LA-UR-17-27338 (Accepted Manuscript)

Statistical inference of empirical constituents in partitioned analysis from integral-effect experiments

Stevens, Garrison Nicole
Atamturktur, Sez
Brown, D. Andrew
Williams, Brian J.
Unal, Cetin

To be published in: Engineering Computations

DOI to publisher's version: 10.1108/EC-07-2016-0264

Permalink to record: http://permalink.lanl.gov/object/view?what=info:lanl-repo/lareport/LA-UR-17-27338

Statistical Inference of Empirical Constituents in Partitioned Analysis from Integral-effect Experiments: An Application in Thermo-mechanical Coupling

G.N. Stevens[1], S. Atamturktur[2], D.A. Brown[3], B.J. Williams[1], and C. Unal[1]

[1]*Los Alamos National Laboratory, Los Alamos, NM 87544*
[2]*Glenn Department of Civil Engineering, Clemson University, Clemson, SC 29632*
[3]*Mathematical Sciences, Clemson University, Clemson, SC 29632*

1 Introduction

Rapid advancements in parallel computing over the last two decades have enabled simulations of complex, coupled systems through partitioning. In partitioned analysis, independently developed *constituent* models communicate, representing dependencies between multiple physical phenomena that occur in the full system. Figure 1 schematically demonstrates a coupled system with two constituent models, each resolving different physical behavior. In this figure, the constituent model, denoted as the "consumer," relies upon some input parameter that is being provided by the constituent model acting as a "feeder". The role of the feeder model is to map operating conditions (i.e. those that are stimulating the process) to consumer inputs, thus providing functional inputs to the consumer model*. Problems arise if the feeder model cannot be built–a challenge that is prevalent for highly complex systems in extreme operational conditions that push the limits of our understanding of underlying physical behavior. Often, these are also the situations where separate-effect experiments isolating the physical phenomena are not available; meaning that experimentally determining the unknown constituent behavior is not possible (Bauer and Holland, 1995; Unal et al., 2013), and that integral-effect experiments that reflect the behavior of the complete system tend to be the only available observations. In this paper, the authors advocate for the usefulness of integral-effect experiments in furthering a model developer's knowledge of the physics principles governing the system behavior of interest.

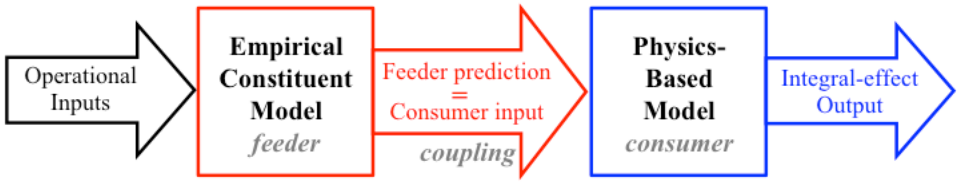


Figure 1. Empirical representation of constituent model to simulate a coupled system through partitioned analysis.

Thermo-mechanical coupling where the thermal model (feeder) influences the behavior of the mechanical model (consumer) is one such multi-physics application that plays a role in a wide range of engineering applications, for example, shape memory alloys that return to their original

* The feeder/consumer constituent relationship shown here represents a weakly coupled system. The case of strongly coupled models, where feedback between constituents creates a loop that must be solved iteratively, is beyond the scope of this paper.

undeformed shapes when heated (Dunić et al., 2012), rock fracture behavior where temperature changes in a rock mass influence the elastic properties (Auricchio et al., 2007; Shen et al., 2014), and nuclear reactors where the high temperature loads experienced during irradiation change the material density, gas production, and thus mechanical behavior of fuel materials (Williamson et al., 2012; Galloway et al., 2015). Modeling of such coupled systems, where material properties are dependent upon temperature, often results in the scenario shown previously in Figure 1, as the precise effects of temperature on material properties in highly complex systems may be unknown and not possible to observe experimentally in an isolated manner. For example, changes in thermal conductivity of metallic fuels due to irradiation in a nuclear reactor is a physical process which is neither theoretically well understood nor possible to isolate in experimental measurements, yet it is known to be influential in the reactor behavior (Bauer and Holland, 1995). In the absence of a feeder thermal constituent model, the consumer mechanical model would be bound to an incomplete (and thus, inaccurate) representation of reality, as the temperature dependency of its parameters are not accounted for. As a result, the mechanical constituent model, if calibrated with one dataset at a given temperature, would be unable to produce validated predictions of another dataset collected at a different temperature (as in the case documented in Jackson et al., (2014)).

This paper presents a statistical inference method in which integral-effect experiments as well as the available mechanical model are used to empirically infer a mathematical representation for the thermal constituent model that is otherwise unattainable (i.e. empirical constituent model in Figure 1). The paper is organized as follows. The inverse analysis methodology is presented in Section 2 followed by a conceptual demonstration in Section 3. Section 4 focuses on a thermo-mechanical application where different temperatures experienced during system operations change a metallic material's crystal properties, and therefore its mechanical behavior. The proposed methodology is applied in a case study of 5182 aluminum, which has an experimentally demonstrated dependence of the material behavior on both temperature and strain rate, for which constituent models are currently unavailable. Section 5 concludes the findings and presents a path forward for future work.

2 Methodology for Inferring Coupling Relationships through Inverse Analysis

2.1 Integral-Effect Experiment-Based Inference

The proposed approach seeks to learn the relationship of physics-based consumer constituent inputs to operational states. Such relationships would be represented as predictions of the feeder constituent model if it were to be available but instead become uncertain input parameters of the consumer model when the feeder model cannot be obtained (recall Figure 1). Of course, the consumer model could also have parameters that are not dependent upon the feeder model. Thus, two sets of uncertain input parameters must be considered for the consumer constituent: *constant parameters,* $\boldsymbol{\theta_c}$ that are not reliant upon other operational conditions and *functional parameters*, $\boldsymbol{\theta_f}$ that are reliant upon operational conditions (Figure 2). Similarly, the physics-based consumer model may also have two different types of operational parameters: parameters that the coupling depends upon (i.e. temperature in the thermo-mechanical example) that are denoted as *dependent*

operational parameters, $\boldsymbol{x_f}$ and the parameters that the coupling does not depend that are be denoted as *independent operational parameters*, $\boldsymbol{x_c}$ (Figure 2).

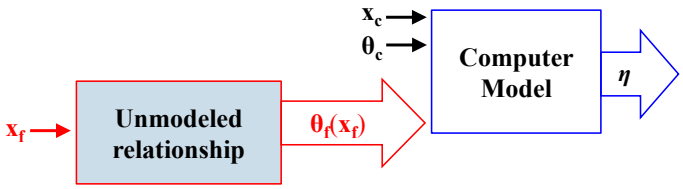


Figure 2. Variables of interest for the feeder and consumer models.

Ultimately, the goal is to represent predictions of the missing feeder constituent model as a functional parameter of the consumer constituent, such that $\boldsymbol{\theta_f \equiv \theta_f(x_f)}$ (Brown and Atamturktur, 2018). This is accomplished through inverse analysis using integral-effect experimental observations, y, as our knowledge of the desired full-system behavior with uncertainty in the form of experimental error, ε (Eq. 1). As shown in Eq. 1, where n is the discrete number of settings for experimental data points, the consumer constituent model, η, effectively becomes a semi-empirical partitioned representation of the coupled system once $\boldsymbol{\theta_f(x_f)}$ is identified and included as an empirically derived feeder constituent.

$$y(\mathbf{x}_f^i) = \eta(\mathbf{x}_f^i, \boldsymbol{\theta}_f(\mathbf{x}_f^i)) + \varepsilon(\mathbf{x}_f^i) \quad \text{where } i = 1, 2, \ldots, n \qquad \text{(Eq. 1)}$$

Additionally, the physics-based consumer constituent model may have its own parametric uncertainty as well as other operational states on which the feeder constituent does not depend. These uncertain physical parameters, $\boldsymbol{\theta_c}$ and the feeder constituent model, $\boldsymbol{\theta_f(x_f)}$ can be inferred simultaneously as shown in Eq. 2, where m indicates the discrete points at which experimental data is available in the control dimension of x_c. Consideration of the consumer model uncertainties concurrently with the empirical model inference will reduce the risk of unwarranted compensations that may otherwise degrade the realism of the empirical model. Inclusion of experiments across various control dimensions may also provide additional data points to inform the functional parameter inference.

$$y(\mathbf{x}_f^i, \mathbf{x}_c^j) = \eta(\mathbf{x}_f^i, \mathbf{x}_c^j, \boldsymbol{\theta}_f(\mathbf{x}_f^i), \boldsymbol{\theta}_c) + \varepsilon(\mathbf{x}_f^i, \mathbf{x}_c^j) \quad \text{where } i=1, 2, \ldots, n \quad \text{and} \quad j=1, 2, \ldots, m \qquad \text{(Eq. 2)}$$

Knowing that the experiments are only a best estimate of the true response, we recognize experimental error at every sample point as $\boldsymbol{\varepsilon(x_f^i, x_c^j)}$. Experimental error is assumed to be independent and identically normally distributed over all inputs with precision, λ, and is modeled accordingly $\varepsilon_i \sim N(0, \lambda \boldsymbol{I})$.

