulators for the aligned curves and the shooting vectors and performed a (modular) elastic Bayesian model calibration as described in section 3.2 to infer the posterior distribution of $\boldsymbol{\theta}$ along with posterior predictions of the curves.

Figure 6 presents some posterior predictive samples after calibration of the aligned curves and the corresponding warping functions and shooting vectors (in blue) with their simulations (in gray) and for the experimental data (in black). Similarly, Figure 6(d) presents some posterior predictive samples in the original data space (in blue) with the simulated curves/model runs (in gray) and the experimental data (in black). The predictive samples cover the experimental data with small uncertainty, showing good predictive performance. A pairwise plot summarizing the posterior samples of $\boldsymbol{\theta}$ (marginal and bivariate) is shown in Figure 7. The true value of $\boldsymbol{\theta}$ is represented by the mark (x) in the lower triangular panels and the vertical line on the diagonal panels. The true parameter values are covered by high-density regions of the posterior distribution, indicating that they are recovered well by the model calibration. Additionally, the posterior distribution of the nuisance parameter θ_2 resembles a uniform distribution (prior) and does not affect the calibration. We note that the construction of the test function results in a bimodal posterior for θ_0 .

To compare our elastic Bayesian calibration method to a standard functional Bayesian calibration approach, we fitted a BMARS emulator on the original simulated curves and directly performed a modular Bayesian model calibration with no alignment. Figure 8 presents the posterior predictive samples after calibration of the simulated data in the original data space, and Figure 9 shows the corresponding parameter samples. We observe that the predictive samples do not resemble the experimental data well and have more than one mode/valley and

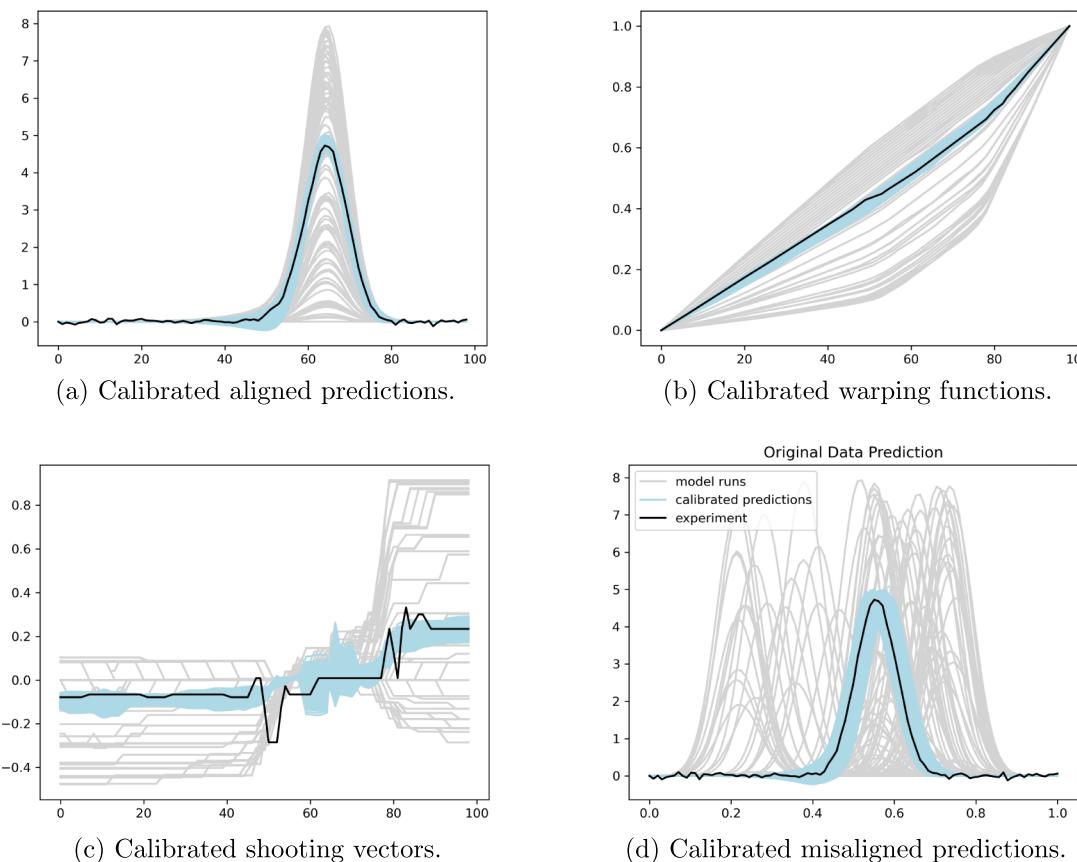


Figure 6. Posterior predictive samples after calibration of the simulated data for Example 1.

large uncertainty, given that phase variability is not taken into account. The primary reason for the improved performance of the elastic calibration in this simulation is the improved emulator accuracy. The elastic emulator root mean squared error (RMSE) on test data is 0.082 versus the standard emulator RMSE of 0.562 on the same test data.

Beyond comparing to the standard functional Bayesian calibration approach, we also compare to calibration using features, calibration only using the aligned data, and calibration where the emulator is for the likelihood instead of the forward model. The results of these approaches are compared in Figure 10, which shows the samples of θ_0 and θ_1 under each method (the samples of θ_2 are uniform in each case and are omitted from the figure) in the lower left. The true setting of θ_0 and θ_1 is denoted with a red x. The top left and bottom right plots show the marginal posterior density for θ_0 and θ_1 , respectively. The top right shows the 95% contour of the joint samples. There are many insights from this figure. When emulating the likelihood, we evaluated the mean squared error between the training sample aligned curves and the experiment aligned curve, repeated the same for the shooting vectors, built a joint BMARS emulator for the two mean squared errors, and calibrated to the zero vector. There are various ways to explore this further, but they are not the focus of this paper.

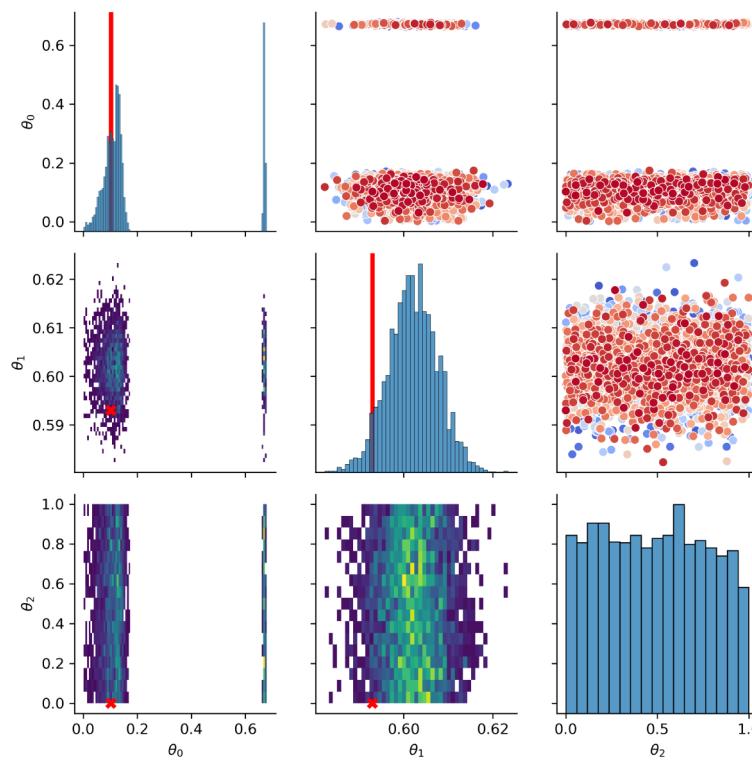


Figure 7. Pairs plot of parameters after calibration to simulated example data with the ‘true’ parameter combination shown by the x and vertical lines.

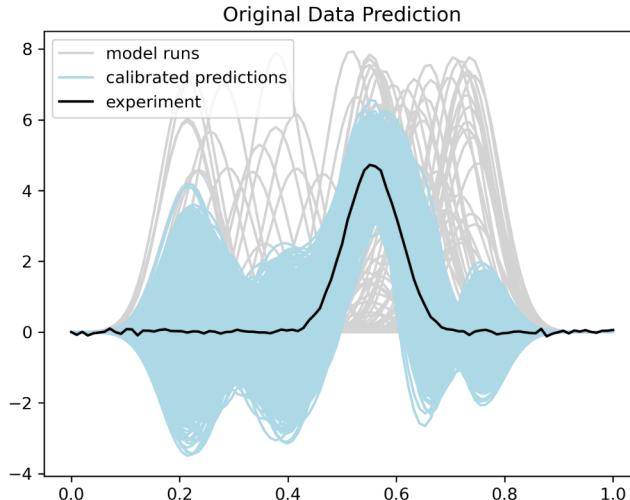


Figure 8. Calibrated predictions under the Bayesian calibration that does not account for misalignment.

