

Nonhierarchical Multi-model Fusion Using Spatial Random Processes

Shishi Chen^{a,b,c,*}, Zhen Jiang^{c,*}, Shuxing Yang^{a,b}, Daniel W. Apley^d, Wei Chen^{c,†}

^a *School of Aerospace Engineering, Beijing Institute of Technology, Beijing 100081, PR China*

^b *Key Laboratory of Dynamics and Control of Flight Vehicle, Ministry of Education, Beijing 100081, PR China*

^c *Department of Mechanical Engineering, Northwestern University, Evanston, Illinois 60208, USA*

^d *Department of Industrial Engineering & Management Science, Northwestern University, Evanston, Illinois 60208, USA*

ABSTRACT

New model fusion techniques based on spatial-random-process modeling are developed in this work for combining variable-fidelity data from simulations and experiments. Existing works in variable-fidelity modeling generally assume a hierarchical structure in which the levels of fidelity of the simulation models can be clearly ranked. In contrast, we consider the nonhierarchical situation in which one wishes to incorporate multiple models whose levels of fidelity are unknown or cannot be differentiated (e.g., if the fidelity of the models changes over the input domain). We propose three new nonhierarchical multi-model fusion approaches with different assumptions or structures regarding the relationships between the simulation models and physical observations. One approach models the true response as a weighted sum of the multiple simulation models and a single discrepancy function. The other two approaches model the true response as the sum of one simulation model and a corresponding discrepancy function, and differ in their assumptions regarding the statistical behavior of the discrepancy functions, such as independence with the true response or a common spatial correlation function. The proposed approaches are compared via numerical examples and a real

* S. Chen and Z. Jiang contributed equally to this work.

† Corresponding author. Tel.: 847-491-7019; Fax: 847-491-3915; E-mail: weichen@northwestern.edu.

engineering application. Furthermore, the effectiveness and relative merits of the different approaches are discussed.

KEY WORDS: multi-model fusion; multi-fidelity modeling; spatial random process; nonhierarchical model fidelity; computer experiments

1. INTRODUCTION

Complex computer simulation models are commonly built to explore the underlying mechanisms of real physical processes, but conducting a large number of simulation runs at many input combinations to obtain an extensive coverage of the input space is usually unaffordable due to high computational cost. A more feasible alternative to simulating a single high-fidelity model is building several models at different lower levels of fidelity, such as intermediate-fidelity physics-based computer-aid engineering (CAE) models, intermediate-fidelity non-physics-based surrogate models, and low-fidelity simplified handbook equations. In addition to lower computational costs, such lower-fidelity models usually have merits in capturing some fundamental features of the true physics and providing useful information for predicting the responses at the input sites where no high-fidelity simulations or physical experiments are conducted due to limitations of time or resources. It is generally beneficial to incorporate the lower-fidelity data together with the high-fidelity data to create an accurate and yet efficient predictive model, a process that we refer to as *model fusion*. Existing model fusion methods generally assume the fidelity levels of the simulation models can be clearly ranked, and they apply a hierarchical treatment. However, information about the ranking of the model fidelities is not always available, due to either a lack of prior knowledge, or a fidelity level that changes over the input space (which we refer to as *range-dependent model fidelity*). Hence, it is of significant interest to develop a more generic model fusion technique for combining information from multiple models with no clear ranking of model fidelities.

Both differentiable and non-differentiable model fidelities widely exist in

simulation-based engineering design. An example of models with clear levels of fidelity is a finite element simulation for the vehicle crashworthiness design which can be run using different resolutions of the grid. Furthermore, a sophisticated physical process often can be modeled using different physical laws and solved using different numerical techniques, such as the aircraft design of aerodynamic components for which the simulations may differ in terms of the reduced physical order (e.g., Euler model vs. potential flow model) or numerical solver (e.g., finite difference method vs. finite element analysis). On the other hand, models with unknown/non-differentiable fidelities could be for a climate system, where the models are developed from different research groups to understand and predict its behavior, but they are based on disparate theories or mechanisms to incorporate the physics and chemistry of the atmosphere, ocean and land surface. In such a situation, the levels of fidelity of different computer models may appear to be similar (unless a large amount of experimental/simulation data have been collected); or the fidelities of simulation models may change over the input space, as the modeler's knowledge on some input regions may be scanty.

Model fusion has received an increasing amount of attention. Early work [1-4] has been focused on *bi-model fusion* where a single low-fidelity simulation model is integrated with the *high-fidelity data* for response predictions. A set of high-fidelity data often include responses from high-fidelity industry-standard CAE simulations and/or physical experiments. The most popular framework for bi-model fusion is proposed by Kennedy and O'Hagan [5] and has been widely adopted in a great deal of previous works in the literature [6-12]. Till now extensive research efforts have been devoted to developing *multi-model fusion* techniques as well to combining high-fidelity data with multiple sources of low-fidelity data for building an accurate and yet efficient predictive model. Some works have extended the preceding bi-model fusion approach with a hierarchical updating framework [13-20]. The higher-fidelity model is always approximated by adding its next lower-fidelity model with a discrepancy function.

Sequentially, the highest-fidelity surrogate model for prediction can be built as the sum of the lowest-fidelity model and all the discrepancy functions. Another category of approaches represents the models and discrepancies using stochastic expansion methods [21, 22] from a numerical analysis point of view. Eldred et al. [23, 24] proposed to use non-intrusive polynomial chaos and stochastic collocation methods to build a surrogate model for the difference between the low-fidelity and high-fidelity models. Xiu et al. [25, 26] employed a greedy procedure based on the information from the lower-fidelity model to collect “important” samples for the higher-fidelity simulations, and then applied the stochastic collocation method to construct an approximate mapping rule between the lower-fidelity and higher-fidelity spaces.

However, all these aforementioned multi-model fusion approaches commonly assume that the fidelity levels of the simulation models can be clearly identified and preliminarily ranked. In the presence of the non-differentiable model fidelity situations, we develop three new nonhierarchical multi-model fusion approaches, which are more flexible to handle various complex model fidelity scenarios. The proposed approaches are constructed based on the spatial-random-process (SRP) modeling which flexibly captures the nonlinearity of the models and inherently quantifies the interpolation uncertainty due to the lack of data. Different assumptions and mechanisms are assigned to the distinct approaches for acquiring the relationships between simulation models and physical observations. The first approach employs a weighted-sum structure to combine the simulation models and adds a single discrepancy function for final prediction. The second approach assigns an individual discrepancy function, independent of the true physical process, for each simulation model. In the third approach, a fully correlated multi-response SRP structure with a common spatial correlation function is adopted for the simulation models and their discrepancy functions altogether.

The remainder of the paper is organized as follows: In Section 2, we briefly review basic concept in SRP modeling and some fundamental ideas of the existing hierarchical

multi-model fusion techniques. Section 3 provides a detailed description of the three proposed nonhierarchical multi-model fusion approaches. In Section 4, the proposed approaches are applied to several numerical examples and a real engineering application, with two validation metrics computed to compare their effectiveness and relative merits. Conclusions and possible extensions of our approaches are presented in Section 5.

2. BACKGROUND FOR MODEL FUSION

The goal of this section is to briefly introduce the basic concept of SRPs and review the existing hierarchical frameworks for multi-model fusion using SRPs.

2.1 Basic Concept of Spatial Random Processes (SRPs)

An SRP is a collection of random variables distributed over some spatial or temporal domain [27, 28], and it has been widely used to emulate the various forms of functions, especially sophisticated computer models with expensive computation. A *Gaussian process* (GP) is one of the most popular SRPs, due to its many desirable properties. In a GP, any finite collection of the random variables follows a multivariate normal distribution. In our work, we apply it to build surrogate models for the functional responses of interest. Let $\mathbf{x} = [x_1, \dots, x_p] \in \mathbb{R}^p$ be a p -dimensional set of input variables, and $y : \mathbb{R}^p \rightarrow \mathbb{R}$ a functional response. A GP model can be built to represent $y(\mathbf{x})$,

$$y(\mathbf{x}) \sim \mathcal{GP}(m(\mathbf{x}), V(\mathbf{x}, \mathbf{x}')), \quad (1)$$

where $m(\mathbf{x})$ is the *mean function* usually expressed as $\mathbf{h}(\mathbf{x})^T \boldsymbol{\beta}$, with $\mathbf{h}(\mathbf{x})$ denoting a column vector of pre-specified polynomial functions (i.e., constant, linear, quadratic, etc.), and $\boldsymbol{\beta}$ a column vector of to-be-determined regression coefficients; $V(\mathbf{x}, \mathbf{x}') = \sigma^2 R(\mathbf{x}, \mathbf{x}')$ is the *covariance function*, representing the spatial covariance between any two inputs \mathbf{x} and \mathbf{x}' of the process, with σ denoting the prior standard deviation and $R(\mathbf{x}, \mathbf{x}')$ the *spatial correlation function*. One popular choice for $R(\mathbf{x}, \mathbf{x}')$ in the computer experiment literature

[29-33] is the Gaussian correlation function

$$R(\mathbf{x}, \mathbf{x}') = \exp\left\{-\sum_{k=1}^p \omega_k (x_k - x'_k)^2\right\}, \quad (2)$$

where $\boldsymbol{\omega} = [\omega_1, \omega_2, \dots, \omega_p]^T$ is a vector of spatial correlation parameters (aka scale or roughness parameters) used to represent the rate at which the correlation between $y(\mathbf{x})$ and $y(\mathbf{x}')$ decays to zero as \mathbf{x} and \mathbf{x}' diverge. Note that Eq. (2) implies a covariance-stationary GP model, which we assume in all examples in this paper, although the mean can vary spatially. However, all of our approaches can be applied directly to nonstationary covariance models, as well.