2.2 Gaussian Process Representation of Functional Parameters

Any available prior knowledge concerning the functional form of the empirical constituent $\boldsymbol{\theta_f(x_f)}$, perhaps in the form of expert opinion or parametric trends observed in predictions from legacy codes, can be incorporated into inference. For example, Atamturktur et al. (2015) determined a functional input parameter by calibrating coefficients of a pre-defined function selected by expert opinion. If prior knowledge regarding $\boldsymbol{\theta_f(x_f)}$ is not available, however, a Gaussian process (GP)

can be used without imposing restrictions on the functional form of $\boldsymbol{\theta_f(x_f)}$ (MacKay, 1998; Neal, 1998; Kennedy and O'Hagan, 2001; Bastos and O'Hagan, 2009). In the case that a model has more than one missing constituent, a separate GP and its associated hyperparameters would need to be inferred for each.

A GP is a stochastic process used to relate points along some control dimension, x, where the nature of the relationship is specified by a mean and covariance function (Eq. 3). The GP applied herein adopts a squared exponential correlation function to relate points along the functional parameter, θ_f, with respect to the dependent operational parameter, x_f. All input parameter values of the model are scaled to a unit hypercube for the inverse analysis, justifying the selection of a mean of 0.5 for the GP. The squared exponential correlation function (Eq. 4) is one of the most commonly implemented, as its realizations are smooth, infinitely differentiable functions capable of representing a wide range of continuous functions (Rasmussen and Williams, 2006; Swiler, 2006). The dimensionality of the functional control parameter is represented by d.

$$\theta_f \sim GP\left(0.5, \lambda_{\theta_f}^{-1} R(x_f, x_f')\right) \quad \text{(Eq. 3)}$$

$$R(x_f, x_f') = \exp\left\{\sum_{k=1}^{d} -4\gamma_{\theta_{f_k}}\left(x_{f_k} - x_{f_k}'\right)^2\right\} \text{ where } k=1,2,\ldots,d \quad \text{(Eq. 4)}$$

For simplicity, the remainder of this discussion assumes that d is equal to one. The GP given in Eq. 4 has two hyperparameters, λ_{θ_f}, the precision parameter which controls the magnitude of variations in θ_f, and γ_{θ_f}, the smoothness parameter which defines the inverse length scale of x_f across which θ_f is expected to be correlated, meaning that a larger value of γ_{θ_f} results in smaller correlations across x_f (Williams and Rasmussen, 1996). These hyperparameters control the functional form of the GP for $\theta_f(x_f)$ and thus must be learned from the data. To do so, prior distributions are assumed for the two GP hyperparameters, λ_{θ_f} and $\rho_{\theta_f} = e^{-\gamma_{\theta_f}}$. Herein, a beta hyperprior (Eq. 5) is applied to ρ_{θ_f}, where the shape parameter b_ρ can be used to enforce smoothness by concentrating the distribution near one. A gamma hyperprior (Eq. 6) is applied to λ_{θ_f}.

$$\rho_{\theta_f} \sim \text{Beta}(1, b_\rho),\ b_\rho > 0 \quad \text{(Eq. 5)}$$

$$\lambda_{\theta_f} \sim \text{Ga}(a_\lambda, b_\lambda),\ a_\lambda, b_\lambda > 0 \quad \text{(Eq. 6)}$$

If the sampled x_f values are identical (or near-identical), the correlation matrix associated with the GP might have problems with matrix inversion due to near-singularity, which is commonly avoided by the addition of some nugget parameter to the diagonal of the correlation matrix (Sacks et al., 1989; Neal, 1998; Santner et al., 2003). Following Ranjan et al. (2011), here the nugget δ is determined with $\delta = \max\left(\lambda_N(\kappa(\boldsymbol{R}) - e^{20})\left(\kappa(\boldsymbol{R})\right)^{-1}(e^{20} - 1)^{-1},\ 0\right)$, where λ_N is the largest eigenvalue of the correlation matrix $\boldsymbol{R}$, $\kappa(\boldsymbol{R})$ is the condition number of $\boldsymbol{R}$, and e^{20} is the threshold on the condition number above which we consider $\boldsymbol{R}$ to be numerically unstable.

2.3 Bayesian Inference for Inverse Analysis

The Bayesian solution to the inverse analysis in Eq. 2 infers the posterior distribution of parameters conditioned upon the experimental measurements (Eq. 7). The experimental data, $y(x_f, x_c)$ are standardized to have a mean of zero and a standard deviation of one.

$$P\left(\boldsymbol{\theta_c}, \boldsymbol{\theta_f}, \boldsymbol{\lambda_{\theta_f}}, \boldsymbol{\rho_{\theta_f}} | \boldsymbol{y}(\boldsymbol{x_f}, \boldsymbol{x_c})\right) \propto L\left(\boldsymbol{y}(\boldsymbol{x_f}, \boldsymbol{x_c}) | \boldsymbol{\theta_c}, \boldsymbol{\theta_f}, \boldsymbol{\lambda_\theta}, \boldsymbol{\rho_\theta}\right) \times P(\boldsymbol{\theta_f}) \times P(\boldsymbol{\theta_c}) \times P\left(\boldsymbol{\lambda_{\theta_f}}\right) \times P\left(\boldsymbol{\rho_{\theta_f}}\right) \quad \text{(Eq. 7)}$$

The posterior distribution given in Equation 7 can be obtained through Markov chain Monte Carlo sampling of the parameter space, requiring hundreds to thousands of runs depending on the dimensionality of the problem (Gilks et al., 1995; Higdon et al., 2004). In this study, Gibbs sampling, a specific case of the Metropolis-Hastings algorithm (Metropolis et al., 1953; Hastings, 1970; Geman and Geman, 1984), is used to sample the approximate posterior, where each step of the sampling is taken from a full conditional distribution of one parameter. Gibbs sampling cycles through each parameter individually to explore the domain (Gelfand and Smith, 1990). In the Gibbs sequence, each time hyperparameters of the GP for $\boldsymbol{\theta_f}$ are updated, the correlation matrix needs to be inverted, an operation of complexity n^3, where n is the size of the correlation function (i.e. number of data points in x_f), while full Gibbs scan for model parameters $\boldsymbol{\theta_f}$ has complexity n^2. Noticing that additional Gibbs samples taken between hyperparameter updates adds only marginally to the computational cost, Neal (1998) suggested conducting subiterations (i.e. additional Metropolis steps) for vector-valued variables within the Gibbs sampler to potentially increase the rate of convergence. Herein, we implement such subiterations on θ_f. In each full iteration all GP hyperparameters are sampled once. From the current set of GP hyperparameters, multiple realizations of θ_f (x_f) are then drawn by subiterations. In other words, for a given Gibbs sample the GP covariance is fixed while the functional parameter defined by said GP is updated by Metropolis steps. The effects of such additional Gibbs samples are discussed in Section 3.2.

3 Proof-of-Concept Demonstration

This section presents an academic proof-of-concept example with synthetically generated integral-effect experiments. Synthetically generating experiments allows control of the true value of the empirical constituent to be inferred, thus providing a means for evaluation of the capability of the proposed inference methodology.

3.1 Model Definition and Integral-effect Experiments

Consider a consumer model, $\eta(\boldsymbol{x_c}, \boldsymbol{\theta_f}) = \boldsymbol{\theta_f} \boldsymbol{x_c}^3$, that relates operational parameters to a system response that has been captured by integral-effect experiments conducted at a finite number of select operational parameter settings (shown in Figure 3). In this example, synthetic integral effect experiments are generated using the 'true' underlying functional dependence, $\boldsymbol{\theta_f} = 2\sqrt{x_f}$. The integral effect experiments demonstrate a dependence upon the control parameter, x_f, but this dependence is not represented in the available consumer constituent model. Predictions of this consumer constituent model where the functional relationship is currently unaccounted for and its prediction errors are shown in Figure 4.