This emulator fits well, but clearly not well enough to capture all of the information that the elastic approach does. The calibration that only uses aligned data is good at capturing θ_1 , but loses most information about θ_0 . The standard and elastic approaches are those shown in

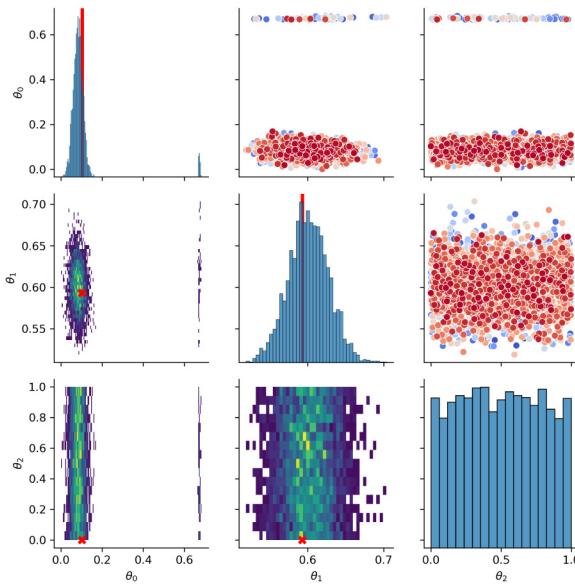


Figure 9. Pairs plot of the posterior samples of the parameters after standard calibration for the simulated example, where the true value is marked by the x and vertical line. The emulator was fitted directly on the original data. Note that the axis limits are different from Figure 7.

Figures 9 and 7, respectively, but put on the same axes, we see that the standard approach loses much information about θ_1 . For the feature-based approach, we selected two features: the maximum of each curve and the timing of the maximum. Given the form of the functions, this should communicate all of the information about θ_0 and θ_1 . However, because the observations were noisy, the features for the observed curve were slightly biased, and this biased the calibration. In practice, features are usually not easy to select, are frequently noisy, and usually do not capture all of the information in the model, so the feature-based calibration could be biased and lose information. To summarize this comparison, all five approaches started with the same data, but the elastic approach resulted in the smallest uncertainty that still captured the truth.

4.2. Example with discrepancy. In this example, we demonstrate the kind of discrepancy modeling that can be included in elastic Bayesian model calibration. We use the function

$$y(t, \mathbf{u}) = \exp \left\{ -2(t - 0.5u_0 - 1.5)^2 / (u_1 + 0.1) \right\} - u_0 \exp \left\{ -10(t - 4)^2 \right\},$$

which consists of two bumps: a positive bump with shifted location and a negative bump with shifted scale. There is no x variable here for simplicity, although the examples in the next section include x variables. We generate 200 model runs using values of \mathbf{u} from the unit square. We treat as our true function

$$z(t) = 1.1 \exp \left\{ -2(t - 0.5(0.6) - 1.5)^2 / (0.3 + 0.1) \right\} - 0.6 \exp \left\{ -10(t - 4.2)^2 \right\},$$

which is the simulator evaluated at $u_0 = 0.6$, $u_1 = 0.3$ with two forms of discrepancy; the true function has a larger first bump and shifted second bump. We include no measurement error.

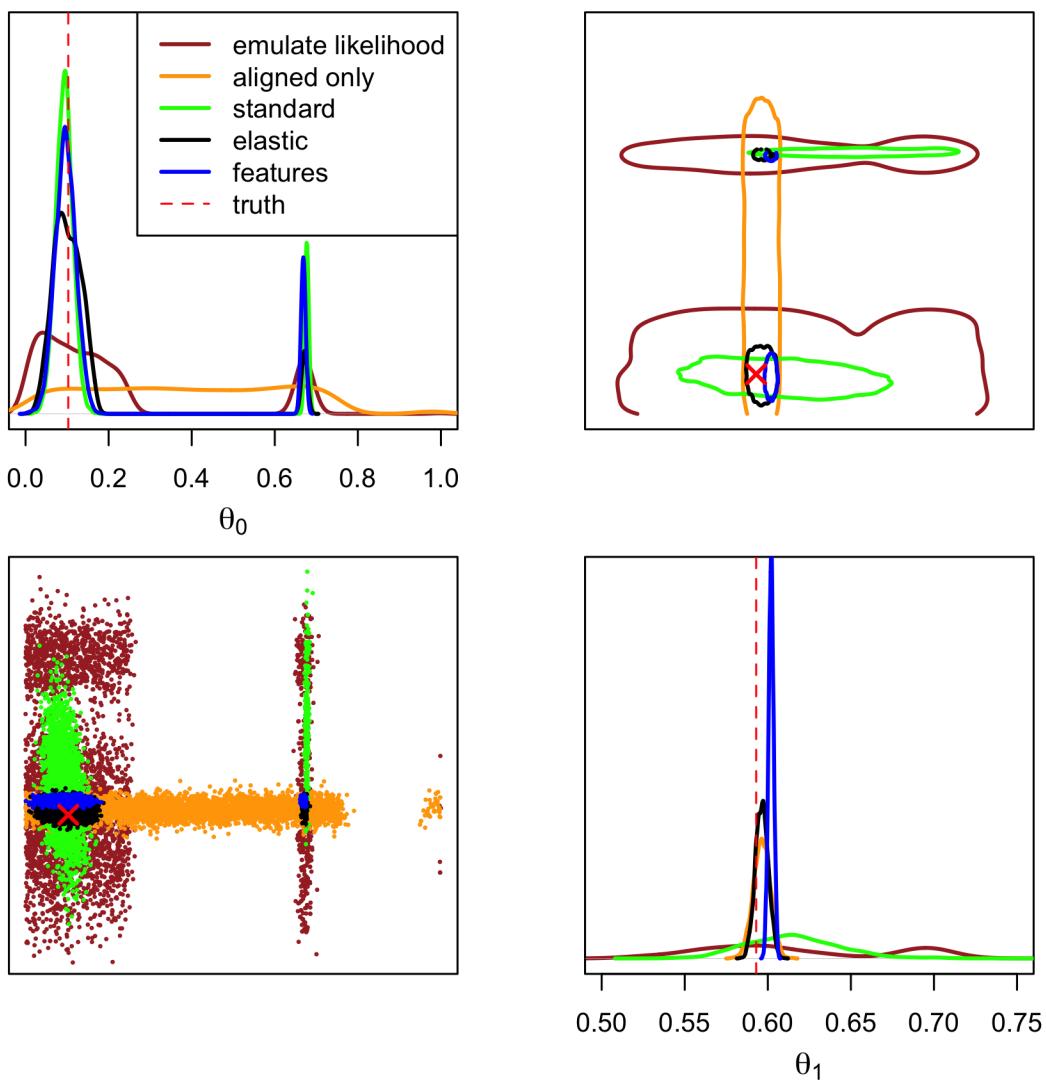


Figure 10. Comparison of the elastic calibration posterior samples with the posterior samples from four other approaches on the simulated example.