The collection of unknown parameters $\boldsymbol{\phi} = \{\boldsymbol{\beta}, \sigma, \boldsymbol{\omega}\}$, referred to as *hyperparameters*, characterizes the GP model. After observing a set of data $\mathbf{d} = [y(\mathbf{x}_1), y(\mathbf{x}_2), \dots, y(\mathbf{x}_n)]^T$ at input settings \mathbf{x}_i ($i = 1, \dots, n$), the maximum likelihood estimation (MLE) method is typically employed to estimate the hyperparameters $\boldsymbol{\phi}$ by maximizing the likelihood (or the log-likelihood) function using a numerical optimization algorithm. Subsequent to estimating $\boldsymbol{\phi}$, a prediction of $y(\mathbf{x}^p)$ at any new input setting \mathbf{x}^p that is not yet tested can be obtained. The detailed equations for calculating the posterior mean prediction $\hat{y}(\mathbf{x}^p)$ and its corresponding mean square error $\text{MSE}[\hat{y}(\mathbf{x}^p)]$ are summarized in [28, 30, 34].

It should be noted that the sense in which the response variable must be Gaussian in order to use GP modeling is quite different than the sense in which a random sample of observations must be Gaussian in order to assume normality in classical statistics. In GP-based response surface metamodeling, we only observe a single response variable y as a function of the input variables \mathbf{x} . The primary assumption behind GP-based modeling is that this response surface function can be viewed as a *single* realization of a Gaussian process. Fig. 1 below shows three such realizations of a GP for the case of a one-dimensional input domain. Given a single observed response surface for any particular application, it is virtually impossible to check whether the Gaussian assumption of the GP model is satisfied; but for similar reasons, neither does it matter. Single

realizations of a GP can have widely different characteristics and appearances (the quite different realizations in Fig. 1 were all generated using the same covariance function, and would have differed more dramatically in appearance if different covariance functions were used). Consequently, a single GP realization can closely resemble practically any actual engineering response surface, if the appropriate covariance function is chosen or fitted, which partly accounts for the widespread usage of GP models for engineering response surface metamodeling.

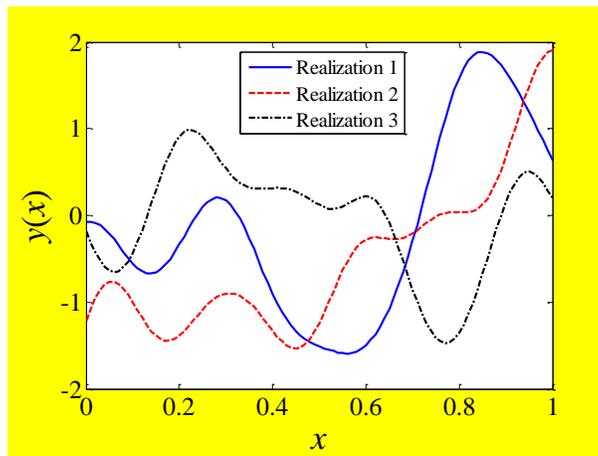


Fig. 1 Three different realizations of a GP

2.2 Existing Hierarchical Multi-model Fusion Approaches

To represent the relationships between simulation models with different levels of fidelity, much prior works (e.g., [13-16, 35]) on multi-model fusion employ the hierarchical model

$$y^{m\{i\}}(\mathbf{x}) = \rho_{i-1} y^{m\{i-1\}}(\mathbf{x}) + \delta_{i-1}(\mathbf{x}) \quad (i = 2, \dots, Q), \quad (3)$$

where $y^{m\{i\}}(\mathbf{x})$ denotes the simulation model response at the i th fidelity level. Higher i corresponds to higher fidelity level, thus $y^{m\{1\}}(\mathbf{x})$ is the lowest-fidelity model and $y^{m\{Q\}}(\mathbf{x})$ the highest. The parameter ρ_{i-1} , referred to as the *scaling factor*, captures the strength of the relationship between $y^{m\{i\}}(\mathbf{x})$ and $y^{m\{i-1\}}(\mathbf{x})$, and $\delta_{i-1}(\mathbf{x})$ represents the discrepancy between $y^{m\{i\}}(\mathbf{x})$ and $\rho_{i-1} y^{m\{i-1\}}(\mathbf{x})$.

In order to handle more complex discrepancies between higher- and lower-fidelity

models (e.g., if the model discrepancy is highly nonlinear over the entire input space), Qian et al. [19, 20] introduced a non-constant scaling factor $\rho_{i-1}(\mathbf{x})$ that depends on \mathbf{x} and is modeled as a polynomial or GP, but such treatment can be computationally prohibitive, especially when Q is large. Most work (e.g., [9-12, 17, 18, 36]) assumes a unit scaling factor $\rho_{i-1} = 1$ for each i to simplify the calculation. When physical experiments $y^e(\mathbf{x})$ are affordable and considered to be the highest-fidelity data, it can be handled via a similar hierarchical model but with an additional random error term ε to represent the experimental variability, i.e.,

$$\begin{aligned} y^e(\mathbf{x}) &= y^{m\{Q\}}(\mathbf{x}) + \delta_Q(\mathbf{x}) + \varepsilon \\ &= y^{m\{1\}}(\mathbf{x}) + \sum_{i=1}^Q \delta_i(\mathbf{x}) + \varepsilon. \end{aligned} \quad (4)$$

$y^{m\{1\}}(\mathbf{x})$ and all $\delta_i(\mathbf{x})$ are represented as GP models. Derivations and further details regarding the preceding can be found in many of aforementioned references (e.g., [13]).

As Q grows, likely so will the amount of collected data required to fit the models. In light of this, Osio and Amon [4] considered an alternative multi-stage approach to mitigate the computational burden. They first performed bi-model fusion using only the data from the models with the lowest two levels of fidelity to obtain a surrogate model for the second lowest fidelity model. They next combined this surrogate model with the data from the third lowest fidelity model to obtain a new surrogate model for the third lowest fidelity model. Proceeding sequentially, they built updated surrogate models in a hierarchical manner, the primary benefit of which is a reduction in the computational costs.

In some situations, the accuracy of computer simulations can be controlled with real-valued tuning parameters, such as the step size when solving the partial differential equations (PDEs) or the mesh density in finite elements analysis (FEA). Tuo et al. [37] proposed to incorporate the tuning parameters \mathbf{t} as additional input variables for simulation models $y^m(\mathbf{x}, \mathbf{t})$, although they implicitly represent different levels of model fidelity.

3. NONHIERARCHICAL MULTI-MODEL FUSION APPROACHES

Denote all Q computer simulation model responses as a row vector $\mathbf{y}^m(\mathbf{x}) = [y^{m\{1\}}(\mathbf{x}), y^{m\{2\}}(\mathbf{x}), \dots, y^{m\{Q\}}(\mathbf{x})]$, and also denote their corresponding discrepancies as $\boldsymbol{\delta}(\mathbf{x}) = [\delta^{\{1\}}(\mathbf{x}), \delta^{\{2\}}(\mathbf{x}), \dots, \delta^{\{Q\}}(\mathbf{x})]$. It is important to note that for the nonhierarchical situation that we consider in this paper, information about model fidelity is assumed unavailable. Consequently, in contrast to the hierarchical situation of Eqs. (3) and (4), for the nonhierarchical situation here the ordering of the simulation models is arbitrary, and the superscripts $\{1\}, \dots, \{Q\}$ no longer serve as the model fidelity level indicators. However, in our work we still assume there is one high-fidelity data source, and it will usually be physical experimental measurement, which is denoted by $y^e(\mathbf{x})$ and is associated with a random observational error $\varepsilon \sim \mathcal{N}(0, \lambda)$. Alternatively, when physical experiments are not affordable or available, $y^e(\mathbf{x})$ could also represent response data from high-fidelity simulation models (in which case the experiment error ε is set to zero).