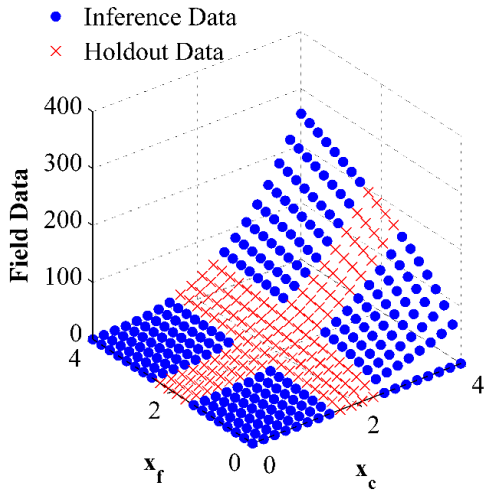


Figure 3. Integral-effect experimental data capturing coupled relationship.

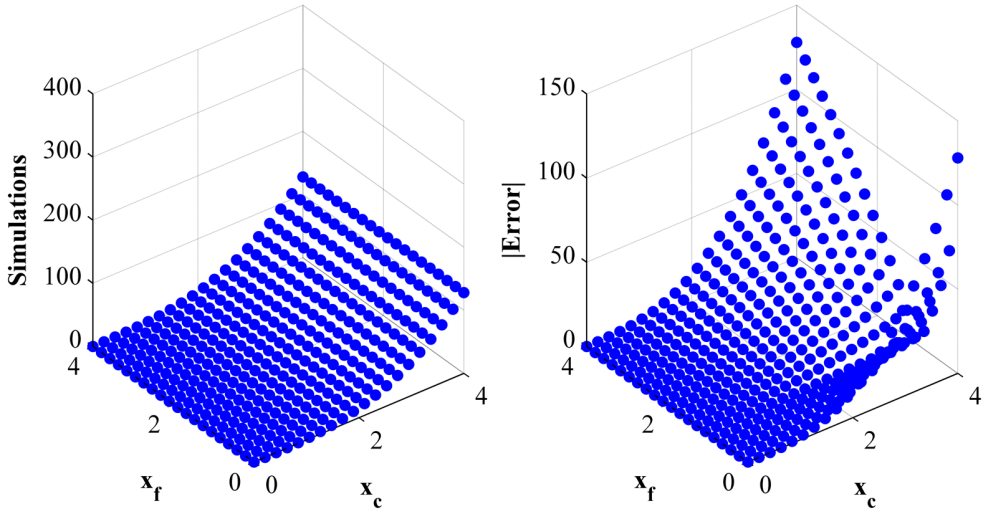


Figure 4. (Left) Physics-based model predictions where relationship to x_f is unmodeled and (right) error resulting in the physics-based predictions due to these modeling assumptions.

The inverse analysis method discussed in Section 2 is implemented to construct the relationship between θ_f and x_f from integral-effect experiments, shown in Figure 3, developing an empirical model to represent $\theta_f = f(x_f)$. Table 1 provides details of the parameter ranges for the inverse analysis, as well as the true functional form of a deterministic feeder model, which is currently missing and will be inferred empirically.

Table 1. Prior distributions and ranges for physics-based model parameters and GP hyperparameters.

Model Parameters				
		Minimum	**Maximum**	**True Value**
State Variables (x)	Independent Control (x_c)	0	4	-
	Dependent Control (x_f)	0	4	-
Uncertain Model Parameters (θ)	Functional Parameter (θ_f)	1	5	$2\sqrt{x_f}$
GP Hyperparameters				
Hyperparameter		**Prior Distribution**		
$\lambda_{\theta f}$		Ga(5, 5)		
$\rho_{\theta f}$		Beta(1, 0.2)		

3.2 Inverse Analysis: Results and Discussion

Twenty points are sampled along both x_c and x_f, with the middle 20% of the data in both operational parameter spaces held out as a validation set (Figure 3). As discussed in Section 2, a GP and its associated hyperparameters are inferred to define the empirical constituent $\theta_f(x_f)$ rather than imposing a user-defined functional form. Prior distributions on the GP hyperparameters are defined to yield an *a priori* mean of 1 and standard deviation of approximately 0.45 for λ_θ and to concentrate the density of ρ_{θ_f} close to 1. During the sampling algorithm, the functional parameter, θ_f is not confined by the prior bounds, as the sampling is allowed to explore beyond the initially defined minimum and maximum values. However, should the analyst prefer to place stricter boundaries on the functional parameter, such restrictions may be implemented (Brown and Atamturktur, 2018).

Burn-in runs are completed for 2000 iterations followed by the drawing of 2000 samples, each of which utilizes ten subiterations during the θ_f sampling. Three Markov chain Monte Carlo chains are carried out, each starting with different initial parameter values to ensure a converged solution and adequate mixing of the chains. Within each of these chains every other sample is retained, resulting in a collection of 3000 samples. The resulting approximate posterior distribution of the empirical constituent model form generated from these samples is shown in Figure 5.

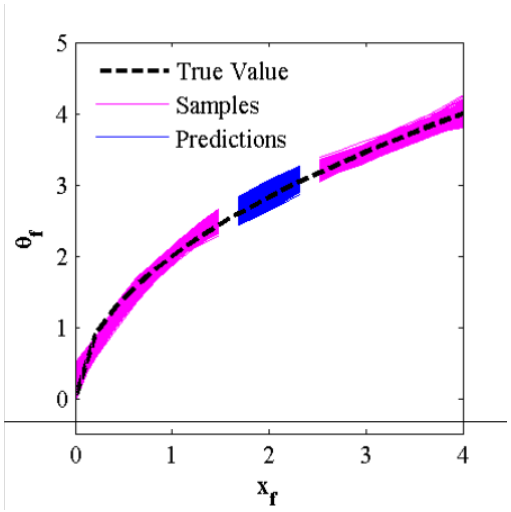


Figure 5. Posterior distribution of the empirical model $\theta_f(x_f)$ with predicted functional behavior at holdout settings.

This empirically inferred feeder model, $\theta_f(x_f)$ can now be coupled to the consumer model. Mean predictions made with this new *experimentally augmented partitioned model* are shown in Figure 6, as well as the error remaining in these predictions. Compared to the simulations with the stand-alone consumer model operating without the functional parameter representation (recall Figure 4), the predictive capability of the model has greatly improved, with the average percent error reducing from 33.42% in stand-alone to 18.05% with the coupled empirical constituent and physics-based constituent system model, a 46% overall reduction in the prediction error. Not only was the predictive capability improved, but knowledge of the underlying coupling physics was also gained through the functional form of the previously unknown dependence.

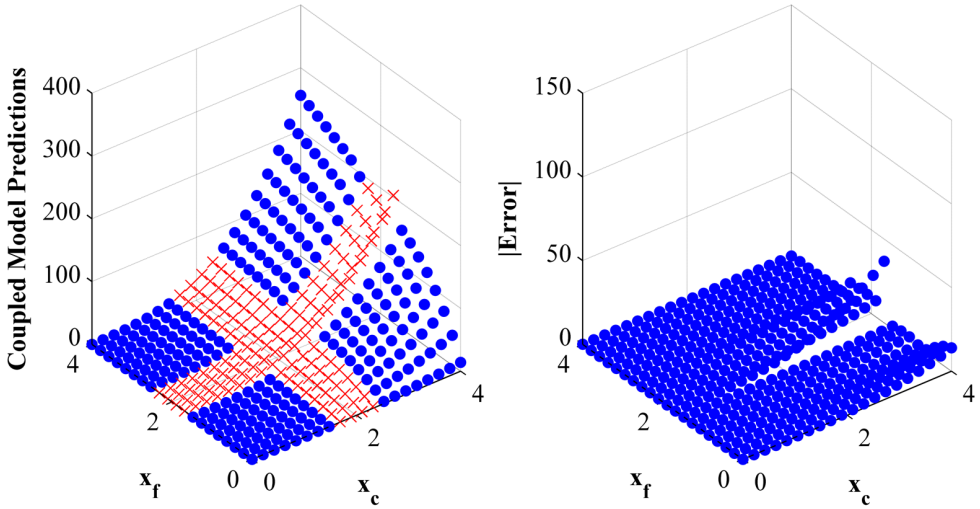


Figure 6. (Left) Predictions of the newly developed experimentally augmented partitioned model with a feeder empirical constituent coupled to the original consumer constituent and (Right) Error remaining in the coupled predictions compared to integral-effect experiments.

3.3 Effect of Subiterative Sampling of the Functional Calibration Parameter

Recall from Section 2.3 that the functional parameter is sampled repeatedly with subiterations in accordance with the recommendations of Neal (1998) to improve the mixing of the Markov chain. Neal (1998) originally proposed subiterations to reduce computational costs in problems where computing the covariance matrix is computationally demanding but computation of the likelihood is fast. In applying this method to engineering problems there are two scenarios that may be encountered; dense parameter sampling and computationally demanding model evaluations.