The model runs and true function are shown in Figure 11. The discrepancy is such that the parameters cannot replicate the behavior of the true function. The challenge of discrepancy modeling will be to do it in such a way that our parameter inference is not biased. Because of the timing discrepancy in the second bump, traditional functional response calibration will struggle because an amplitude-only discrepancy will essentially lead to us losing information from the second bump. We use basis expansions for each discrepancy model so that $\delta_{\tilde{y}} = \mathbf{D}_{\tilde{y}}\boldsymbol{\beta}_{\tilde{y}}$ and $\boldsymbol{\delta}_v = \mathbf{D}_v\boldsymbol{\beta}_v$. If our discrepancy was indexed by variable \mathbf{x} in this problem, we would use $\boldsymbol{\beta}_{\tilde{y}}(\mathbf{x})$ and $\boldsymbol{\beta}_v(\mathbf{x})$ in place of $\boldsymbol{\beta}_{\tilde{y}}$ and $\boldsymbol{\beta}_v$. For both $\mathbf{D}_{\tilde{y}}$ and \mathbf{D}_v we use B-spline basis functions shown in Figure 12. The first panel in Figure 12 shows that we limit our discrepancy basis functions to the part of t that corresponds to the first bump because we know that is

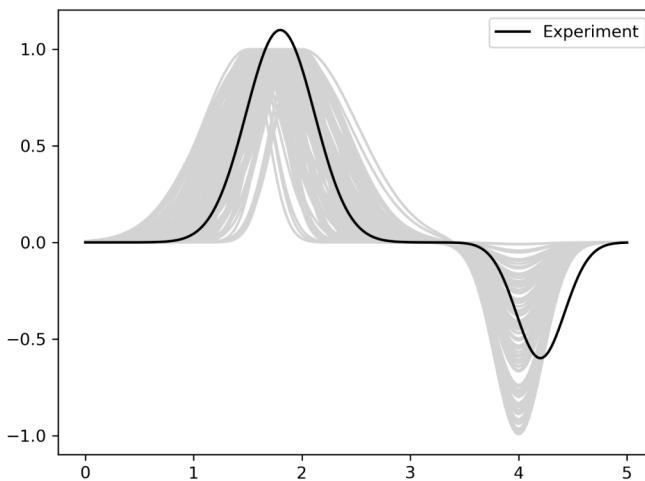


Figure 11. Simulated curves $y(t, \mathbf{u}_1), \dots, y(t, \mathbf{u}_{200})$ and experiment data $z(t)$ for the second simulated example.

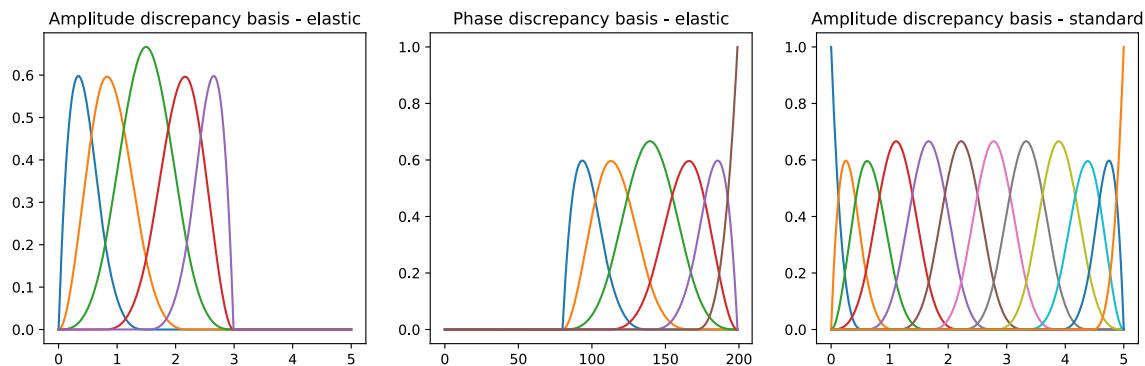


Figure 12. Basis functions defining $\mathbf{D}_{\tilde{y}}$ (left) and \mathbf{D}_v (middle) and basis functions used for standard calibration (right).

where the amplitude discrepancy is. The second panel in Figure 12 shows that we limit our phase discrepancy to the timing of the second bump because we know there is a phase discrepancy there. When we perform calibration with these discrepancy models, we are able to recover the settings of $\boldsymbol{\theta}$ that generated the data, as shown in Figure 13, while standard calibration is more likely to get biased parameter inference. Figure 14 shows the posterior predictions of our aligned data, shooting vectors, warping functions, and misaligned data when discrepancies are included and excluded. Figure 15 shows the inferred discrepancy under the elastic and standard approaches. Under the standard approach, the timing shift is captured by a sinusoidal discrepancy in amplitude, which limits the amount of information this part of the calibration can provide. On the other hand, the elastic approach finds a timing shift of 0.2 (the correct amount).

5. Dynamic material model calibration. We demonstrate the application of the elastic functional Bayesian calibration method to two real world applications, a Ta EoS calibration and an Al material strength calibration.

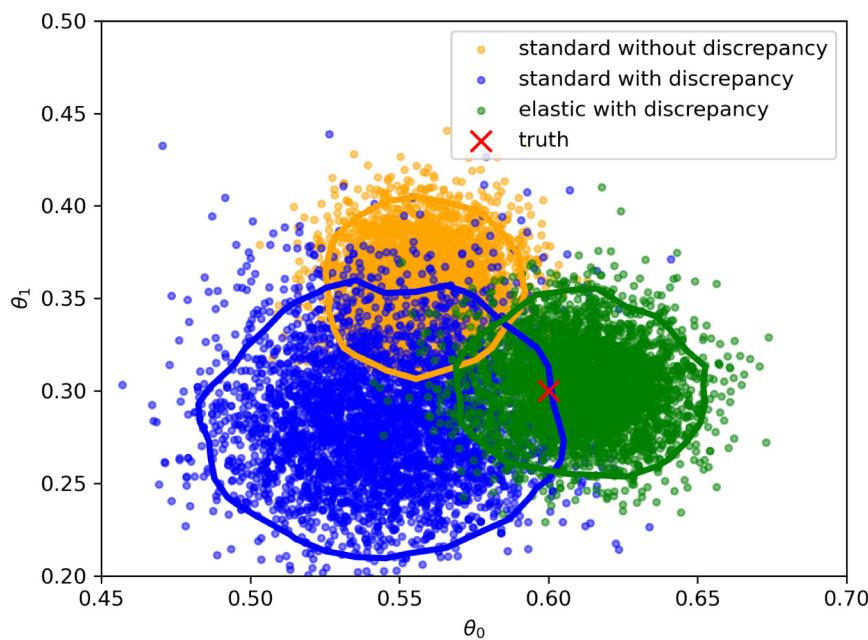


Figure 13. Comparison of elastic and standard inference of the calibration parameters. The points are posterior samples, while the lines are the 95% contour for each posterior. Properly including the phase discrepancy in the elastic model allows us to get more accurate calibration than the standard approach, which only includes amplitude discrepancy.

5.1. Ta Z-machine. In this application, we seek to calibrate the EoS of Ta with data generated from pulsed magnetic fields (Brown et al., 2014). We are seeking to estimate parameters describing the compressibility (relationship between pressure and density) to understand better how materials compress to extreme pressures. Ta is an ideal material for this study because it is able to remain in its initial crystal structure to pressures up to 10 million times standard atmospheric pressure (Soderlind and Moriarty, 1998). A description of the Ta experiments is shown in Figure 2. The experiments were conducted using Sandia National Laboratories’ Z-machine, which is a pulsed power drive that can deliver massive electric currents over short time scales. These currents were forced to flow along an Al panel, producing a large magnetic pressure that drives a time-dependent stress wave (impulse) into the system. Ta samples and transparent LiF windows were glued to the panel such that the stress wave propagates sequentially through each of these materials.

The material properties are modeled using a physically motivated form given by Vinet et al. (1989). This form describes the pressure-density ($P - \rho$) response as

$$P(\rho) = 3B_0 \left(\frac{1-\eta}{\eta^2} \right) \exp \left(\frac{3}{2}(B'_0 - 1)(1-\eta) \right),$$

where $\eta = \sqrt[3]{\rho_0/\rho}$, ρ_0 is the initial density and B_0 and B'_0 are the bulk modulus and its pressure derivative at ambient conditions. From the computer experiment perspective, we will work with 6 inputs (3 EoS parameters (ρ, B_0, B'_0) and 3 experiment-specific parameters) and output velocity curves on a grid of 100 equidistant time points. As described in