For the i th computer model, suppose we have collected a set of M_i simulation response observations $\mathbf{d}^{m\{i\}} = [y^{m\{i\}}(\mathbf{x}_1^{m\{i\}}), y^{m\{i\}}(\mathbf{x}_2^{m\{i\}}), \dots, y^{m\{i\}}(\mathbf{x}_{M_i}^{m\{i\}})]^T$ at input sites $\mathbf{X}^{m\{i\}} = [\mathbf{x}_1^{m\{i\}}, \mathbf{x}_2^{m\{i\}}, \dots, \mathbf{x}_{M_i}^{m\{i\}}]^T$. Also suppose a set of N experimental observations $\mathbf{d}^e = [y^e(\mathbf{x}_1^e), y^e(\mathbf{x}_2^e), \dots, y^e(\mathbf{x}_N^e)]^T$ has been obtained at input sites $\mathbf{X}^e = [\mathbf{x}_1^e, \mathbf{x}_2^e, \dots, \mathbf{x}_N^e]^T$. Previous literature [31, 38, 39] suggested a space-filling experimental design for the input sites in $\mathbf{X}^{m\{i\}}$ and \mathbf{X}^e . Let $\mathbf{d} = [(\mathbf{d}^{m\{1\}})^T, \dots, (\mathbf{d}^{m\{Q\}})^T, (\mathbf{d}^e)^T]^T$ denote all the collected data from all simulation models and experiments together. The aim of multi-model fusion is to integrate these data and make predictions for y^e at L untested input sites $\mathbf{X}^p = [\mathbf{x}_1^p, \mathbf{x}_2^p, \dots, \mathbf{x}_L^p]^T$.

In the GP model, the collected data \mathbf{d} together with the to-be-predicted responses $y^e(\mathbf{X}^p)$ follow a multivariate normal distribution

$$\begin{bmatrix} \mathbf{d} \\ y^e(\mathbf{X}^p) \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \mathbf{H} \\ \mathbf{H}_p \end{bmatrix} \boldsymbol{\beta}, \begin{bmatrix} \mathbf{V}_d & \mathbf{T}_p^T \\ \mathbf{T}_p & \mathbf{V}_p \end{bmatrix}\right), \quad (5)$$

where $\boldsymbol{\beta}$ is a column vector of unknown regression coefficients; \mathbf{H} and \mathbf{V}_d are the polynomial and covariance matrices for \mathbf{d} , respectively, and \mathbf{H}_p and \mathbf{V}_p are the polynomial and covariance matrices for $y^e(\mathbf{X}^p)$, respectively. \mathbf{T}_p denotes the covariance matrix between $y^e(\mathbf{X}^p)$ and \mathbf{d} . If we are able to find a reasonable formulation to construct all these matrices \mathbf{H} , \mathbf{H}_p , \mathbf{V}_d , \mathbf{V}_p , \mathbf{T}_p [analogously to what previous research has done in Eqs. (1) and (2) by using parametric forms for $m(\mathbf{x})$ and $V(\mathbf{x}, \mathbf{x}')$ and a model to relate the various types of collected data], and if we are able to estimate the values of the necessary hyperparameters, then the final prediction of $y^e(\mathbf{X}^p)$ and its corresponding mean square error (MSE) can be easily obtained by

$$\begin{aligned} \hat{y}^e(\mathbf{X}^p) &= \mathbf{H}_p \hat{\boldsymbol{\beta}} + \mathbf{T}_p \mathbf{V}_d^{-1} (\mathbf{d} - \mathbf{H} \hat{\boldsymbol{\beta}}), \\ \text{MSE}[\hat{y}^e(\mathbf{X}^p)] &= \mathbf{V}_p - \mathbf{T}_p \mathbf{V}_d^{-1} \mathbf{T}_p^T \\ &\quad + (\mathbf{H}_p^T - \mathbf{H}^T \mathbf{V}_d^{-1} \mathbf{T}_p^T)^T \mathbf{W} (\mathbf{H}_p^T - \mathbf{H}^T \mathbf{V}_d^{-1} \mathbf{T}_p^T), \end{aligned} \quad (6)$$

where $\hat{\boldsymbol{\beta}} = \mathbf{W} \mathbf{H}^T \mathbf{V}_d^{-1} \mathbf{d}$ is an estimate of $\boldsymbol{\beta}$, and $\mathbf{W} = (\mathbf{H}^T \mathbf{V}_d^{-1} \mathbf{H})^{-1}$.

Therefore, the key consideration in multi-model fusion is what formulation we should choose for the overarching model to represent the relationships between the data from each of the individual simulation models and from the physical experiments; the specific nature of the response predictor depends heavily on this choice of formulation, as seen from Eq. (6). In light of this, we develop, investigate, and compare three new nonhierarchical multi-model fusion approaches as described in the following subsections.

3.1 Model Formulations and Assumptions

3.1.1 Approach 1: Weighted-Sum, WS

Consider the true physical response $y^f(\mathbf{x}) = y^e(\mathbf{x}) - \varepsilon$, i.e., which represents the experimental response $y^e(\mathbf{x})$ but without the observational error ε . In this approach we

model $y^t(\mathbf{x})$ as a linear combination of simulation models $\mathbf{y}^m(\mathbf{x})$ together with a single discrepancy function $\delta(\mathbf{x})$, i.e.,

$$y^t(\mathbf{x}) = y^e(\mathbf{x}) - \varepsilon = \mathbf{y}^m(\mathbf{x})\boldsymbol{\rho} + \delta(\mathbf{x}), \quad (7)$$

where $\boldsymbol{\rho} = [\rho^{(1)}, \dots, \rho^{(Q)}]^T$ is a column vector of unknown weight parameters that will be estimated from the data, with each entry corresponding to one simulation model. $\delta(\mathbf{x})$ is a residual function that captures the discrepancy between the weighted sum $\mathbf{y}^m(\mathbf{x})\boldsymbol{\rho}$ and $y^t(\mathbf{x})$. Notice that $\delta(\mathbf{x})$ is different than the elements of $\boldsymbol{\delta}(\mathbf{x})$, which are the individual discrepancy functions for each simulation model.

Following the work of Kennedy and O'Hagan [5], and analogous to the models in Eqs. (3) and (4), we assume in this approach that the simulation models $\mathbf{y}^m(\mathbf{x})$, the residual discrepancy function $\delta(\mathbf{x})$ and the experimental error ε are a priori independent, i.e. that $\text{Cov}(y^{m(i)}(\mathbf{x}), \delta(\mathbf{x}')) = 0$, $\text{Cov}(y^{m(i)}(\mathbf{x}), \varepsilon) = 0$, and $\text{Cov}(\delta(\mathbf{x}), \varepsilon) = 0$ for all \mathbf{x}, \mathbf{x}' , and $i \in [1, 2, \dots, Q]$. These independence assumptions will simplify many calculations in this approach.

3.1.2 Approach 2: Parallel Combination, PC

In this approach, we represent $y^t(\mathbf{x})$ as the sum of a simulation model $y^{m(i)}(\mathbf{x})$ and its corresponding discrepancy function $\delta^{(i)}(\mathbf{x})$, separately for *each* simulation model:

$$y^t(\mathbf{x}) = y^e(\mathbf{x}) - \varepsilon = y^{m(i)}(\mathbf{x}) + \delta^{(i)}(\mathbf{x}) \quad (i = 1, \dots, Q). \quad (8)$$

We propose two variations, (a) and (b), of this approach with different assumptions to represent the relationships between the simulation models and discrepancy functions.

(a) Discrepancy Independent of the True Response (DIT)

We define the DIT assumption as each model discrepancy $\delta^{(i)}(\mathbf{x})$ is independent of the true physical process $y^t(\mathbf{x})$, i.e. $\text{Cov}(\delta^{(i)}(\mathbf{x}), y^t(\mathbf{x}')) = 0$, for all \mathbf{x}, \mathbf{x}' , and $i \in [1, 2, \dots, Q]$.

There is a closely-related alternative to the DIT assumption that one may be tempted to consider, which, for the reasons discussed below, is not appropriate in this approach. Specifically, in the co-kriging literature [16, 35, 40-43] with a single computer model, it

is common to assume that the simulation model $y^m(\mathbf{x})$ is independent of the discrepancy function $\delta(\mathbf{x})$. For our model formulation as Eq. (8), if we had assumed that $\text{Cov}(y^{m\{i\}}(\mathbf{x}), \delta^{j\}(\mathbf{x}')) = 0$ (for all \mathbf{x}, \mathbf{x}' , and $i, j \in [1, 2, \dots, Q]$), this would imply that for any $i \neq j$,

$$\begin{aligned} 0 &= \text{Cov}\left(y^{m\{i\}}(\mathbf{x}), \delta^{j\}(\mathbf{x}')\right) = \text{Cov}\left(y^{m\{i\}}(\mathbf{x}), y^{m\{i\}}(\mathbf{x}') + \delta^{i\}(\mathbf{x}') - y^{m\{j\}}(\mathbf{x}')\right) \\ &= \text{Cov}\left(y^{m\{i\}}(\mathbf{x}), y^{m\{i\}}(\mathbf{x}')\right) - \text{Cov}\left(y^{m\{i\}}(\mathbf{x}), y^{m\{j\}}(\mathbf{x}')\right). \end{aligned} \quad (9)$$

Since i and j are interchangeable, so are \mathbf{x} and \mathbf{x}' , we yield that $\text{Cov}(y^{m\{i\}}(\mathbf{x}), y^{m\{i\}}(\mathbf{x}')) = \text{Cov}(y^{m\{i\}}(\mathbf{x}), y^{m\{j\}}(\mathbf{x}')) = \text{Cov}(y^{m\{j\}}(\mathbf{x}), y^{m\{j\}}(\mathbf{x}'))$, which implies that any two simulation models are perfectly correlated (i.e., their correlation coefficient is 1) and with the same variance. In other words, this assumption yields that all Q models are identical and only differ from each other by a constant. This is obviously an unreasonable assumption for engineering applications (if the computer models are identical up to a constant, they are essentially the same model, which defeats the purpose of having multiple models). Even if we relax the foregoing assumption so that only $\text{Cov}(y^{m\{i\}}(\mathbf{x}), \delta^{i\}(\mathbf{x}')) = 0$ for all $i \in [1, 2, \dots, Q]$ (i.e., a model is only independent of its corresponding discrepancy function but not the other discrepancy functions), this will result in an indefinite covariance matrix \mathbf{V}_a in Eq. (5), which can result in a physically-impossible negative prediction MSE.