3.3.1 Large Covariance Matrix

As Neal (1998) presumed, the dense sampling of the dependent control parameter, or several dependent control parameters for a single functional parameter, will result in a large covariance matrix requiring inversions for every sample of the GP hyperparameters. In this case, dense sampling refers to the number of dependent control points, x_f, at which simulations are compared to experimental measurements. A fast-running model (less than 0.001 second to evaluate) is assumed and an analysis of the effect of subiterations in dense sampling is completed. The nominal model presented in Section 3.1 is used as a baseline for comparison. Size of the covariance matrix is increased by sampling the dependent control parameter, x_f, on a finer grid. In every case shown, 20% of the data remains as a holdout set and the number of total samples of the functional parameter each (meaning the number of subiterations multiplied by the number of Gibbs iterations) is kept constant at 1000 for three chains with different starting values, producing a full distribution of 3000 samples.

Figure 7 shows the degree to which the overall error of the empirical function is decreased with subiterations as well as the decrease in computational time required to complete the inference as the number of subiterations is increased, with the number of samples in the x_f dimension (and therefore the size of the covariance matrix) is denoted by N_x. Table 2 further emphasizes this point, showing a continued decrease in computational time with increasing number of subiterations. Also shown in Table 2 is that the error of the empirical model stabilizes at around four subiterations. In all of the different dimensionality cases tested, accuracy and computational efficiency are found to improve together as the number of subiterations increases. The first row of Table 2 shows the initial computational error and computational time resulting with one subiteration and 1000 Gibbs iterations. Figure 6 and Table 2 illustrate the accuracy of the functional parameter by root mean squared error:

$$\text{RMSE} = \sqrt{\frac{\sum_{i=1}^{N_x}\left(\theta_f^{true}(x_f{}^i) - \theta_f^{inferred}(x_f{}^i)\right)^2}{N_x}} \qquad \text{(Eq. 8)}$$

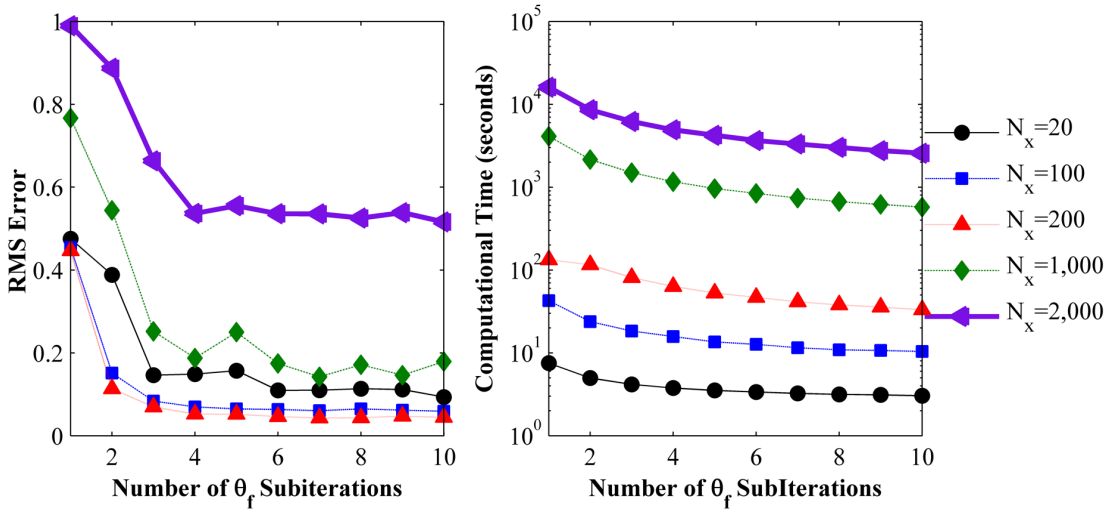


Figure 7. Convergence of parameter value error, standard deviation, and total computational time as number of subiterations of the θ_f parameter is increased.

Table 2. Accuracy and computational gains with increased parameter dimensionality.

Number of Subiterations	20 samples		200 samples		2000 samples	
	% Decrease in Error	*% Decrease in Comp. Time*	*% Decrease in Error*	*% Decrease in Comp. Time*	*% Decrease in Error*	*% Decrease in Comp. Time*
1	**0.475**	**7.5 s**	**0.455**	**42.6 s**	**0.99**	**16,079 s**
2	18.3%	33.8%	66.8%	44.0%	10.4%	46.5%
3	69.2%	44.5%	81.6%	56.9%	33.0%	61.4%
4	68.7%	49.9%	84.7%	63.0%	45.7%	69.4%
5	66.9%	53.1%	85.8%	68.1%	43.9%	73.7%
6	76.9%	55.1%	86.1%	70.2%	45.9%	77.3%
7	76.9%	56.8%	86.7%	72.9%	45.9%	79.4%
8	76.1%	57.9%	85.7%	74.4%	46.9%	81.2%
9	76.5%	58.6%	86.4%	74.9%	45.6%	82.9%
10	80.3%	59.6%	87.0%	75.6%	47.9%	84.0%

3.3.2 Computationally Demanding Model

The second case worth investigating is when the computational cost of the physics-based model is on the order of a few seconds (such as that of the VPSC model presented in Section 4), and the analyst chooses to evaluate the model itself rather than bypassing it with an emulator during computation of the likelihood. The desire to maintain the physics through model evaluations rather than low-fidelity surrogates is evident in the push for high-fidelity model implementation as well as growing research in multi-fidelity models when possible (Ng and Willcox, 2014). Such is the situation to be investigated here.

Suppose a set number of Gibbs samples for every parameter and hyperparameter has been selected and set to 1000. However, the chain mixing may be found insufficient as convergence is not achieved with this setting. In this case, though the problem may be low dimensional, the analyst may choose to begin conducting subiterations of the functional parameter to improve the convergence, requiring an increased number of evaluations of the physics-based model within each

Gibbs step. The results of such a decision are shown here for the nominal model presented in Section 3.1 with x_f sampled at 20 locations, 16 of which are used for the inference and 4 of which are kept as a holdout set, resulting in a low dimensional 16x16 covariance matrix.

Figure 8 illustrates the change in the resulting empirical constituent for a varying number of subiterations of the vector-valued parameter, θ_f. In this figure, every case utilizes three chains with different starting values and total 1000 samples of the functional parameter each (meaning the number of subiterations multiplied by the number of Gibbs iterations always equals 1000) with 1000 total runs being used in the burn-in step followed by 1000 total runs in the sampling step. Although each of the figures are generated with 3000 total samples once the three chains are combined, the GPs trained with up to two Gibbs subiterations of θ_f or less exhibit unsatisfactory convergence, as shown by the fact that all 1000 samples for each chain overlay each other making the appearance of only 3 samples drawn rather than 3000. These findings indicate that the proposal distributions being generated with 1 or 2 subsamples are consistently rejected, causing the adaptive step-size to reduce towards zero and the proposals values to become constant. This problem is relieved as the number of subiterations is increased.