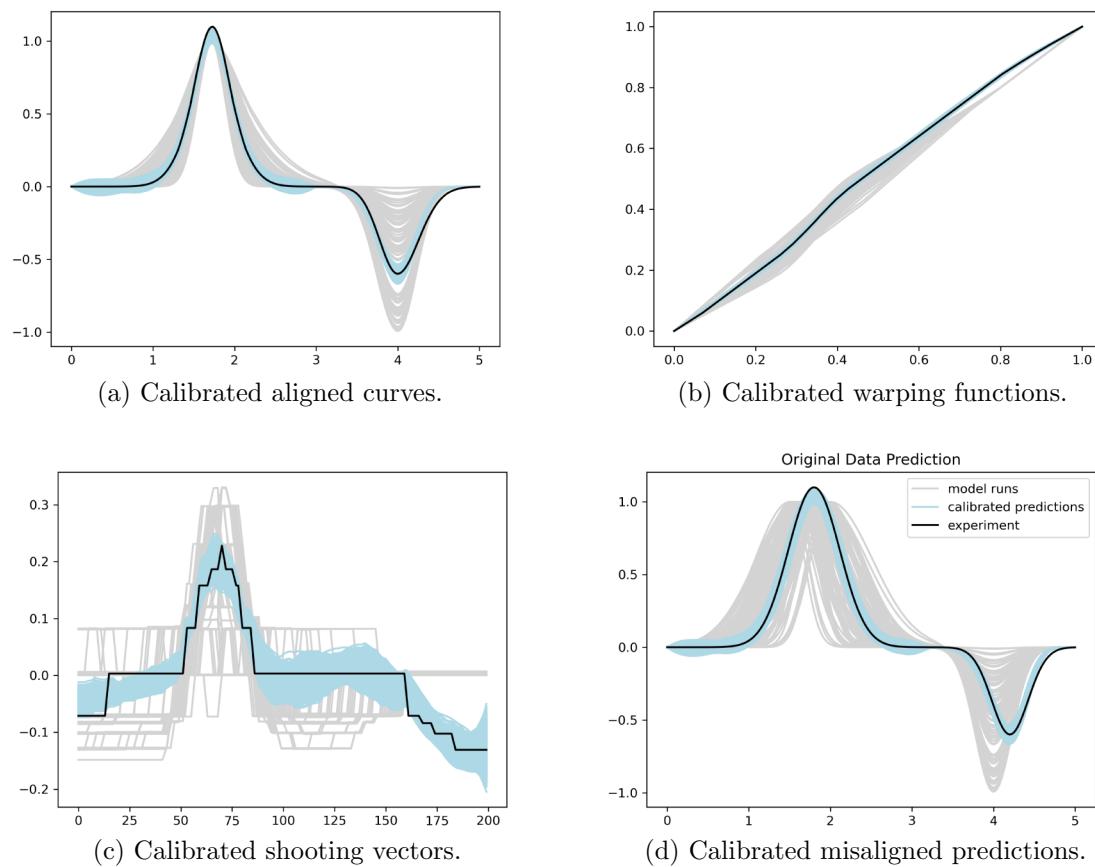


Figure 14. Posterior predictive samples after calibration of the simulated data for Example 2.

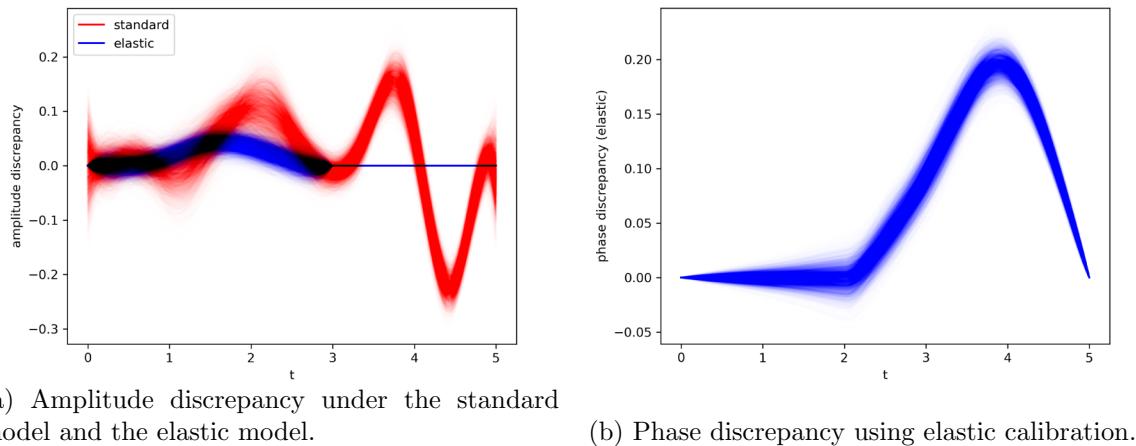


Figure 15. Comparison of standard and elastic discrepancy modeling for simulated example.

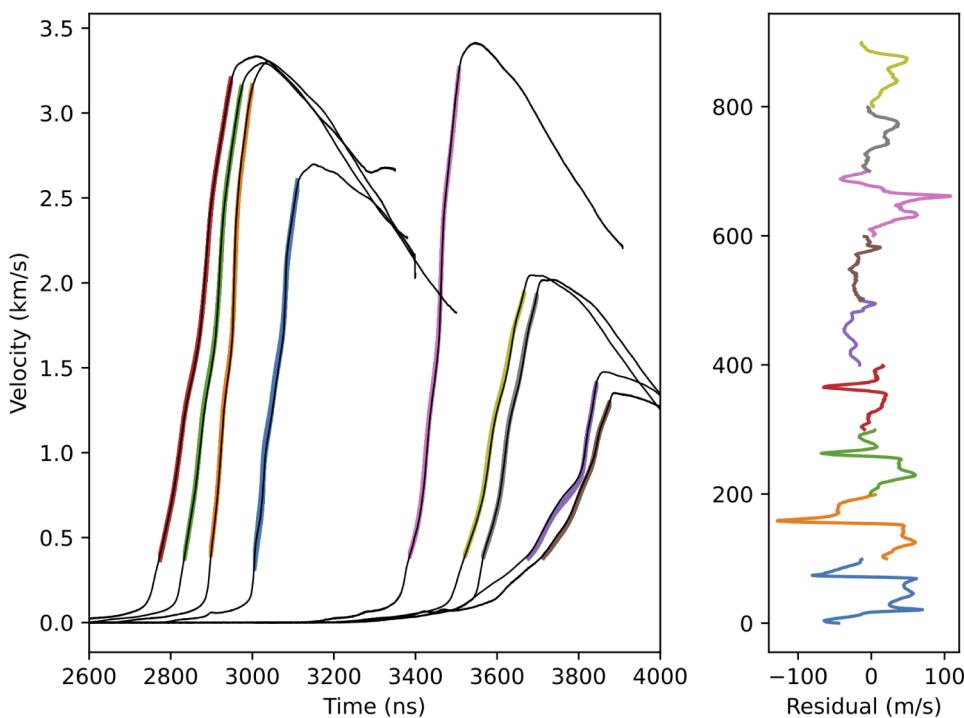


Figure 16. (Left) Experiment velocities of Ta shown in black compared with 95% prediction intervals from the elastic functional Bayesian calibration. (Right) Corresponding color-coded residuals (difference between experiment and calibration mean) for each experiment.

(Brown et al., 2014), the main goal is to provide inferences on the 3 EoS parameters with uncertainty quantification and to propagate these inferences to the Vinet model.

As was done in the simulated example, the computer model output was aligned to the experimental data, and an emulator was fitted to the aligned computer model and the corresponding shooting vectors. We then performed a modular elastic Bayesian model calibration. There are a total of 9 experiments.

Figure 16 presents the elastic Bayesian model calibration results for the Ta experiments. The black curves shown in the figure correspond to the experimental velocity curves for the 9 experiments. The shaded colored regions are the 95% prediction intervals that result from the elastic functional Bayesian calibration. The prediction intervals exhibit good agreement with each experimental curve. The residuals, defined as the difference between the experimental data and the calibrated predicted mean, are shown in the right panel of Figure 16. Each of the residuals are color coded to the corresponding experiment from the left panel. The full functional approach has tighter coverage of the experimental curves and smaller residual values compared to the resulting predictions and residuals that stem from the approach of Brown and Hund (2018).