In light of the preceding, our assumption that all model discrepancy functions are independent of the true physical process is more appropriate. Model discrepancy arises from the modeling process and is different with respect to various simulation models, while the true physical process stays the same regardless of the models.

(b) Common Spatial Correlation (CSC)

In this variant of the PC approach, we do not make any independency assumptions between the simulation models, their discrepancy functions, and the true physical process. Rather, a fully correlated multi-response GP (MRGP) structure is adopted for all simulation models and the discrepancy functions, which is expected to provide more flexibility for integrating different responses from computer simulations and experimental observations, because all relationships between them are modeled and

empirically estimated from the data. To simplify the implementation, all simulation models and their discrepancy functions are assumed to share the same spatial correlation function [i.e., the same $R(\mathbf{x}, \mathbf{x}')$, as defined right after Eq. (1)]. This would often be a reasonable assumption, as the spatial correlation function represents the spatial roughness/smoothness of the response surface, and all responses (true physics, simulation, and their differences) are modeling the same physical phenomenon and, hence, may be expected to have similar roughness/smoothness.

3.2 SRP Modeling for Different Responses

3.2.1 Approach 1: WS

An MRGP model is fit to the simulation responses. Based on the previous work [40, 41, 43], the prior for this MRGP model can be denoted as:

$$\mathbf{y}^m(\bullet) \sim \mathcal{GP}\left(\mathbf{h}^m(\bullet)^T \mathbf{B}^m, \Sigma^m R^m(\bullet, \bullet)\right), \quad (10)$$

where $\mathbf{h}^m(\mathbf{x})$ is the polynomial basis vector defined as Section 2.1 associated with a matrix of unknown regression coefficients $\mathbf{B}^m = [\boldsymbol{\beta}^{m\{1\}}, \boldsymbol{\beta}^{m\{2\}}, \dots, \boldsymbol{\beta}^{m\{Q\}}]$, where $\boldsymbol{\beta}^{m\{i\}} = [\beta_1^{m\{i\}}, \beta_2^{m\{i\}}, \dots, \beta_p^{m\{i\}}]^T$, for polynomial regression of the mean function of the simulation models. The prior covariance function is assumed to be the product of a non-spatial $Q \times Q$ covariance matrix Σ^m and a spatial correlation function $R^m(\mathbf{x}, \mathbf{x}')$, the parameters for both of which must be estimated from the data. That is, the covariance between two models at two input sites is $\text{Cov}[y^{m\{i\}}(\mathbf{x}), y^{m\{j\}}(\mathbf{x}')] = \Sigma_{i,j}^m R^m(\mathbf{x}, \mathbf{x}')$, where $\Sigma_{i,j}^m$ is the covariance between the i th and j th simulation model responses at the same spatial location ($\mathbf{x} = \mathbf{x}'$). We adopt the Gaussian correlation function of Eq. (2), parameterized by the roughness parameters $\boldsymbol{\omega}^m$. Thus, the MRGP hyperparameters for all the simulation models are $\boldsymbol{\phi}_1^m = \{\mathbf{B}^m, \Sigma^m, \boldsymbol{\omega}^m\}$.

The residual discrepancy function $\delta(\mathbf{x})$ is also represented as a GP model:

$$\delta(\bullet) \sim \mathcal{GP}\left(\mathbf{h}^\delta(\bullet)^T \boldsymbol{\beta}^\delta, \sigma_\delta^2 R^\delta(\bullet, \bullet)\right). \quad (11)$$

Similarly, the hyperparameters of the discrepancy function GP model are $\phi_1^\delta = \{\boldsymbol{\beta}^\delta, \sigma_\delta, \boldsymbol{\omega}^\delta\}$. According to Eq. (7) and the a priori independency assumption between $\mathbf{y}^m(\mathbf{x})$, $\delta(\mathbf{x})$ and ε (Sec. 3.1.1), the priors for the simulation MRGP model and the residual discrepancy GP model are combined to form the prior for the experimental response GP model:

$$y^e(\bullet) \sim \mathcal{GP}\left(\mathbf{h}^m(\bullet)^T \mathbf{B}^m \boldsymbol{\rho} + \mathbf{h}^\delta(\bullet)^T \boldsymbol{\beta}^\delta, \boldsymbol{\rho}^T \boldsymbol{\Sigma}^m \boldsymbol{\rho} R^m(\bullet, \bullet) + \sigma_\delta^2 R^\delta(\bullet, \bullet) + \lambda I(\bullet, \bullet)\right), \quad (12)$$

where $I(\mathbf{x}, \mathbf{x}')$ is the indicator function defined as 1 when $\mathbf{x} = \mathbf{x}'$ and 0, otherwise. Let $\boldsymbol{\phi} = \{\phi_1^m, \phi_1^\delta, \boldsymbol{\rho}, \lambda\}$ denote the collection of the unknown hyperparameters, estimation of which is discussed in Section 3.3.

Based on the assumptions of model formulation as Eq. (7), the covariance of the experimental response and the i th simulation model response is

$$\begin{aligned} \text{Cov}\left(y^e(\mathbf{x}), y^{m\{i\}}(\mathbf{x}')\right) &= \text{Cov}\left(\mathbf{y}^m(\mathbf{x})\boldsymbol{\rho} + \delta(\mathbf{x}) + \varepsilon, \mathbf{y}^m(\mathbf{x}')e_i\right) \\ &= \text{Cov}\left(\mathbf{y}^m(\mathbf{x})\boldsymbol{\rho}, \mathbf{y}^m(\mathbf{x}')e_i\right) = \boldsymbol{\rho}^T \boldsymbol{\Sigma}^m e_i R^m(\mathbf{x}, \mathbf{x}'), \end{aligned} \quad (13)$$

where e_i is a column unit vector, whose i th entry is one and other entries are zero. Similarly, the covariance of the i th and the j th simulation model responses is

$$\text{Cov}\left(y^{m\{i\}}(\mathbf{x}), y^{m\{j\}}(\mathbf{x}')\right) = e_i^T \boldsymbol{\Sigma}^m e_j R^m(\mathbf{x}, \mathbf{x}'), \quad (14)$$

where $e_i^T \boldsymbol{\Sigma}^m e_j = \Sigma_{i,j}^m$ is simply the (i,j) th entry of the simulation MRGP $\boldsymbol{\Sigma}^m$.

3.2.2 Approach 2(a): PC-DIT

In this approach, we model the true physical process as a GP with hyperparameters $\phi_2^t = \{\boldsymbol{\beta}^t, \sigma_t, \boldsymbol{\omega}^t\}$ and the discrepancy functions as an MRGP with hyperparameters $\phi_2^\delta = \{\mathbf{B}^\delta, \boldsymbol{\Sigma}^\delta, \boldsymbol{\omega}^\delta\}$,

$$\begin{aligned} y^t(\bullet) &\sim \mathcal{GP}\left(\mathbf{h}^t(\bullet)^T \boldsymbol{\beta}^t, \sigma_t^2 R^t(\bullet, \bullet)\right), \\ \delta(\bullet) &\sim \mathcal{GP}\left(\mathbf{h}^\delta(\bullet)^T \mathbf{B}^\delta, \boldsymbol{\Sigma}^\delta R^\delta(\bullet, \bullet)\right). \end{aligned} \quad (15)$$

The hyperparameters in ϕ_2^t and ϕ_2^δ are defined similarly to ϕ_1^δ and ϕ_1^m in Eqs. (11) and (10), respectively, of the WS approach. $R^t(\mathbf{x}, \mathbf{x}')$ and $R^\delta(\mathbf{x}, \mathbf{x}')$ are the Gaussian correlation

functions for the true physical process GP model and the discrepancy MRGP model, respectively. Combining Eqs. (8) and (15), the GP model for the experimental response is

$$y^e(\bullet) \sim \mathcal{GP}\left(\mathbf{h}^t(\bullet)^T \boldsymbol{\beta}^t, \sigma_t^2 R^t(\bullet, \bullet) + \lambda I(\bullet, \bullet)\right). \quad (16)$$

Let $\boldsymbol{\phi} = \{\boldsymbol{\phi}_2^t, \boldsymbol{\phi}_2^\delta, \lambda\}$ denote the collection of yet-to-be-estimated hyperparameters.