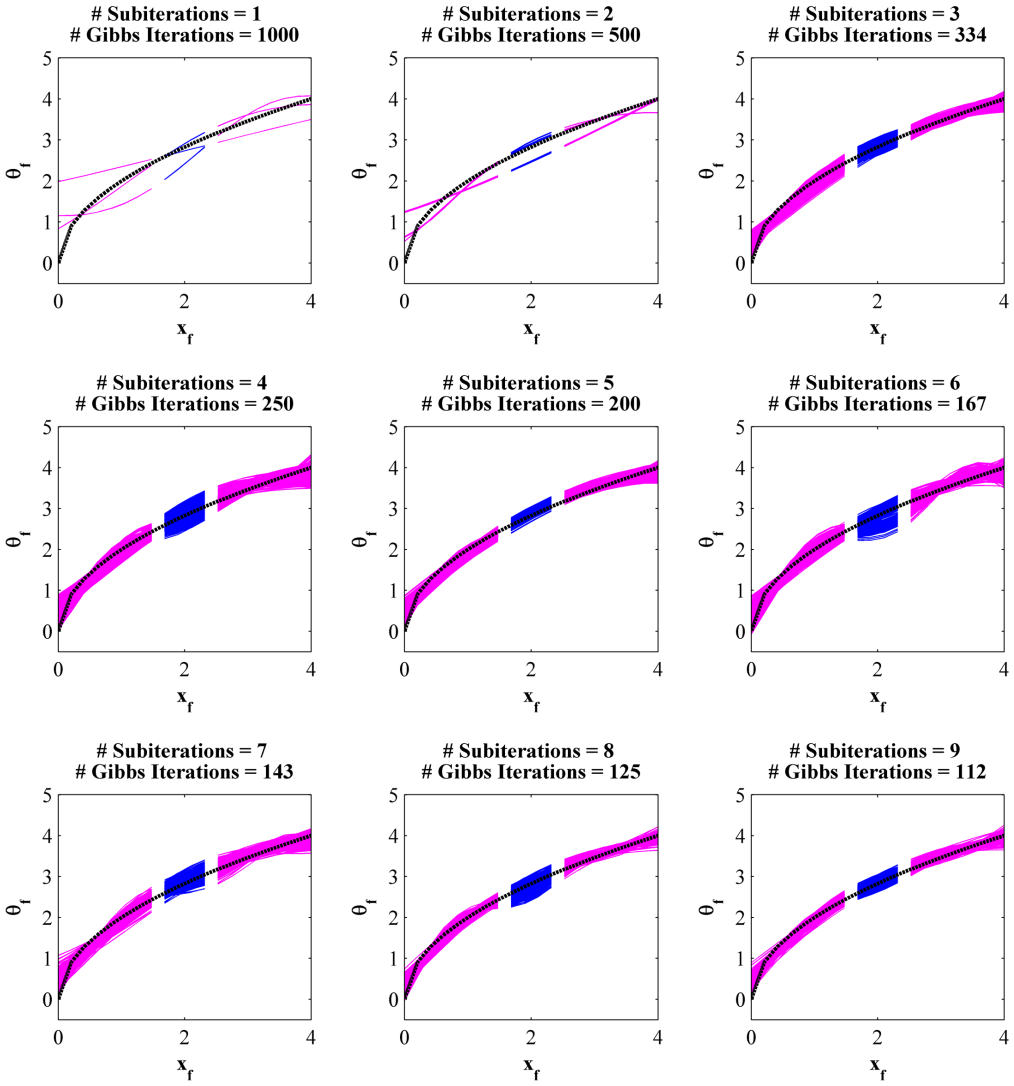


Figure 8. Improved convergence of posterior distributions with increased subiterations of the θ_f.

The small difference in samples from three subiterations to four, and likewise as subiterations continue to increase, is noticeable. Results demonstrate that once chains have sufficiently mixed and converged, little gains in prediction accuracy and precision may be gained from increased subsampling. Though the accuracy does not continue to improve with subiterations, the computational time does continue to decrease, demonstrating the capability of increased subiterations to support the use of more computationally demanding models. Figure 9 illustrates this trend, where the computational time continues to decrease without gaining or sacrificing significant neither accuracy nor precision, where computational time is the total time required to sample the three distinct chains run in serial.

Table 3 further emphasizes this point, where computational time is the total time required to sample the three distinct chains. In addition to the accuracy of the functional parameter estimation, uncertainty remaining in the estimation is represented in Figure 9 by standard deviation, which is averaged over all x_f points.

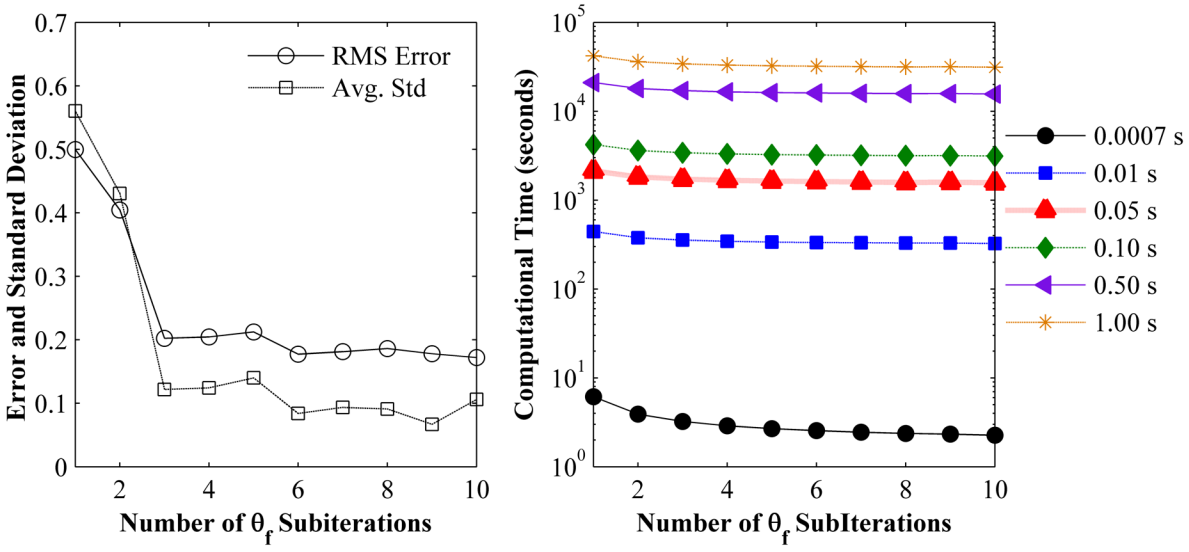


Figure 9. Convergence of parameter value error, standard deviation, and total computational time as number of subiterations of the θ_f parameter is increased.

Table 3. Change in inference results as number of iterations is increased.

Number of Subiterations	RMS error	Percent Decrease in Error from 1 sample	Computational Time (seconds)	Percent Decrease in Computation Time from 1 sample
1	0.322	--	14.9	--
2	0.200	**37.9%**	9.18	**38.4%**
3	0.136	**57.8%**	7.25	**51.3%**
4	0.140	**56.5%**	6.36	**57.3%**
5	0.102	**68.3%**	5.95	**60.1%**
6	0.114	**64.6%**	5.63	**62.2%**
7	0.100	**68.9%**	5.43	**63.6%**
8	0.107	**66.8%**	5.10	**65.8%**
9	0.118	**63.4%**	5.03	**66.2%**
10	0.101	**68.6%**	5.06	**66.0%**

Figure 8 and Table 3 are computed with the nominal computer model, which operates at approximately $7x10^{-4}$ seconds per model evaluation. This computational time is on the order of a surrogate model. However, the computation becomes more demanding as the physics-based model becomes increasingly demanding, such as the plasticity model to be discussed in Section 4. As such, the computational time of the physics-based model is increased and the analysis of the gains in accuracy and total computation time is assessed, the results of which are shown in Table 4, where the initial computational time with one subiteration and 1000 Gibbs iterations is shown in the row where number of samples is equal to one. This table serves to demonstrate the consistent trend in the percent decrease in computational time across the varied model run times, although this trend is difficult to see in Figure 9 due to the drastically different initial total computational times for each of the models. Another notable observation from Table 4 is that the rate at which computational demands decrease becomes less as the time required for a single model evaluation

increases, though the trend of continued gains in total computational time is still observed. These findings indicate that subiterations may alleviate the computational demands associated with the implementation of physics-based models and reduce the need to use surrogate models in the functional parameter sampling.

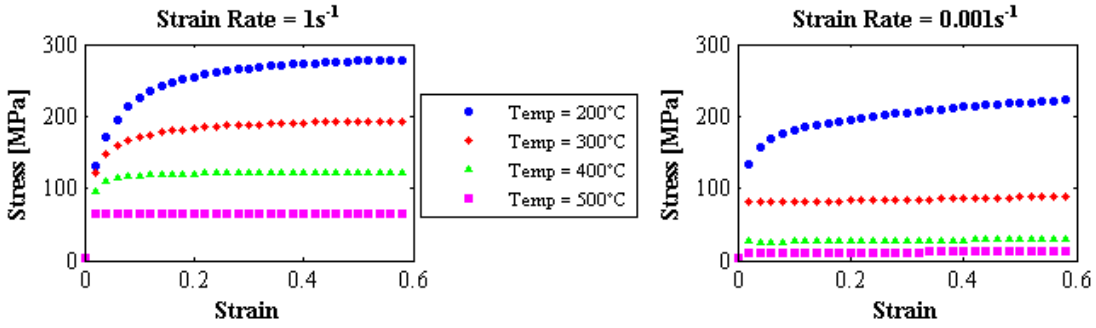
Table 4. Reduction in computational demands with the addition of subiterations.