Furthermore, Figure 17 presents a pairwise plot of the samples from the posterior distribution of the calibrated EoS parameters for Ta with the elastic approach. The posterior distributions are well concentrated within the parameter space, and the corresponding posterior medians compare to those found in Brown and Hund (2018) with a reduced uncertainty

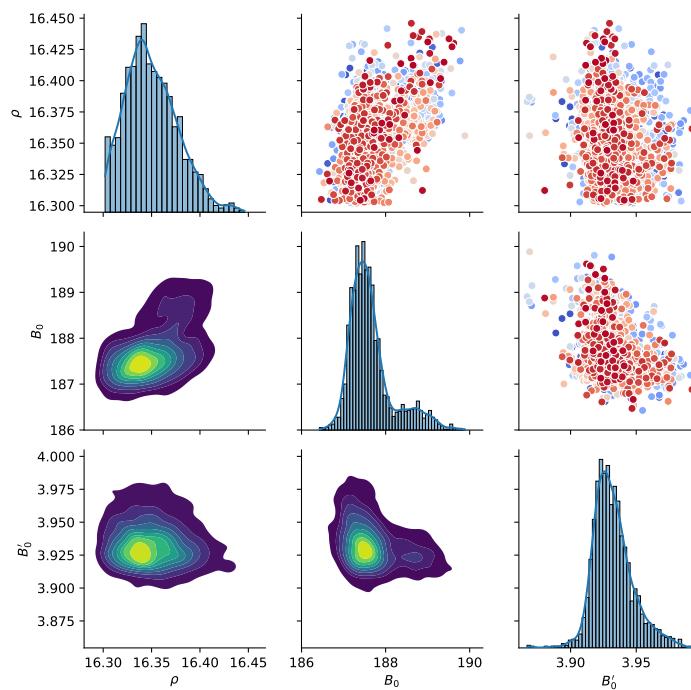


Figure 17. Pairs plot of the posterior densities of EoS parameters for Ta.

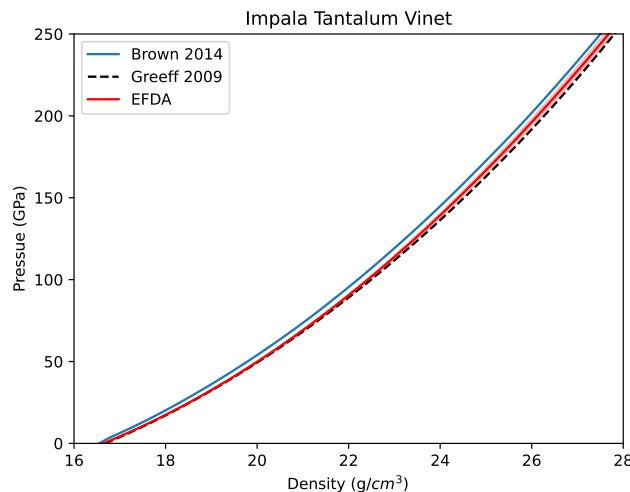


Figure 18. Calibrated Ta material response to the Vinet model compared with the analytic results in Brown et al. (2014) and the theoretical calculations in Greeff et al. (2009).

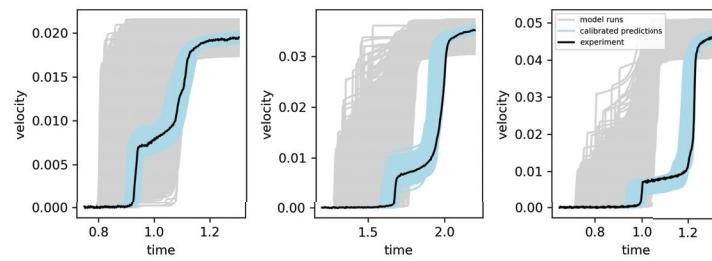
around the same parameter values. It should be remarked that the elastic FIDA calibration approach does not involve any likelihood scaling as in Brown and Hund (2018). To assess the calibration further, we compared the posterior estimates of the EoS to those reported in the literature for the Vinet model. Figure 18 presents a plot of the pressure versus density Vinet curve with a 95% credible interval generated from the elastic calibration shown in red. The blue curve corresponds to the loading path determined analytically by using the standard

techniques in the dynamic materials community as described in Brown et al. (2014). The dashed curve is the state-of-the-art theoretical calculation given by Greeff et al. (2009), which has been used to simulate these types of Ta experiments. The posterior estimate of the curve from the elastic calibration lies just above the theoretical calculation and below the analytical result, which corresponds with what is expected when using the analytic analysis (Kraus et al., 2016). The estimates of the physical parameters resulting from the elastic Bayesian model calibration are similar to those from previous work.

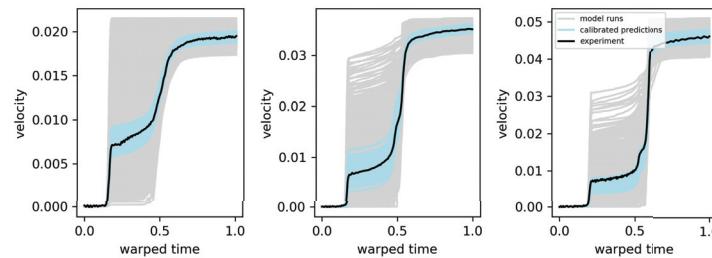
5.2. Flyer plate impact. Material strength characterizes how a material temporarily or permanently deforms as it experiences pressure. This is of interest in various areas of science and engineering, with applications in the aerospace, medical, and automotive industries (Gray et al., 2005). When experimentation is difficult, material strength models are an important tool for predicting how a material will react to pressure. One such model is the Johnson–Cook strength model (Johnson and Cook, 1983), for which the model behavior is dictated by a relatively small collection of material-specific physical parameters. To set these parameters, scientists and engineers rely on experimentation, hence calibrating the material strength model parameters to experimental measurements.

In this application, we consider the calibration of the Johnson–Cook material strength model for an Al alloy using a set of plate impact experiments. Walters et al. (2018) performed a Bayesian model calibration using plate impact experiments by reducing the velocimetry curves to a small set of features considered to be important to strength. We show how elastic Bayesian model calibration relies on the entire curve without using human-intensive feature engineering. Plate impact experiments achieve high pressure on a material sample by shooting a flyer at high velocity into the sample (plate). Lasers measure how the free surface of the sample moves as the shock wave moves through it. The result is a trace of the velocity of the free surface of the sample over time, called a velocimetry curve. From the computer experiment perspective, we will work with 11 inputs: 5 Johnson–Cook parameters plus 6 experiment-specific parameters (2 for each experiment) and the output velocimetry curves on a grid of 200 time points. Figure 19(a) shows measured velocimetry curves along with 1000 simulated velocimetry curves using different settings of the Johnson–Cook model and for 3 experiments. To obtain these simulations, the Johnson–Cook model is used within a larger hydrodynamics code that is expensive to evaluate. In this example, we do not use the observations as the alignment reference and instead use the first model run. Figure 19(b) shows the aligned functions, Figure 19(c) shows warping functions, and Figure 19(d) shows the shooting vectors corresponding to the measured and simulated curves. The misalignment of the experimental curves to the simulated is clear for the second and third experiments. The 2 right plots of Figure 19(a) show the experimental data to the right of all of the simulations. This is reflected in the warping functions in the 2 right plots of Figure 19(c), which push the experimental timings later than any of the simulations. In the shooting vectors of Figure 19(d), this is reflected with large positive values early in the vector and large negative values later in the vector, with varying patterns in the middle of the vector.

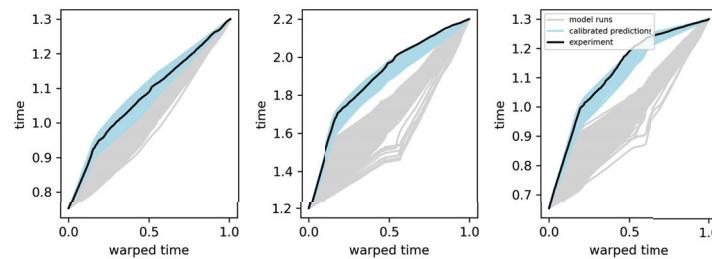
We use the aligned data and shooting vectors from the 1000 simulations to build an emulator and perform the elastic calibration. To allow for a time shift discrepancy for experiments 2 and 3, we use three piecewise constant and one piecewise linear basis functions in shooting



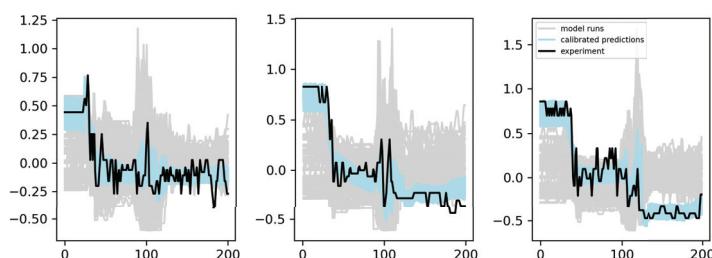
(a) Original Data and 1000 simulated velocimetry curves, with calibrated posterior predictions.