Based on Eq. (8) and the DIT assumption, we can derive the covariance formulations between the experiment, the simulation models, and their corresponding discrepancies, which are required in the subsequent hyperparameter estimation and response prediction, as follows. For all $i, j \in [1, 2, \dots, Q]$,

$$\begin{aligned} \text{Cov}\left(y^{m\{i\}}(\mathbf{x}), y^{m\{j\}}(\mathbf{x}')\right) &= \text{Cov}\left(y^t(\mathbf{x}) - \delta^{\{i\}}(\mathbf{x}), y^t(\mathbf{x}') - \delta^{\{j\}}(\mathbf{x}')\right) \\ &= \sigma_t^2 R^t(\mathbf{x}, \mathbf{x}') + \boldsymbol{\Sigma}_{i,j}^\delta R^\delta(\mathbf{x}, \mathbf{x}'), \\ \text{Cov}\left(y^{m\{i\}}(\mathbf{x}), y^e(\mathbf{x}')\right) &= \text{Cov}\left(y^t(\mathbf{x}) - \delta^{\{i\}}(\mathbf{x}), y^t(\mathbf{x}') + \varepsilon\right) = \sigma_t^2 R^t(\mathbf{x}, \mathbf{x}'), \\ \text{Cov}\left(\delta^{\{i\}}(\mathbf{x}), y^{m\{j\}}(\mathbf{x}')\right) &= \text{Cov}\left(\delta^{\{i\}}(\mathbf{x}), y^t(\mathbf{x}') - \delta^{\{j\}}(\mathbf{x}')\right) = -\boldsymbol{\Sigma}_{i,j}^\delta R^\delta(\mathbf{x}, \mathbf{x}'), \\ \text{Cov}\left(\delta^{\{i\}}(\mathbf{x}), y^e(\mathbf{x}')\right) &= \text{Cov}\left(\delta^{\{i\}}(\mathbf{x}), y^t(\mathbf{x}') + \varepsilon\right) = 0, \end{aligned} \quad (17)$$

where $\boldsymbol{\Sigma}_{i,j}^\delta$ is the (i,j) th entry of the discrepancy MRGP $\boldsymbol{\Sigma}^\delta$, representing the covariance between the i th and j th model discrepancies.

3.2.3 Approach 2(b): PC-CSC

We collectively denote the responses from all simulation models and their discrepancies as $\mathbf{a}(\mathbf{x}) = [y^{m\{1\}}(\mathbf{x}), \dots, y^{m\{Q\}}(\mathbf{x}), \delta^{\{1\}}(\mathbf{x}), \dots, \delta^{\{Q\}}(\mathbf{x})] = [\mathbf{y}^m(\mathbf{x}), \boldsymbol{\delta}(\mathbf{x})]$. Note that according to Eq. (8), all the model discrepancy functions apart from $\delta^{\{1\}}(\mathbf{x})$ can be expressed as:

$$\delta^{\{i\}}(\mathbf{x}) = y^{m\{1\}}(\mathbf{x}) + \delta^{\{1\}}(\mathbf{x}) - y^{m\{i\}}(\mathbf{x}), \quad (i = 2, \dots, Q). \quad (18)$$

Therefore, some elements of $\mathbf{a}(\mathbf{x})$ are redundant (completely determined by other elements), so we only need to consider a subset $\mathbf{c}(\mathbf{x}) = [\mathbf{y}^m(\mathbf{x}), \delta^{\{1\}}(\mathbf{x})]$ of $\mathbf{a}(\mathbf{x})$. The complete set $\mathbf{a}(\mathbf{x})$ can be recovered from

$$\mathbf{a}(\mathbf{x})^T = \mathbf{G}\mathbf{c}(\mathbf{x})^T, \quad (19)$$

$$\mathbf{G} = \begin{bmatrix} \mathbf{I}_{(Q+1) \times (Q+1)} \\ \mathbf{1}_{(Q-1) \times 1} & -\mathbf{I}_{(Q-1) \times (Q-1)} & \mathbf{1}_{(Q-1) \times 1} \end{bmatrix},$$

where \mathbf{I} and $\mathbf{1}$ are the identity matrix and a matrix of ones, respectively, whose dimensionalities are denoted by their subscripts.

The reduced set of responses $\mathbf{c}(\mathbf{x})$ can be modeled as an MRGP

$$\mathbf{c}(\cdot) \sim \mathcal{GP}(\mathbf{m}^c(\cdot), \Sigma^c R^c(\cdot, \cdot)), \quad (20)$$

where $\mathbf{m}^c(\mathbf{x}) = [\mathbf{h}^m(\mathbf{x})^T \mathbf{B}^m, \mathbf{h}^\delta(\mathbf{x})^T \boldsymbol{\beta}^{\delta\{1\}}]$ is the mean function, and Σ^c is the non-spatial covariance matrix defined as previously. $R^c(\mathbf{x}, \mathbf{x}')$ is the common Gaussian correlation function we assign to both the simulation models and their discrepancy functions.

From Eqs. (19) and (20), the complete set of responses $\mathbf{a}(\mathbf{x})$ also follows an MRGP model

$$\mathbf{a}(\cdot) \sim \mathcal{GP}(\mathbf{m}^c(\cdot) \mathbf{G}^T, \mathbf{G} \Sigma^c \mathbf{G}^T R^c(\cdot, \cdot)) \triangleq \mathcal{GP}(\mathbf{m}^a(\cdot), \Sigma^a R^c(\cdot, \cdot)), \quad (21)$$

and from Eqs. (8) and (21), the GP model for the experimental response is

$$y^e(\cdot) \sim \mathcal{GP}(\mathbf{h}^m(\cdot)^T \boldsymbol{\beta}^{m\{1\}} + \mathbf{h}^\delta(\cdot)^T \boldsymbol{\beta}^{\delta\{1\}}, (\Sigma_{1,1}^a + 2\Sigma_{1,Q+1}^a + \Sigma_{Q+1,Q+1}^a) R^c(\cdot, \cdot) + \lambda I(\cdot, \cdot)). \quad (22)$$

Let $\boldsymbol{\phi} = \{\mathbf{B}^m, \boldsymbol{\beta}^{\delta\{1\}}, \Sigma^c, \boldsymbol{\omega}^c, \lambda\}$ denote the collection of unknown hyperparameters in this approach.

Based on Eq. (8) and the CSC assumption, we can derive the covariance formulations between the experiment and the simulation models as well as the discrepancy functions, which are required in the subsequent hyperparameter estimation and response prediction.

For all $i \in [1, 2, \dots, Q]$, we have

$$\begin{aligned} \text{Cov}(y^e(\mathbf{x}), y^{m\{i\}}(\mathbf{x}')) &= \text{Cov}(y^{m\{1\}}(\mathbf{x}) + \delta^{\{1\}}(\mathbf{x}) + \varepsilon, y^{m\{i\}}(\mathbf{x}')) \\ &= (\Sigma_{1,i}^a + \Sigma_{Q+1,i}^a) R^c(\mathbf{x}, \mathbf{x}'), \\ \text{Cov}(y^e(\mathbf{x}), \delta^{\{i\}}(\mathbf{x}')) &= \text{Cov}(y^{m\{1\}}(\mathbf{x}) + \delta^{\{1\}}(\mathbf{x}) + \varepsilon, \delta^{\{i\}}(\mathbf{x}')) \\ &= (\Sigma_{1,Q+i}^a + \Sigma_{Q+1,Q+i}^a) R^c(\mathbf{x}, \mathbf{x}'). \end{aligned} \quad (23)$$

3.3 Estimation of Hyperparameters for SRP Models

Because all simulation models and experiment are modeled as GPs, the joint Gaussian distribution for the collected data \mathbf{d} from their responses can be expressed as,

$$\mathbf{d} \sim \mathcal{N}(\mathbf{H}\boldsymbol{\beta}, \mathbf{V}_d), \quad (24)$$

which is contained in Eq. (5). Each of $\boldsymbol{\beta}$, \mathbf{H} and \mathbf{V}_d are constructed as functions of the to-be-estimated hyperparameters $\boldsymbol{\phi}$, details of which for the different approaches can be found in Appendix A. The MLEs of $\boldsymbol{\phi}$ are obtained by maximizing the log-likelihood function

$$l(\{\boldsymbol{\phi} \setminus \boldsymbol{\beta}\} | \mathbf{d}) = \frac{1}{2} \log |\mathbf{W}| - \frac{1}{2} \log |\mathbf{V}_d| - \frac{1}{2} (\mathbf{d} - \mathbf{H}\hat{\boldsymbol{\beta}})^T \mathbf{V}_d^{-1} (\mathbf{d} - \mathbf{H}\hat{\boldsymbol{\beta}}) + \text{constant}, \quad (25)$$

using numerical optimization, e.g., a genetic algorithm or simulated annealing. Notice that in the right-hand-side of Eq. (25), we have used the closed-form expression $\hat{\boldsymbol{\beta}} = \mathbf{W}\mathbf{H}^T \mathbf{V}_d^{-1} \mathbf{d}$ for the MLE of $\boldsymbol{\beta}$ [see Eq. (6)].

3.4 SRP-Based Response Predictions

After estimating all the hyperparameters, we can predict the experimental response $y^e(\mathbf{X}^p)$ at any new input site \mathbf{X}^p via Eq. (6), using the MLEs of $\boldsymbol{\phi}$. Details regarding how to construct the matrices required in Eq. (6) can be found in Appendix B.