Number of SubIterations	Percent Decrease in Comp. Time				
	0.01 s per evaluation	*0.05 s per evaluation*	*0.10 s per evaluation*	*0.50 s per evaluation*	*1.00 s per evaluation*
1	**443 s**	**2,132 s**	**4,217 s**	**21,040 s**	**42,050 s**
2	15.1%	14.4%	14.3%	14.3%	14.3%
3	19.9%	19.0%	18.9%	18.9%	18.9%
4	22.5%	21.6%	21.5%	21.5%	21.4%
5	24.0%	23.1%	22.9%	22.9%	22.9%
6	24.8%	24.1%	23.7%	23.7%	23.7%
7	25.2%	25.0%	24.3%	24.4%	24.4%
8	25.8%	25.8%	24.9%	25.0%	25.0%
9	25.8%	25.6%	24.7%	24.8%	24.8%
10	26.9%	26.5%	25.7%	25.7%	25.7%

4 5182 Aluminum with Temperature and Strain Rate Dependencies

4.1 Viscoplastic Self-Consistent Model and Experiments

The viscoplastic self-consistent (VPSC) model predicts texture evolution of highly anisotropic polycrystalline materials. One such material is the 5182 aluminum alloy, which in addition to exhibiting viscoplastic behavior also displays temperature and strain rate dependencies, captured by experiments conducted by Chen et al. (1998) (Figure 10). In an effort to identify the cause of the dependence, Stout et al. (1998) collected texture measurements in combination with the mechanical measurements. Texture measurements revealed that 5182 aluminum favors the classic (101) deformation texture at temperatures below 300°C but transitions to a combination of the classic (101) and static recrystallization (001) textures at temperatures above 400°C (Figure 10). Available experimental stress-strain curves reveal this temperature dependence at two separate strain rates (0.001 and 1 s^{-1}), as shown in Figure 10. Stress values are collected at 30 points along each of the four curves, with strains ranging from 0 to 0.58 and samples spaced 0.02 apart. Despite this known temperature dependence, there is currently no constituent model for the thermal analysis available to couple with VPSC and represent the effects of temperature on material behavior. The current model configuration is shown in Figure 11.



(a) Strain Rate = 1s^{-1} *(b) Strain Rate = 0.001s^{-1}*

Figure 10. Uniaxial compression experimental tests of 5182 Al and different strain rates and temperatures.

The VPSC model predicts the behavior of polycrystals with viscoplastic deformations using the governing equation shown in Eq. 9:

$$\dot{\varepsilon} = \dot{\gamma}_o \sum_{s=1}^{2} m^s \left(\frac{|m^s:\sigma|}{\tau_o^s}\right)^{n_g} \mathrm{sgn}(m^s:\sigma) \qquad \text{(Eq. 9)}$$

where σ is the stress applied to the crystal, $\dot{\varepsilon}$ is the strain rate, s is the number of slip and twinning systems active in the material (which is two in the case of the glide-only 5182 aluminum model), m^s is the Schmid tensor associated with glide, τ_o^s is the critical resolved shear stress, the exponent, n_g, represents the inverse of rate-sensitivity for glide activity, and $\dot{\gamma}_o$ denotes a normalization factor. The strain rate equation is summed over all active slip systems, N_s. Within this equation two parameters associated with the glide behavior n_g and resolved shear stress for the first deformation system, τ_o^1 are uncertain. While the glide parameter remains constant throughout the domain (Atamturktur et al., 2015), the critical resolved shear stress represents a hardening function and therefore is the parameter related to the changing texture evolution with respect to temperature and strain rate. This thermo-mechanical coupling between hardening parameters, temperature and strain rate on texture development can be accounted for through an empirical constituent model making τ_o^s a function of T, thus resulting an *experimentally augmented partitioned model*.

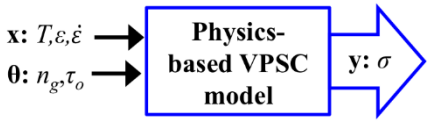


Figure 11. Current physics-based VPSC model without thermal constituent to represent thermo-mechanical coupling.

4.2 Inference of VPSC Weakly Coupled Physics: State-Aware Problem Formulation

Incorporating the empirical model with the physics-based plasticity model, the experimentally augmented coupled model may be written as:

$$\sigma = \eta\left(\varepsilon, \dot{\varepsilon}, n_g, \tau_0(T)\right) \qquad \text{(Eq. 10)}$$

where stress is the prediction to be compared with experiments (recall Figure 10). The roles of the input parameters in the inverse procedure and their respective ranges for the analysis are detailed in Table 5. Uniform prior distributions are assumed for both parameters with upper and lower bounds as indicated in Table 5. No constraints are placed on the form of the empirical feeder model inferred to represent the thermal constituent. These large ranges and minimal constraints leave much flexibility to the inverse analysis, making it possible to learn as much as possible from the experimental data.

Table 5.Parameter values of VPSC model and prior distribution of GP hyperparameters.

Model Parameter			**Minimum**	**Maximum**
State Variables (x)	Independent Control (x_c)	Strain (ε)	0	0.58
	Dependent Control (x_f)	Temperature (T) [C°]	180	570
Uncertain Model Parameters (θ)	Constant Parameter (θ_c)	n_g	1	5
	Functional Parameter (θ_f)	τ_0 [MPa]	2	1500
GP Hyperparameter			**Prior Distribution**	
$\lambda_{\theta f}$			Ga(5, 5)	
$\rho_{\theta f}$			Beta(1, 0.2)	

Prior distributions on the GP hyperparameters for the functional parameter, τ_0, are defined the same as in Section 3.2. Burn-in runs are completed for 800 iterations followed by the drawing of 1000 samples, each of which utilized ten subiterations of the θ_f sampling. Three MCMC chains are carried out, each starting with different initial values of all the parameter values to ensure that a converged solution is reached, ultimately leading to a collection of 3000 samples. To confirm the accuracy of the inverse procedure, 20% of the experimental data (ranging from a strain of 0.24 to 0.34) is held out during the inverse analysis for all temperature settings.

4.3 Results of Inverse Analysis

The posterior distribution of the τ_0 function is shown in Figure 12, where grey lines illustrate 3000 realizations and the solid black and dashed black lines denote the mean and one standard deviation of the posterior, respectively. Posterior distributions of the GP hyperparameters are shown in Figure 13. Statistics of the posterior for the empirical thermo-mechanical model at the available temperature settings are provided in Table 6.

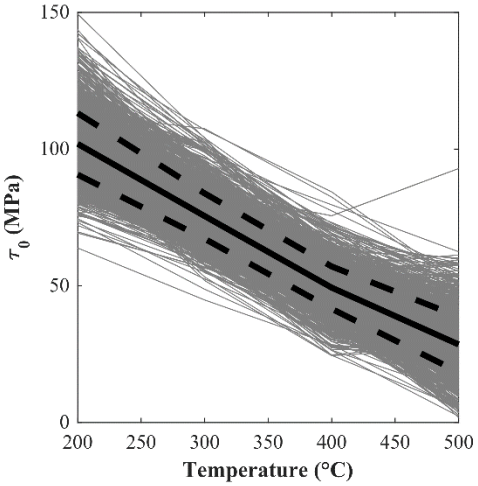


Figure 12. Posterior distribution of the functional parameter τ_0 at a strain rate of 1 s^{-1}.

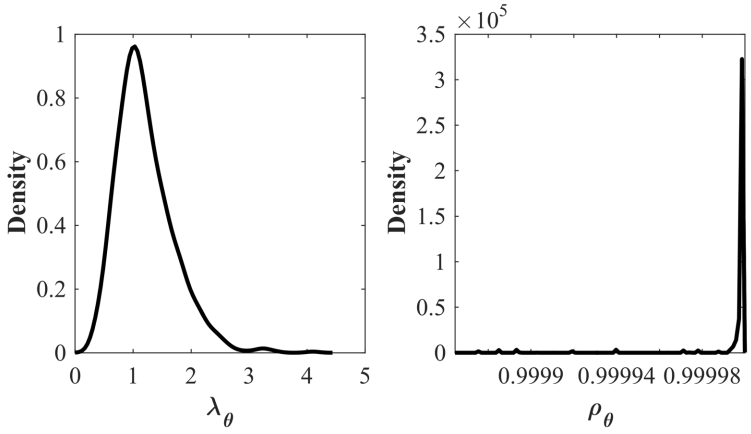


Figure 13. Posterior distributions of the empirical GP model hyperparameters.

Recall that the parameter n_g in the physics-based VPSC model is also uncertain though it has no dependence on the operational settings. The posterior distribution obtained for n_g, shown in Figure 14, is relatively little Bayesian learning when the inverse analysis is completed at a strain rate of 1 s^{-1}. The inability to calibrate n_g implies that the model's stress predictions are insensitive at this strain rate value, which is confirmed by an analysis of variance conducted at the both strain rate settings. The sensitivity of model output to n_g indeed varies greatly as a function of strain rate: τ_0 contributes to 100% of the variability in model predictions at a strain rate of 1 s^{-1} while the R^2 values in the main effect screening at a strain rate of 0.001 s^{-1} are 42% and 58% for τ_0 and n_g, respectively. Posterior distributions of the constant and functional parameters with respect to experiments at a strain rate of 0.001 s^{-1} are shown in Figure 15. Comparison of the calibrated parameter values and remaining uncertainty at both strain rate settings is shown in Table 6, where the mean and standard deviation of posteriors for the functional parameter are shown at available temperature settings, as the posterior varies throughout the operational domain. The difference in the τ_0 functions for the two strain rates is to be expected given the known dependence on strain rate (Atamturktur et al., 2015; Chen et al., 1998; Lebensohn et al., 2010; Stout et al., 1998).