(b) Aligned measurement and simulation curves, with calibrated posterior predictions.



(c) Warping functions for measurement and simulation curves, with calibrated posterior predictions.



(d) Shooting vectors for measurement and simulation curves, with calibrated posterior predictions.

Figure 19. Posterior predictions from the elastic Bayesian model calibration for the flyer data example for the original data, the aligned curves, the warping functions, and the shooting vectors.

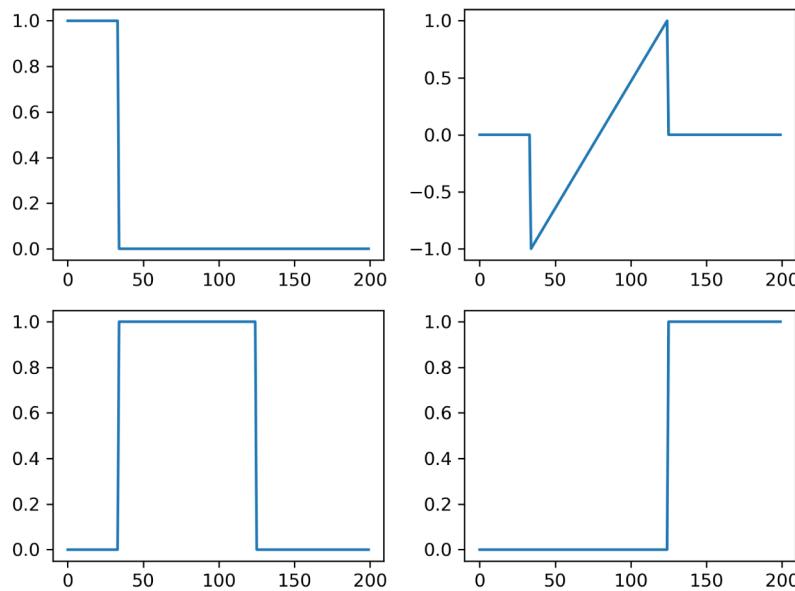


Figure 20. Basis functions used to capture a time shift discrepancy in shooting vector space for the flyer plate example.

vector space, as shown in Figure 20. More specifically, we parameterize $\delta_v(\mathbf{x}_i) = \mathbf{D}\beta(\mathbf{x}_i)$, where \mathbf{D} is the 200×4 matrix of basis functions and $\beta(\mathbf{x}_i)$ is a vector of 4 basis coefficients that can vary by experiment. In the approach of Higdon et al. (2008), $\beta_k(\cdot)$ would be assigned a Gaussian process prior so that the discrepancy could be predicted for a new experiment with settings \mathbf{x}^* . In our case, we are not interested in predicting the discrepancy at new experimental settings, so we specify a standard normal prior for each $\beta_k(\mathbf{x}_i)$. This indicates that we want to allow for a time shift but would prefer to make $\beta_k(\mathbf{x}_i) = 0$ if possible. The prior variance of $\beta_k(\mathbf{x}_i)$ can be modified in order to favor more or less discrepancy. The form of the basis functions in Figure 20 is carefully chosen to be constant in domains where the aligned data are roughly constant and a linear function in the region where the aligned data are most active. Specifically, we think of the aligned curves as having three parts: (1) a nearly constant region at the beginning; (2) a quick jump in velocity, a plateau, and another quick jump; and (3) a final plateau. We choose our basis functions such that we allow for constant discrepancy shifts in the shooting vectors in the (1) and (3) regions and linear shifts in the (2) region. We largely arrived at this combination by trial and error made sensible in hindsight. Recall that the transformation of shooting vector to warping function involves the exponential map and an integral. Portions of the shooting vector that are positive indicate faster-than-identity warping, while negative values indicate slower-than-identity warping. The warping functions are constrained to start at 0 and end at 1, so the constant shifts in shooting vectors at the first and last plateaus allow for more extreme change in timing there, where it does not have much effect, to compensate for the linear portion of the shooting vector discrepancy, which has the more noticeable effect.

Figure 19 presents the posterior predictive samples of the velocimetry curves after calibration shown in blue. In a similar way, Figure 19(b), (c), and (d) show the posterior predictive

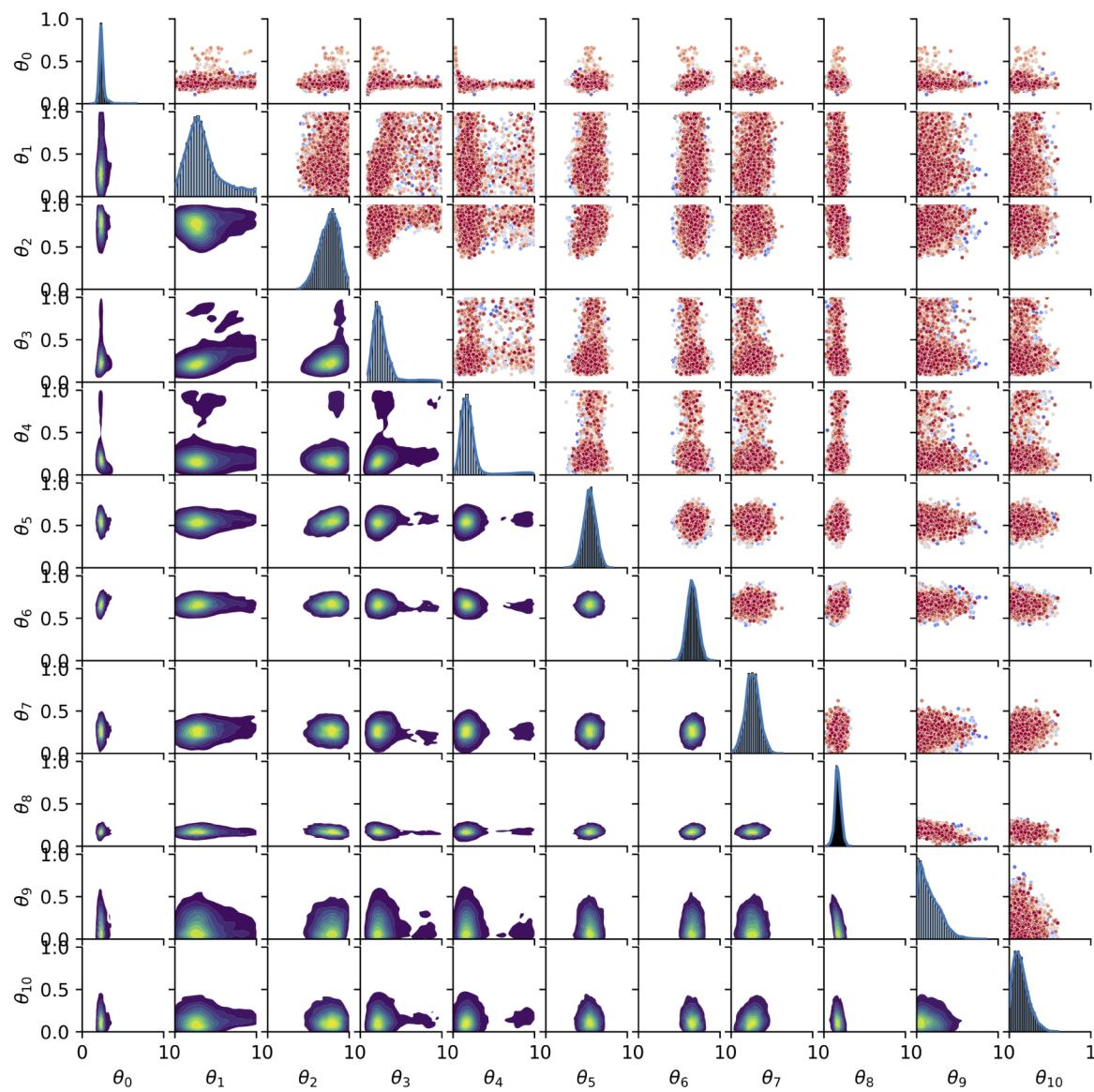


Figure 21. Pairwise plot and marginal densities for the posterior distribution of the 11 calibrated parameters for the flyer plate experiment.

samples of the aligned curves, warping functions, and shooting vectors, respectively. In all of these cases, the predictive samples cover the experimental data well. Furthermore, Figure 21 presents a pairs plot of the posterior distribution of the 11 calibrated parameters with diagonal elements representing marginal distributions while the lower diagonal shows bivariate contours and the upper diagonal shows pair plots of the samples. Compared to [Walters et al. \(2018\)](#), we are able to use the entirety of the functional data, while their approach used a few hand-selected features.