We can also predict the model discrepancy $\boldsymbol{\delta}(\mathbf{X}^p)$ [or $\delta(\mathbf{X}^p)$ in the WS approach] at the untested input site \mathbf{X}^p analogous to how we predicted the experimental response $y^e(\mathbf{X}^p)$. That is, we first express the joint Gaussian distribution of \mathbf{d} and $\boldsymbol{\delta}(\mathbf{X}^p)$ as [analogous to Eq. (5)]

$$\begin{bmatrix} \mathbf{d} \\ \text{vec}(\boldsymbol{\delta}(\mathbf{X}^p)) \end{bmatrix} \sim \mathcal{N} \left(\begin{bmatrix} \mathbf{H} \\ \tilde{\mathbf{H}}_p \end{bmatrix} \boldsymbol{\beta}, \begin{bmatrix} \mathbf{V}_d & \tilde{\mathbf{T}}_p^T \\ \tilde{\mathbf{T}}_p & \tilde{\mathbf{V}}_p \end{bmatrix} \right), \quad (26)$$

where $\text{vec}(\cdot)$ is the vectorized matrix (by stacking its columns). $\tilde{\mathbf{H}}_p$ and $\tilde{\mathbf{V}}_p$ are the polynomial and covariance matrices for $\boldsymbol{\delta}(\mathbf{X}^p)$, respectively. $\tilde{\mathbf{T}}_p$ denotes the covariance matrix between $\boldsymbol{\delta}(\mathbf{X}^p)$ and \mathbf{d} . Details of these matrices for the different approaches can be

found in Appendix C. Then, the posterior mean prediction of $\delta(\mathbf{X}^p)$ and its MSE are given by [analogous to Eq. (6)]

$$\begin{aligned} \text{vec}(\hat{\delta}(\mathbf{X}^p)) &= \tilde{\mathbf{H}}_p \hat{\boldsymbol{\beta}} + \tilde{\mathbf{T}}_p \mathbf{V}_d^{-1} (\mathbf{d} - \mathbf{H} \hat{\boldsymbol{\beta}}), \\ \text{MSE} \left[\text{vec}(\hat{\delta}(\mathbf{X}^p)) \right] &= \tilde{\mathbf{V}}_p - \tilde{\mathbf{T}}_p \mathbf{V}_d^{-1} \tilde{\mathbf{T}}_p^T \\ &\quad + (\tilde{\mathbf{H}}_p^T - \mathbf{H}^T \mathbf{V}_d^{-1} \tilde{\mathbf{T}}_p^T)^T \mathbf{W} (\tilde{\mathbf{H}}_p^T - \mathbf{H}^T \mathbf{V}_d^{-1} \tilde{\mathbf{T}}_p^T). \end{aligned} \quad (27)$$

Note that once the unknown hyperparameters ϕ are estimated, the computational cost of response prediction is negligible, using the analytical expressions in Eqs. (6) and (27). Therefore, our proposed approaches are also applicable in other domains, such as design optimization and uncertainty analysis.

4. CASE STUDIES

In this section, three numerical examples and a six-dimensional engineering application (Sec. 4.4) are investigated to test the performance of the three proposed nonhierarchical multi-model fusion approaches. The numerical examples feature different scenarios, such as “clearly different model fidelities” (Sec. 4.1), “similar model fidelities” (Sec. 4.2), and “range-dependent model fidelities” (Sec. 4.3). Their mathematical expressions can be found in Appendix D. For the purpose of illustration, all examples comprise of two simulation models and the highest fidelity data from experiments.

To compare the effectiveness and relative merits of the proposed approaches, we perform a validation test in each case study using two metrics: (1) root-mean-square error (RMSE) for assessing the accuracy of the *mean* prediction of the updated model, and (2) u-pooling for measuring the *overall* model prediction capability in a manner that considers its ability to quantify the effects of both interpolation uncertainty and experimental variability. RMSE is one of the most popular validation metrics to assess the mean predictive capability of a model, but it does not assess the model’s ability to accurately quantify the uncertainty in the predictions which is important for

understanding how close the predicted value is to the true value. With our proposed multi-model fusion approaches, there are both interpolation uncertainty associated with lack of data that can be quantified using the SRP prediction error (i.e., MSE) and the measurement error introduced by experimental observations. To account for the uncertainties in both the updated model and the experiments, the u-pooling metric [44-46] is widely used to measure the *overall* prediction performance of different approaches. The u-pooling metric was developed based on the idea of the area metric [44, 47] to measure the agreement between entire distributions of model predictions and uncertain experimental observations when the data is sparse at multiple validation sites. Based on the probability integral transform (PIT), an u-value u_i for each experimental observation $y^e(\mathbf{x}_i)$ is calculated using the cumulative distribution function (CDF) of the prediction distribution $F_{\mathbf{x}_i}^m(\bullet)$ at the validation site \mathbf{x}_i as

$$u_i = F_{\mathbf{x}_i}^m(y^e(\mathbf{x}_i)) \quad (28)$$

Subsequently, a distribution of all the u-values u can be characterized using the empirical CDF. As stated in [44], if the prediction distribution is identical to the actual experimental distribution, then the u values should follow a standard uniform distribution within $[0, 1]$. By calculating the area difference of the empirical distribution of u and that of the standard uniform distribution, the u-pooling metric can quantify the mismatch or dispersion of the distributions of responses from both simulated prediction and experimental test in a global sense. The value of the u-pooling metric is between 0 and 0.5; a larger area difference indicates less agreement, and hence a less accurate simulation model.

4.1 Numerical Example 1: Clearly Different Model Fidelities

This one-dimensional example is used to demonstrate the effectiveness of our proposed multi-model fusion approaches for the situation where the fidelity levels of the simulation

models are clearly different, which is a common assumption for the existing hierarchical approaches in the literature. We consider it as a baseline test example. The low-fidelity model 1 with 5 samples (green triangles) and the high-fidelity model 2 with another 5 samples (blue squares) are shown in Fig. 2(a). Additionally, four (4) experimental observations (red dots) are collected. The true physical response is plotted as a reference. Note that if the input locations for one simulation model are spaced tightly in some input regions, the covariance matrix \mathbf{V}_d might be nearly singular. This can be addressed by using the design of experiment (DOE) methods to avoid the clustering of input locations, or by adding a small nugget effect in the model to allow the invertibility of \mathbf{V}_d [48]. In our examples, the input locations for simulating each model are randomly chosen to be space-filling within the whole input space.

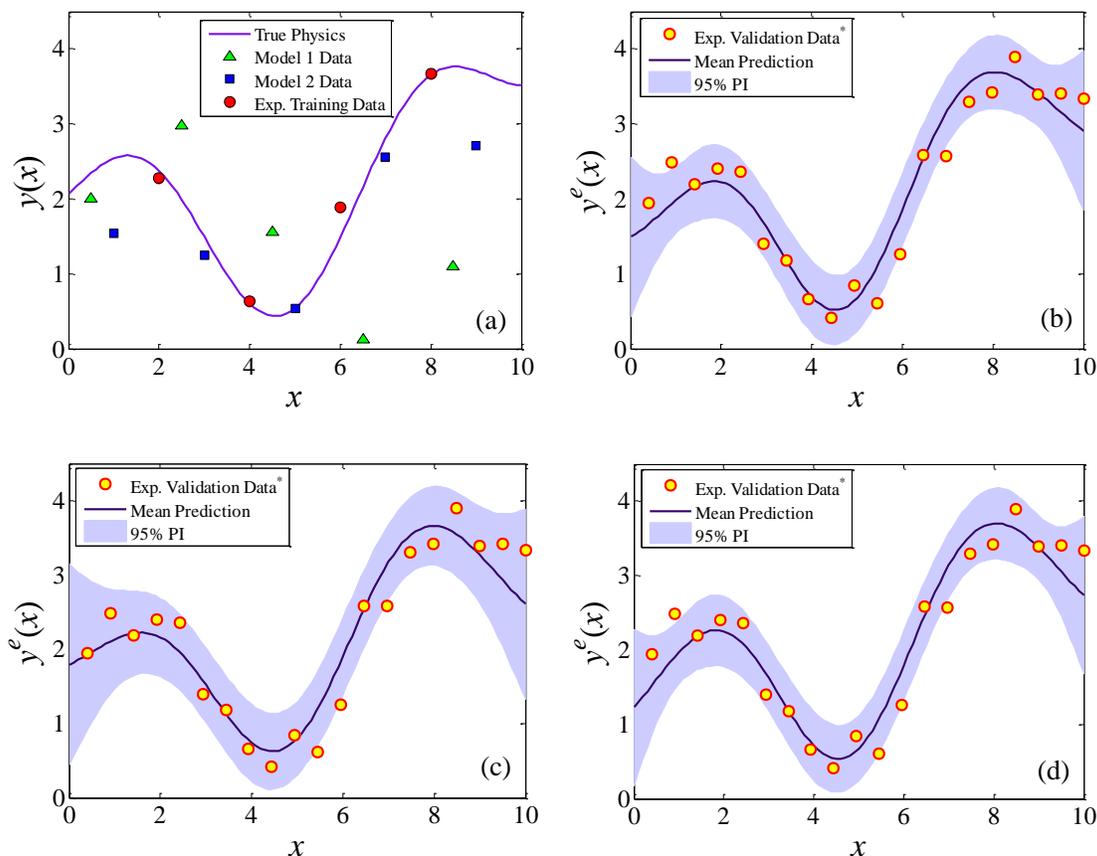


Fig. 2 An example with differentiable model fidelities. True physics and collected data are shown in (a). Validation points*, the mean prediction, and the 95% prediction interval (PI) after multi-model fusion using (b) WS approach, (c) PC-DIT approach, and (d)

PC-CSC approach are plotted.