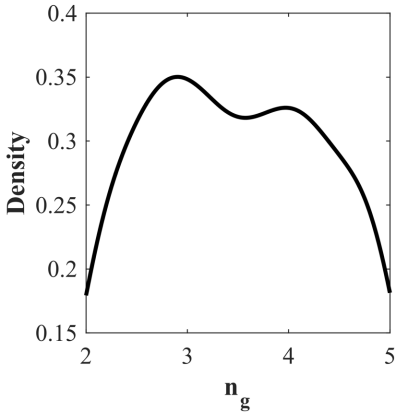


Figure 14. Posterior distribution of constant VPSC physics-based model parameter when operating at a strain rate of 1s^{-1}.

In Table 6, the difference in standard deviations of inferred parameter values, indicating uncertainty remaining in the predictions, are worth noting for the two different strain rate settings. When the inverse analysis is completed with respect to experiments conducted at a strain rate of 1 s^{-1} (where the model is highly sensitive to τ_0 and not at all sensitive to n_g) the standard deviation of n_g is 24.9% of the mean value. When the model is calibrated with respect to experiments conducted at a strain rate of 0.001 s^{-1} (where the model also becomes sensitive to n_g), the standard deviation of n_g is reduced to 10.7% of the mean value. Results of this analysis suggest that the model calibration would improve if both strain rate settings could be considered simultaneously. Provided the opportunity for more data at other strain rate settings, training a function for τ_0 dependent upon both temperature and strain rate would be the best path forward.

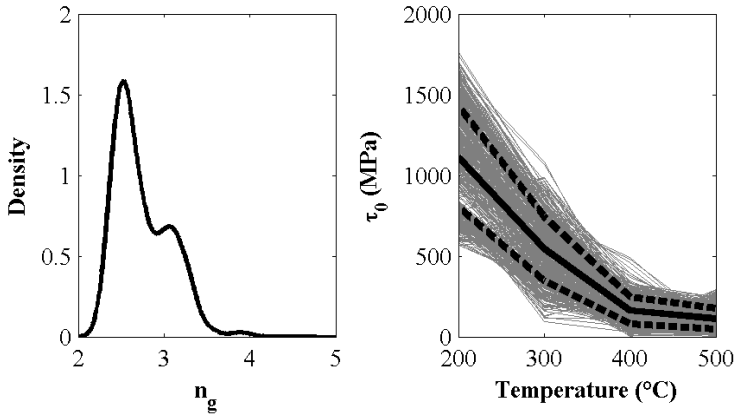


Figure 15. Posterior distributions of the (left) constant VPSC model parameter, n_g and (right) functional parameter τ_0 inferred for operations at a strain rate of 0.001 s^{-1}.

Table 6. Comparison of the empirical thermo-mechanical constituent model and uncertain VPSC model parameter for inverse analysis with different strain rates.

Parameter	Temperature Setting (C°)	*Strain Rate = 1 s^{-1}*		*Strain Rate = 0.001 s^{-1}*	
		Mean	**Standard Deviation**	**Mean**	**Standard Deviation**
Constant Parameter (θ_c) n_g	--	3.57	0.89	2.80	0.30
Functional Parameter (θ_f) τ_0 [MPa]	200	102.2	11.23	1115.3	310.6
	300	75.70	8.29	545.5	198.7
	400	49.44	7.68	162.4	83.41
	500	28.28	10.05	112.0	64.95

With the model parameters and corresponding GP hyperparameters determined through the inverse analysis, the resulting GP model for $\tau_0(T)$ is used to determine a trained polynomial function that is then implemented as an empirical thermal constituent model and coupled with the existing physics-based VPSC plasticity model. The newly available experimentally augmented partitioned model simulating the thermo-mechanical coupling is shown in Figure 16. Figure 17 illustrates ultimate stress predictions of the coupled model compared with the available experimental data. Ultimate stress is the stress value obtained at the last strain value, which is typically the prediction of greatest interest to decision makers as it relates to the point at which failure is most likely to occur. Table 7 provides statistics of the predictions of the newly available experimentally augmented model with the GP thermal constituent coupled to the VPSC mechanical constituent. This table captures an averaged picture of results of the empirically augmented model across all strain settings as opposed to only the final strain shown in Figure 17.

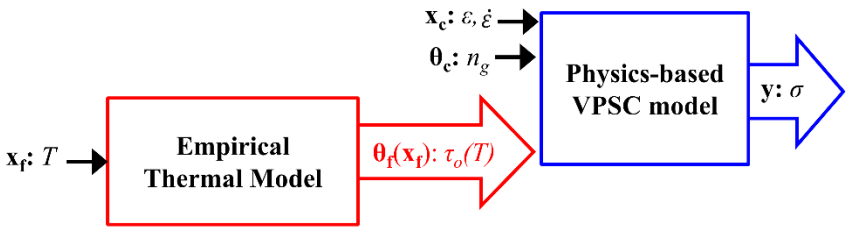


Figure 16. Coupled thermo-mechanical model, composed of an empirical thermal constituent and physics-based plasticity constituent, for predicting mechanical behavior of 5182 aluminum alloy at varying temperature settings.

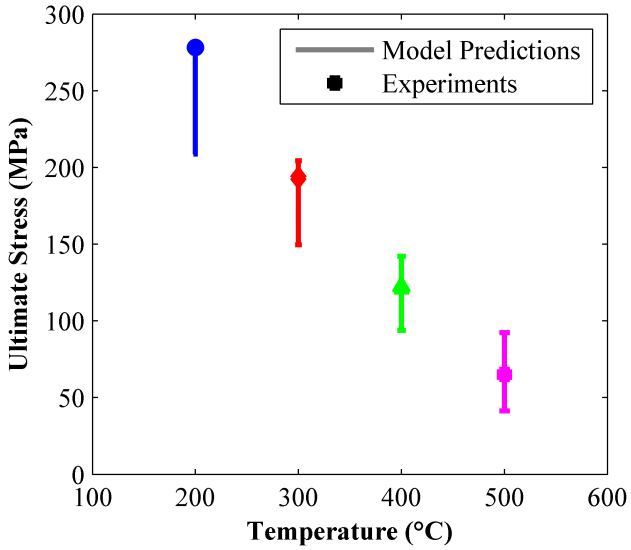


Figure 17. Ultimate stress predictions taken at the maximum strain value at a strain rate of $1s^{-1}$ with remaining uncertainty.

Table 7. Statistics of the experimentally augmented GP-VPSC thermo-mechanical model value at a strain rate of $1s^{-1}$.

	Predictions of Coupled Thermo-mechanical Model $[\sigma(T,\varepsilon)]$	
Temperature Setting (C°)	RMS Error (MPa)	Average Standard Deviation (MPa)
200	34.9	36.5
300	14.9	27.7
400	3.50	24.0
500	2.29	25.6

5 Conclusions

Model developers are recognizing more than ever the need to account for interrelated physics and scales through coupled modeling. Partitioned analysis presents an attractive solution for these complex problems. However, the physics of one or more constituents in these relationships is sometimes unknown and therefore cannot be modeled directly. The omission of a constituent model degrades the predictive capability of the full system, causing model bias. Traditional model calibration in the Bayesian context requires this model bias to be represented by a discrepancy term. However, determining missing relationships during the parameter calibration can potentially mitigate this model bias without a discrepancy model (Atamturktur et al. 2016).

Deciphering model bias is problem that becomes more pronounced as the complexity of the system increases and knowledge regarding the underlying physics becomes increasingly incomplete. For instance, the physics-based model may be a legacy code that cannot be easily changed. Missing coupling relationships can occur in legacy codes when, for example, new materials or systems are being implemented in the code where an operational state, which was not influential previously, becomes influential, or when new experiments become available showing dependence that was previously unknown. Furthermore, the empirical model will be intrinsically application-dependent based on the integral-effect experiments used in the inverse analysis. Therefore, different empirical models may be inferred for a variety of applications and partitioned analysis makes swapping empirical models for different applications a simple process.

The notion of treating the empirically derived function as an independent constituent model is attractive as it isolates the empirical components (conditioned upon experiments) from the physics-based components (based on fist principle understanding). This transparency, which would be lost if the empirical relationship was directly integrated into the physics-based model with a monolithic approach, allows for easy updating of the empirical model as new data becomes available. Maintaining separation also provides clear indication of experimentally-conditioned model components to future code developers and designers.