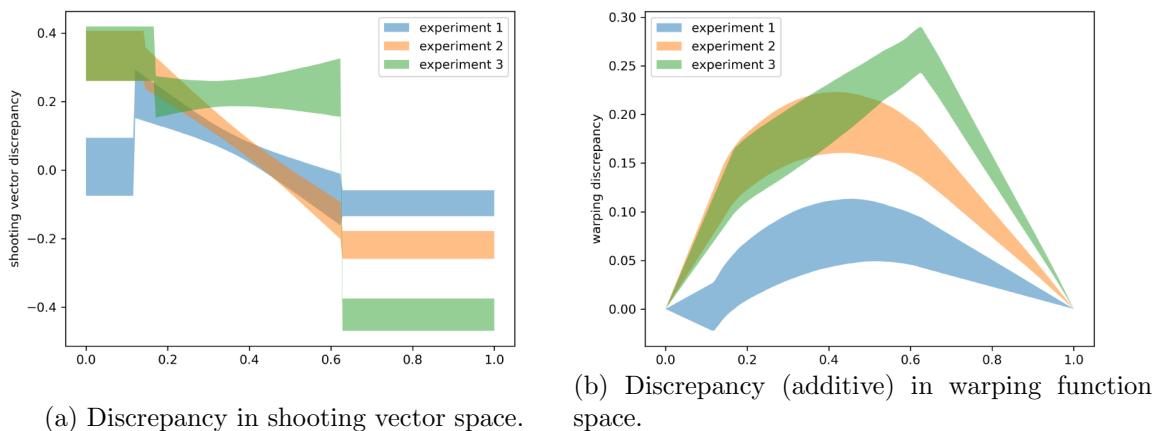


Figure 22. Estimated posterior discrepancy per experiment for the Flyer plate impact calibration.

Figure 22(a) shows the posterior distribution of $\delta_v(\mathbf{x}_1)$, $\delta_v(\mathbf{x}_2)$, and $\delta_v(\mathbf{x}_3)$, the shooting vector discrepancies. Figure 22(b) shows the difference between two sets of warping functions—those that include the discrepancy in Figure 22(a) and those that do not—which illustrates the type of shift that these discrepancies induce.

The code to reproduce this analysis is available at https://github.com/lanl/impala/blob/master/examples/shpb-flyer-Al5083_pooled.py.

6. Conclusion. Model calibration for functional responses is typically more difficult than in more traditional settings. Adjusting input parameters and discrepancy so that model output matches experimental output is seldom trivial, but can be even more challenging for functional data when altering input parameters results in amplitude and phase variability in the responses. Traditional methods ignore these aspects, which are unique to models with functional responses, putting them at risk for higher levels of bias and lower levels of efficiency. In this paper, we develop methods to handle amplitude and phase variability in a systematic way, resulting in more efficient and more accurate estimates of the model parameters. The improvements in these estimators are achieved through better handling of the functional responses as opposed to collecting more data.

The elastic Bayesian model calibration procedure presented here uses information from both the amplitude and phase space to calibrate the parameters, in contrast with more common calibration methods, which would only look at the amplitude space. The benefits of this approach are demonstrated on a toy problem where amplitude and phase variability are present. The toy problem shows that the elastic Bayesian model calibration approach results in superior estimation of the model parameters and predicted functional responses. Information about the amount of warping necessary to align the functions provides an additional indirect benefit because it provides valuable insight into the model's input parameters. Standard methods, which do not handle the phase and amplitude variability separately, may result in predicted functional responses that do not fit the data well and suggest much larger uncertainty in calibration parameters. These results make a strong case for the elastic modeling approach because most functional data have some misalignment and phase discrepancy.

Applying the method to the Z-machine data and flyer-plate data generated similar results to previous studies without the need of likelihood scaling as in Brown and Hund (2018) or feature selection as in Walters et al. (2018). The elastic Bayesian calibration model yields credible intervals with good agreement of the experimental data using a principled approach. From a theoretical perspective, the elastic Bayesian calibration approach is satisfying in that it treats the model's output as functional throughout the calibration procedure.

Potential future work may investigate some of the additional modeling assumptions made in the elastic calibration approach, such as the independence assumptions in the likelihood and the treatment of the warping decomposition uncertainty.

REFERENCES

- M. J. BAYARRI, J. O. BERGER, J. CAFEO, G. GARCIA-DONATO, F. LIU, J. PALOMO, R. J. PARTHASARATHY, R. PAULO, J. SACKS, AND D. WALSH (2007a), *Computer model validation with functional output*, Ann. Statist., 35, pp. 1874–1906, <https://doi.org/10.1214/009053607000000163>.
- M. J. BAYARRI, J. O. BERGER, AND F. LIU (2009), *Modularization in Bayesian analysis, with emphasis on analysis of computer models*, Bayesian Anal., 4, pp. 119–150, <https://doi.org/10.1214/09-BA404>.
- M. J. BAYARRI, J. O. BERGER, R. PAULO, J. SACKS, J. A. CAFEO, J. CAVENDISH, C. LIN, AND J. TU (2007b), *A framework for validation of computer models*, Technometrics, 49, pp. 138–154, <https://doi.org/10.1198/004017007000000092>.
- J. M. BOTELER AND D. P. DANDEKAR (2006), *Dynamic response of two strain-hardened aluminum alloys*, J. Appl. Phys., 100, 054902, <https://doi.org/10.1063/1.2336492>.
- J. BROWN AND L. HUND (2018), *Estimating material properties under extreme conditions by using Bayesian model calibration with functional outputs*, J. R. Stat. Soc. Ser. C, 67, pp. 1023–1045, <https://doi.org/10.1111/rssc.12273>.
- J. L. BROWN, C. S. ALEXANDER, J. R. ASAY, T. J. VOGLER, D. H. DOLAN, AND J. L. BELOF (2014), *Flow strength of tantalum under ramp compression to 250 GPa*, J. Appl. Phys., 115, 043530, <https://doi.org/10.1063/1.4863463>.
- J. BRYNJARSDÓTTIR AND A. O'HAGAN (2014), *Learning about physical parameters: The importance of model discrepancy*, Inverse Problems, 30, 114007, <https://doi.org/10.1088/0266-5611/30/11/114007>.
- W. CHENG, I. L. DRYDEN, AND X. HUANG (2016), *Bayesian registration of functions and curves*, Bayesian Anal., 11, pp. 447–475, <https://doi.org/10.1214/15-BA957>.
- G. COLLINS, D. FRANCOM, AND K. RUMSEY (2024), *Bayesian projection pursuit regression*, Stat. Comput., 34, 29, <https://doi.org/10.1007/s11222-023-10334-z>.
- D. FRANCOM, B. SANZO, V. BULAEVSKAYA, D. LUCAS, AND M. SIMPSON (2019), *Inferring atmospheric release characteristics in a large computer experiment using Bayesian adaptive splines*, J. Amer. Statist. Assoc., 114, pp. 1450–1465, <https://doi.org/10.1080/01621459.2018.1562933>.
- D. FRANCOM, B. SANZO, AND A. KUPRESANIN (2022), *Landmark-warped emulators for models with misaligned functional response*, SIAM/ASA J. Uncertain. Quantif., 10, pp. 125–150, <https://doi.org/10.1137/20M135279X>.
- D. FRANCOM, B. SANZO, A. KUPRESANIN, AND G. JOHANNESSEN (2018), *Sensitivity analysis and emulation for functional data using Bayesian adaptive splines*, Stat. Sinica, 28, pp. 791–816, <https://doi.org/10.5705/ss.202016.0130>.
- D. FRANCOM, P. TRUBEY, ET AL. (2023), *LNL/impala*, <https://github.com/lnl/impala>.
- J. GATTIKER, N. KLEIN, E. LAWRENCE, AND G. HUTCHINGS (2023), *LNL/SEPIA*, <https://doi.org/10.5281/zenodo.4048801>.
- G. T. R. GRAY, P. J. MAUDLIN, L. M. HULL, Q. K. ZUO, AND S. CHEN (2005), *Predicting material strength, damage, and fracture the synergy between experiment and modeling*, J. Failure Anal. Prevention, 5, pp. 7–17, <https://doi.org/10.1361/15477020523725>.
- C. W. GREEFF, S. P. RUDIN, S. D. CROCKETT, AND J. M. WILLS (2009), *The cold equation of state of tantalum*, in AIP Conf. Proc., American Institute of Physics, Melville, NY, pp. 681–684, <https://doi.org/10.1063/1.3295231>.