* For illustration, we only plot evenly 20 out of 100 total validation points.

The results from the three proposed multi-model fusion approaches are shown in Fig. 2(b)~(d). For validation and comparison, 100 additional experimental data are chosen evenly in the input space. From visual comparison, all three approaches handle this example well. They provide similar mean predictions for the experimental responses, and most of the validation points are located within the 95% prediction interval (PI) with only a few exceptions. The PC-DIT approach performs a little better in both of the edge regions with relatively larger MSE that seems to represent the randomness of the experimental response better.

For a rigorous and quantitative comparison, the RMSE and u-pooling metrics for different approaches are listed in Table 1.

Table 1: Metrics for example 1 with differentiable model fidelities

Approach	WS	PC-DIT	PC-CSC
RMSE	0.3448	0.3600	0.3697
u-pooling	0.0639	0.0505	0.0724

The WS approach excels slightly on mean prediction with the smallest RMSE value, but its u-pooling metric is worse than the PC-DIT approach. The PC-CSC approach has the worst mean (RMSE) and overall (u-pooling) prediction performances. But in consideration of the small difference between the metric values, all three multi-model fusion approaches perform almost equally well in this example.

4.2 Numerical Example 2: Similar Model Fidelities

The second example is one-dimensional as well, but the fidelity levels of the simulation models cannot be explicitly identified, as shown in Fig. 3(a), thus the existing hierarchical multi-fidelity modeling approaches are infeasible in this situation. The

collected data contain three (3) samples (green triangles) from the model 1, seven (7) samples (blue squares) from the model 2, and three (3) additional experiment observations (red dots).

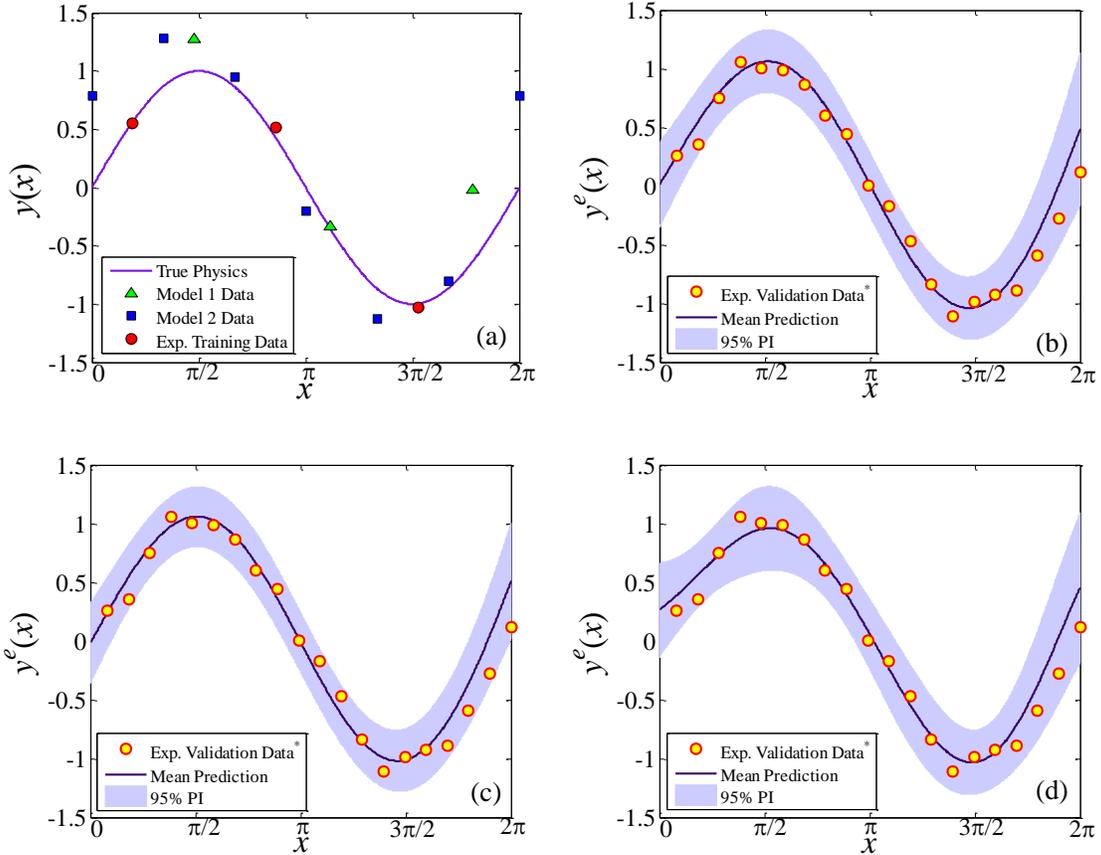


Fig. 3 An example with similar model fidelities. True physics and known data are collected in (a). Validation points*, the mean prediction, and the 95% PI after multi-model fusion using (b) WS approach, (c) PC-DIT approach, and (d) PC-CSC approach are plotted.

* For illustration, we only plot evenly 20 out of 100 total validation points.

The final predictions for experiment response using the three proposed approaches, shown in Fig. 3(b)~(d) illustrate that they work well to address the lack-of-fidelity-identification situation as their mean predictions match well with the true physical response in most of the input region. The validation points are all bounded within the 95% prediction intervals of the three approaches.

Table 2: Metrics for example 2 with similar model fidelities

Approach	WS	PC-DIT	PC-CSC
RMSE	0.1530	0.1636	0.1573
u-pooling	0.0805	0.0786	0.0924

100 additional experiment observations distributed evenly in the input space are used for confirmation (see Table 2). In terms of the mean prediction, the WS approach outperforms slightly, but the similar RMSE values indicate no statistically significant difference between the three approaches. The WS and PC-DIT approaches achieve the same distribution agreement between model predictions and experiments with similar u-pooling metric values, and they both perform slightly better than the PC-CSC approach, which provides slightly worse predictions around the left edge of the input region, as shown in Fig. 3(d).

4.3 Numerical Example 3: Range-Dependent Model Fidelities

In this one-dimensional example, there are 5 simulation observations from each model and 4 experimental observations, as shown in Fig. 4(a). The fidelity levels of the simulation models change dramatically over the input space, such that model 1 performs better within the left region $[0, 5]$, while model 2 is much more trustworthy within the right region $[5, 10]$. Thus we cannot differentiate the level of fidelity in a global sense for each model. Such problems with range-dependent model fidelity hamper the feasibility of the existing hierarchical approaches for multi-model fusion.

Fig. 4(b)~(d) show the final predictions of the experiment response using our three proposed approaches, respectively. It can be judged visually that the PC-DIT approach performs best with accurate mean prediction and small MSE over the whole input space. The PC-CSC approach produces satisfactory mean prediction as well, but it fades a little with relatively larger MSE at both edge regions of the input space. All validation points are bounded within the 95% PIs of the PC-DIT and PC-CSC approaches. Although small

MSE can be achieved using the WS approach, it suffers from large discrepancies in mean prediction, as shown in Fig. 4(b) that some validation points are out of the 95% PI.