The inverse analysis proposed herein infers, from readily available integral-effect experimental measurements, the functional relationships neglected in physics-based models, providing the model developer with an empirical constituent model that can be coupled to the existing physics-based model to account for important dependencies. In addition to yielding a more accurate computer model, inferring this unknown relationship from observations provides insight into a more complete mathematical representation of the nature of missing physical relationships, furthering the scientific understanding of these processes. In the case that all missing physics in the model is explained by the parameter relationships inferred by dependent calibration, the model bias of traditional calibration could be eliminated. However, if other relevant physics or parameters remain absent from the model, then model bias will remain and a discrepancy model will be necessary in the calibration, thus modifying Eq. 2 to include a discrepancy model, δ, as shown in Eq. 11. Defining this discrepancy model across several dependent and independent control domains is an area of interest for future work.

$$y(\mathbf{x}_f^i, \mathbf{x}_c^j) = \eta(\mathbf{x}_f^i, \mathbf{x}_c^j, \boldsymbol{\theta}_f(\mathbf{x}_f^i), \boldsymbol{\theta}_c) + \varepsilon(\mathbf{x}_f^i, \mathbf{x}_c^j) + \delta(\mathbf{x}_f^i, \mathbf{x}_c^j) \quad \text{(Eq. 11)}$$

where i=1, 2, …, n and j=1, 2, …, m

Finally, the GP approach developed herein is capable of providing not only GP hyperparameter values, but also recommendations of parametric model forms for future use as the empirical constituent is coupled to the existing physics-based model. The advantage of beginning the functional exploration with a GP is the few number of hyperparameters required, allowing the MCMC to deal with smaller dimensional parameter spaces, thus maintaining feasible computational times. The GP inferred by few hyperparameters would suggest a proper parametric

model form for further inference. The parametric model may then be applied for sampling highly discretized control domains without the computational constraint of a large covariance matrix.

The proposed methodology has demonstrated the capability to infer a physical dependence of 5182 aluminum material properties on the temperature at which the material is being loaded, producing an empirical model to be coupled with the existing VPSC physics-based model. These results show the promise of Gaussian process inverse analysis to demystify influential thermo-mechanical relationships underlying complex materials implemented in engineering and science today. The case study presented with 5182 aluminum also illustrates a path forward for continued development of the method to provide empirical constituent models dependent upon more than one operational state. Furthermore, the possibility of missing physics of the computer model as a consequence of missing model parameters resulting in a level of discrepancy bias remaining is recognized. Hence, discrepancy should also be included in this calibration framework.

References

Atamturktur, S., Hegenderfer, J., Williams, B., Egeberg, M., Lebensohn, R.A., and Unal, C. (2015). A Resource Allocation Framework for Experiment-Based Validation of Numerical Models. Mech. Adv. Mater. Struct. *22*, 641–654.

Atamturktur, S., Stevens, G.N. and Brown, D.A. (2016). Empirically Improving Model Adequacy in Scientific Computing. International Modal Analysis Conference, Anaheim, CA.

Auricchio, F., Fugazza, D., and Desroches, R. (2007). Rate-dependent Thermo-mechanical Modelling of Superelastic Shape-memory Alloys for Seismic Applications. J. Intell. Mater. Syst. Struct. *19*, 47–61.

Bastos, L.S., and O'Hagan, A. (2009). Diagnostics for Gaussian Process Emulators. Technometrics *51*, 425–438.

Bauer, T.H., and Holland, J.W. (1995). In-Pile Measurement Of The Thermal Conductivity Of Irradiated Metallic Fuel. Nucl. Technol. *110*, 407–421.

Brown, D.A. and Atamturktur, S. (2018), "Nonparametric functional calibration of computer models," *Statistica Sinica*, 28, to appear.

Chen, S.R., Stout, M.G., Kocks, U.F., MacEwen, S.R., and Beaudoin, A.J. (1998). Constitutive Modeling of a 5182 Aluminum as a Function of Strain Rate and Temperature. In Hot Deformation of Alluminum Alloys II Symposium, (Rosemont, Illinois),.

Dunić, V., Busarac, N., Rakić, D., Slavković, V., Slavković, R., and Živković, M. (2012). Thermo-Mechanical Coupling Procedure Using Partitioned Approach-Application To Arc Welding Simulation. J. Serbian Soc. Comput. Mech. *6*, 29–40.

Galloway, J., Unal, C., Carlson, N., Porter, D., and Hayes, S. (2015). Modeling Constituent Redistribution in U-Pu-Zr Metallic Fuel Using the Advanced Fuel Performance Code BISON. Nucl. Eng. Des. *286*, 1–17.

Gelfand, A.E., and Smith, A.E. (1990). Sampling-based Approaches to Calculating Marginal Densities. J. Am. Stat. Assoc. *85*, 398–409.

Geman, S., and Geman, D. (1984). Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images. IEEE Trans. Pattern Anal. Mach. Intell. 721–741.

Gilks, W.R., Best, N.G., and Tan, K.K.C. (1995). Adaptive Rejection Metropolis Sampling within Gibbs Sampling. Appl. Stat. *44*, 455.

Hastings, W.K. (1970). Monte Carlo Sampling Methods Using Markov Chains and Their Applications. Biometrika *57*, 97.

Higdon, D., Kennedy, M., Cavendish, J.C., Cafeo, J.A., and Ryne, R.D. (2004). Combining Field Data and Computer Simulations for Calibration and Prediction. SIAM J. Sci. Comput. *26*, 448–466.

Jackson, K.E., Littell, J.D., Horta, L.G., Annett, M.S., Fasanella, E.L., and Seal, M.D. (2014). Impact testing and simulation of composite airframe structures.

Kennedy, M.C., and O'Hagan, A. (2001). Bayesian Calibration of Computer Models. J. R. Stat. Soc. Ser. B Stat. Methodol. *63*, 425–464.

Lebensohn, R.A., Hartley, C.S., Tomé, C.N., and Castelnau, O. (2010). Modeling The Mechanical Response Of Polycrystals Deforming By Climb And Glide. Philos. Mag. *90*, 567–583.

MacKay, D.J. (1998). Introduction to Gaussian Processes. NATO ASI Ser. F Comput. Syst. Sci. *168*, 133–166.

Metropolis, N., Rosenbluth, A.W., Rosenbluth, M.N., Teller, A.H., and Teller, E. (1953). Equation of State Calculations by Fast Computing Machines. J. Chem. Phys. *21*, 1087.

Neal, R.M. (1998). Regression and Classification Using Gaussian Process Priors. Bayesian Stat. *6*, 1–16.

Ng, L.W.T., and Willcox, K.E. (2014). Multifidelity approaches for optimization under uncertainty: Multifidelity Approaches For Optimization Under Uncertainty. Int. J. Numer. Methods Eng. *100*, 746–772.

Ranjan, P., Haynes, R., and Karsten, R. (2011). A Computationally Stable Approach to Gaussian Process Interpolation of Deterministic Computer Simulation Data. Technometrics *53*, 366–378.

Rasmussen, C.E., and Williams, C.K.I. (2006). Gaussian processes for machine learning (Cambridge, Mass: MIT Press).

Sacks, J., Welch, W.J., Mitchell, T.J., and Wynn, H.P. (1989). Design and Analysis of Computer Experiments. Stat. Sci. 409–423.

Santner, T.J., Williams, B.J., and Notz, W.I. (2003). The Design And Analysis Of Computer Experiments (Springer Science & Business Media).

Shen, B., Stephansson, O., and Rinne, M. (2014). Modelling Rock Fracturing Processes (Dordrecht: Springer Netherlands).

Stout, M.G., Chen, S.R., Kocks, U.F., Schwartz, A.J., Macewen, S.R., and Beaudoin, A.J. (1998). Mechanisms Responsible for Texture Development in a 5182 Aluminum Alloy Deformed at Elevated Temperature. (Rosemont, Illinois),.

Swiler, L.P. (2006). Bayesian Methods in Engineering Design Problems (Sandia National Laboratories).

Unal, C., Williams, B.J., Yacout, A., and Higdon, D.M. (2013). Application of advanced validation concepts to oxide fuel performance codes: LIFE-4 fast-reactor and FRAPCON thermal-reactor fuel performance codes. Nucl. Eng. Des. *263*, 102–128.

Williams, C.K.I., and Rasmussen, C.E. (1996). Gaussian Processes for Regression. Adv. Neural Inf. Process. Syst.

Williamson, R.L., Hales, J.D., Novascone, S.R., Tonks, M.R., Gaston, D.R., Permann, C.J., Andrs, D., and Martineau, R.C. (2012). Multidimensional multiphysics simulation of nuclear fuel behavior. J. Nucl. Mater. *423*, 149–163.