- M. GU AND J. O. BERGER (2016), *Parallel partial Gaussian process emulation for computer models with massive output*, Ann. Appl. Statist., 10, pp. 1317–1347, <https://doi.org/10.1214/16-AOAS934>.
- M. GU AND L. WANG (2018), *Scaled Gaussian stochastic process for computer model calibration and prediction*, SIAM/ASA J. Uncertain. Quantif., 6, pp. 1555–1583, <https://doi.org/10.1137/17M1159890>.
- Y. GUAN, C. SAMPSON, J. DEREK TUCKER, W. CHANG, A. MONDAL, M. HARAN, AND D. SULSKY (2019), *Computer model calibration based on image warping metrics: An application for sea ice deformation*, J. Agric. Biol. Environ. Stat., 24, pp. 444–463, <https://doi.org/10.1007/s13253-019-00353-7>.
- D. HIGDON, J. GATTIKER, B. WILLIAMS, AND M. RIGHTLEY (2008), *Computer model calibration using high-dimensional output*, J. Amer. Stat. Assoc., 103, pp. 570–583, <https://doi.org/10.1198/016214507000000888>.
- D. HIGDON, M. KENNEDY, J. C. CAVENDISH, J. A. CAFEO, AND R. D. RYNE (2004), *Combining field data and computer simulations for calibration and prediction*, SIAM J. Sci. Comput., 26, pp. 448–466, <https://doi.org/10.1137/S1064827503426693>.
- L. HORVATH AND P. KOKOSZKA (2012), *Inference for Functional Data with Applications*, Springer, Cham.
- G. HUTCHINGS, B. SANSÓ, J. GATTIKER, D. FRANCOM, AND D. PASQUALINI (2023), *Comparing emulation methods for a high-resolution storm surge model*, Environmetrics, 34, e2796, <https://doi.org/10.1002/env.2796>.
- G. R. JOHNSON AND W. H. COOK (1983), *A constitutive model and data for metals subjected to large strains, high strain rates, and high temperatures*, in Proceedings of the 7th International Symposium on Ballistics, pp. 541–547.
- M. C. KENNEDY AND A. O'HAGAN (2001), *Bayesian calibration of computer models*, J. R. Stat. Soc. Ser. B Stat. Methodol., 63, pp. 425–464, <https://doi.org/10.1111/1467-9868.00294>.
- W. KLEIBER, S. R. SAIN, AND M. J. WILTBERGER (2014), *Model calibration via deformation*, SIAM/ASA J. Uncertain. Quantif., 2, pp. 545–563, <https://doi.org/10.1137/130935367>.
- R. G. KRAUS, J.-P. DAVIS, C. T. SEAGLE, D. E. FRATANDUONO, D. C. SWIFT, J. L. BROWN, AND J. H. EGGERT (2016), *Dynamic compression of copper to over 450 GPa: A high-pressure standard*, Phys. Rev. B, 93, 134105, <https://doi.org/10.1103/PhysRevB.93.134105>.
- R. W. LEMKE, M. D. KNUDSON, D. E. BLISS, K. COCHRANE, J. DAVIS, A. A. GIUNTA, H. C. HARJES, AND S. A. SLUTZ (2005), *Magnetically accelerated, ultrahigh velocity flyer plates for shock wave experiments*, J. Appl. Phys., 98, 073530, <https://doi.org/10.1063/1.2084316>.
- Y. LU, R. HERBEI, AND S. KURTEK (2017), *Bayesian registration of functions with a Gaussian process prior*, J. Comput. Graph. Stat., 26, pp. 894–904, <https://doi.org/10.1080/10618600.2017.1336444>.
- J. S. MARRON, J. O. RAMSAY, L. M. SANGALLI, AND A. SRIVASTAVA (2015), *Functional data analysis of amplitude and phase variation*, Statist. Sci., 30, pp. 468–484, <https://doi.org/10.1214/15-STS524>.
- M. PLUMMER (2015), *Cuts in Bayesian graphical models*, Stat. Comput., 25, pp. 37–43, <https://doi.org/10.1007/s11222-014-9503-z>.
- J. O. RAMSAY AND X. LI (1998), *Curve registration*, J. R. Stat. Soc. Ser. B, 60, pp. 351–363, <https://doi.org/10.1111/1467-9868.00129>.
- J. O. RAMSAY AND B. W. SILVERMAN (2005), *Functional Data Analysis*, Springer, Cham.
- J. SACKS, W. J. WELCH, T. J. MITCHELL, AND H. P. WYNN (1989), *Design and analysis of computer experiments*, Statist. Sci., 4, pp. 409–423, <https://doi.org/10.1214/ss/1177012413>.
- M. E. SAVAGE, L. F. BENNETT, D. E. BLISS, W. T. CLARK, R. S. COATS, J. M. ELIZONDO, K. R. LECHIEN, H. C. HARJES, J. M. LEHR, J. E. MAENCHEN, D. H. McDANIE, M. F. PASIK, T. D. POINTON, A. C. OWEN, D. B. SEIDEL, D. L. SMITH, B. S. STOLTZFUS, K. W. STRUVE, W. A. STYGAR, L. K. WARNE, J. R. WOODWORTH, C. W. MENDEL, K. R. PRESTWICH, R. W. SHOUP, D. L. JOHNSON, J. P. CORLEY, K. C. HODGE, T. C. WAGONER, AND P. E. WAKELAND (2007), *An overview of pulse compression and power flow in the upgraded Z pulsed power driver*, in 2007 16th IEEE International Pulsed Power Conference Vol. 2, IEEE, pp. 979–984, <https://doi.org/10.1109/PPPS.2007.4652354>.
- P. SODERLIND AND J. A. MORIARTY (1998), *First-principles theory of Ta up to 10 Mbar pressure: Structural and mechanical properties*, Phys. Rev. B, 57, pp. 10340–10350, <https://doi.org/10.1103/PhysRevB.57.10340>.
- A. SRIVASTAVA AND E. KLASSEN (2016), *Functional and Shape Data Analysis*, Springer, Cham.
- A. SRIVASTAVA, W. WU, S. KURTEK, E. KLASSEN, AND J. S. MARRON (2011), *Registration of Functional Data Using Fisher-Rao Metric*, preprint, arXiv:1103.3817v2 [math.ST], <http://arxiv.org/abs/1103.3817v2>.

- J. D. TUCKER (2023), *fdasrsf*, https://github.com/jdtuck/fdasrsf_python.
- J. D. TUCKER, W. WU, AND A. SRIVASTAVA (2013), *Generative models for functional data using phase and amplitude separation*, *Comput. Stat. Data Anal.*, 61, pp. 50–66, <https://doi.org/10.1016/j.csda.2012.12.001>.
- P. VINET, J. ROSE, J. FERRANTE, AND J. SMITH (1989), *Universal features of the equation of state of solids*, *J. Phys. Condens. Matt.*, 1, pp. 1941–1963, <https://doi.org/10.1088/0953-8984/1/11/002>.
- D. J. WALTERS, A. BISWAS, E. C. LAWRENCE, D. C. FRANCOM, D. J. LUSCHER, D. A. FREDENBURG, K. R. MORAN, C. M. SWEENEY, R. L. SANDBERG, J. P. AHRENS, AND C. A. BOLME (2018), *Bayesian calibration of strength parameters using hydrocode simulations of symmetric impact shock experiments of Al-5083*, *J. Appl. Phys.*, 124, 205105, <https://doi.org/10.1063/1.5051442>.
- B. WILLIAMS, D. HIGDON, J. GATTIKER, L. MOORE, M. MCKAY, AND S. KELLER-MCNULTY (2006), *Combining experimental data and computer simulations, with an application to flyer plate experiments*, *Bayesian Anal.*, 1, pp. 765–792, <https://doi.org/10.1214/06-BA125>.
- W. WU AND A. SRIVASTAVA (2011), *An information-geometric framework for statistical inferences in the neural spike train space*, *J. Comput. Neurosci.*, 31, pp. 725–748, <https://doi.org/10.1007/s10827-011-0336-x>.