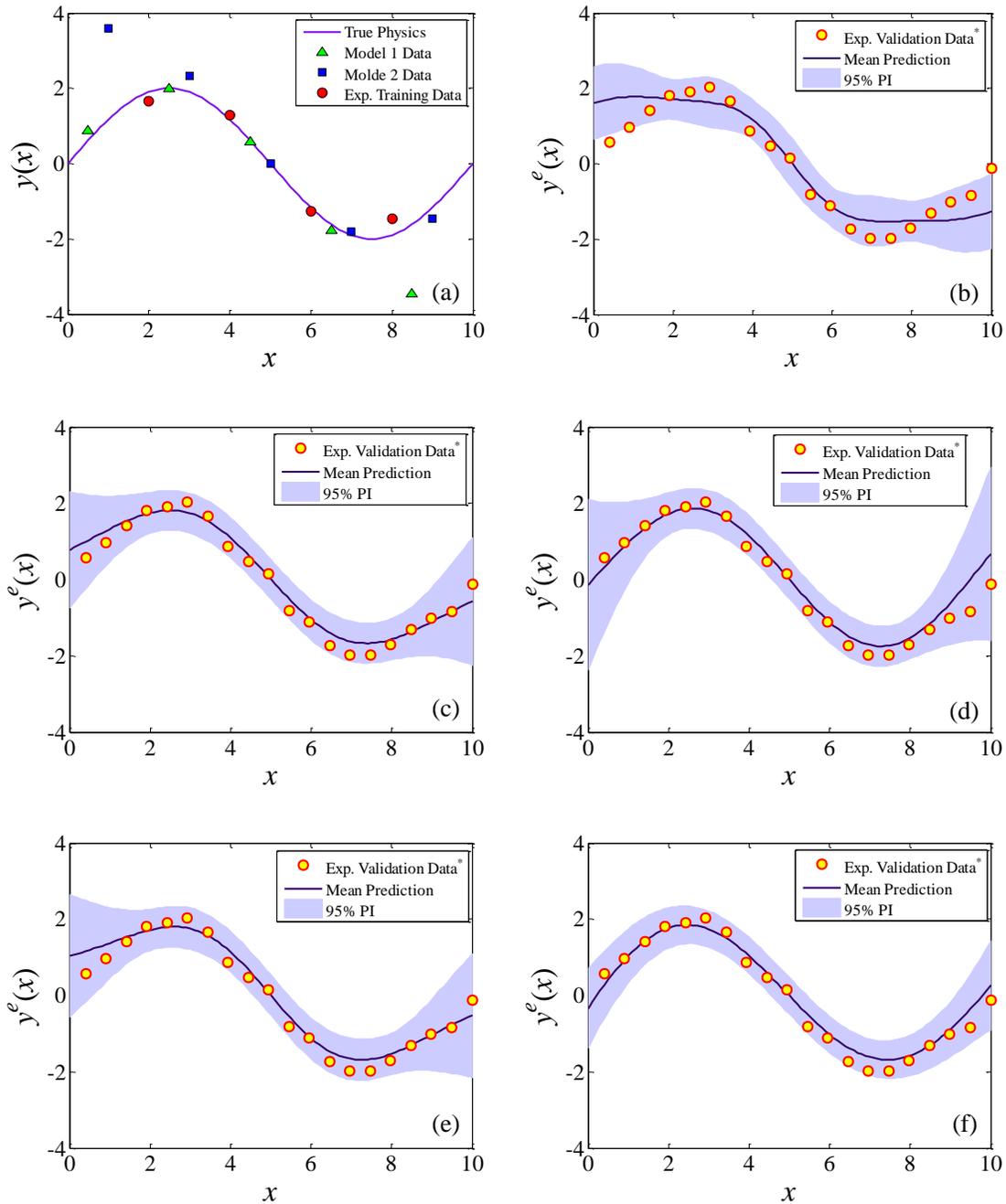


Fig. 4 An example with range-dependent model fidelities. True physics and collected data are shown in (a). Validation points*, the mean prediction, and the 95% PI after multi-model fusion using (b) WS approach, (c) PC-DIT approach, (d) PC-CSC approach, (e) WS approach with $0.01 < \omega^{\mathcal{D}} < 10$, and (f) WS approach with $0.01 < \omega^{\mathcal{D}} < 5$ are plotted.

* For illustration, we only plot evenly 20 out of 100 total validation points.

The RMSE and u-pooling metrics of the three proposed approaches are computed and listed in the first three columns in Table 3, respectively, with 100 evenly distributed validation points. From their small RMSE and u-pooling metrics, we can find that the PC-DIT and PC-CSC approaches are able to handle this range-dependent model fidelity case well. The RMSE metric of the WS approach is relatively large, indicating a worse mean prediction performance than the other two approaches, which is consistent with the preceding visual comparison. However, its u-pooling metric is slightly better, because it overestimates the mean prediction in the left input region while underestimates it in the right region. This results in a relatively more uniform u in the u-pooling calculation.

Table 3: Metrics for example 3 with range-dependent model fidelities

Approach	WS with $0.01 < \omega^\delta < 50$	PC-DIT	PC-CSC	WS with $0.01 < \omega^\delta < 10$	WS with $0.01 < \omega^\delta < 5$
RMSE	0.5692	0.3329	0.3542	0.3598	0.2996
u-pooling	0.0797	0.0952	0.0966	0.1035	0.0823

It should be noted that the WS approach fades with unsatisfactory predictions in such a range-dependent model fidelity case. In order to overcome this difficulty, we tested a smaller prior upper bound for the roughness parameter ω^δ of the residual discrepancy function $\delta(x)$ when estimating the MLEs (originally the prior range of ω^δ is set to be $0.01 < \omega^\delta < 50$). The experiment predictions from the WS approach but with smaller prior ranges as $0.01 < \omega^\delta < 10$ and $0.01 < \omega^\delta < 5$ are shown in Fig. 4(e) and (f), respectively. Their corresponding RMSE and u-pooling metrics are listed in the last two columns in Table 3. The metric values, after reducing the prior upper bound of ω^δ , are fairly small and competitive with those of the PC-DIT and PC-CSC approaches. Hence, the smaller discrepancy roughness strategy works effectively to mitigate the adverse effect from the global-sense weight parameter estimations in the WS approach.

The predictions of the residual discrepancy function $\delta(x)$ from the WS approach with

$0.01 < \omega^\delta < 50$ and $0.01 < \omega^\delta < 5$ are shown in Fig. 5(a) and (b), respectively. It can be found that the discrepancy with smaller prior upper bound for ω^δ is quite smooth, which meets with our expected performance. For this example, both simulation models overall play approximately equal roles in the multi-model fusion and predict the experiment response well in different halves of the input space. Thus close global-sense weight parameters would be estimated, which result in large discrepancy at both edge regions of the input space and smaller discrepancy in the central region, as shown in Fig. 5(b). Therefore, assigning a priori knowledge of smaller roughness parameter is anticipated to capture this smooth discrepancy and to improve the prediction performance of multi-model fusion using the WS approach.

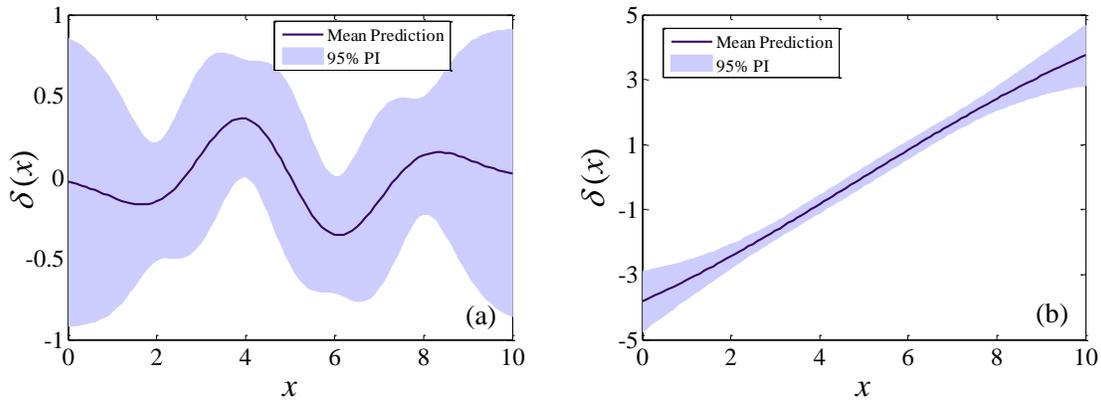


Fig. 5 The mean prediction and 95% PI of the residual discrepancy function $\delta(x)$ after multi-model fusion using the WS approach with (a) $0.01 < \omega^\delta < 50$, and (b) $0.01 < \omega^\delta < 5$ are plotted.

4.4 Engineering Application: Fluidized-Bed Process

A top-spray fluidized bed microencapsulation processing unit is frequently used in the food industry to tune the effect of functional ingredients and additives. An important thermo-dynamic response of food producer's interest is the steady-state outlet air temperature T_2 . Dewettinck et al. [49] investigated a physical experiment and several associated computer simulation models for predicting the steady-state thermodynamic operation point of a Glatt GPC-1 fluidized-bed unit. Reese et al. [1] proposed a recursive

Bayesian hierarchical linear model to simultaneously analyze the combined data from both experiment tests and computer simulations. Qian and Wu [19] analyzed the same data using their proposed Bayesian hierarchical Gaussian process model.

Several factors can potentially affect the steady-state thermo-dynamic operating points, including relevant process variables such as the fluidization air velocity (V_f), the air temperature from the pump (T_a), the coating solution flow rate (R_f), the coating solution temperature (T_s), the coating solution dry matter content (M_d), the atomization air pressure (P_a), and some other existing ambient variables such as the room temperature (T_r) and the room humidity (H_r).

Dewettinck et al. collected the physical experiment responses $T_{2, \text{exp}}$ under 28 different process conditions of particular interest using distilled water as the coating solution (i.e., $M_d = 0$), and setting its temperature T_s at 20°C. As a consequence, six factors (H_r , T_r , T_a , R_f , P_a , V_f) with different values are considered in the analyses. Apart from performing physical experiments, Dewettinck et al. also developed several computer simulation models to predict the steady-state outlet air temperature for each process condition, but there exist major differences among those simulation models. Previous works [1, 19, 49] detailed the data about six input variables, and their responses from the experiment tests and computer simulations under the 28 different conditions. Qian and Wu [19] considered the data from only one computer model ($T_{2,2}$) and physical experiment ($T_{2,\text{exp}}$), while we incorporate additional data from a less accurate computer model ($T_{2,1}$) in our work. We will show that incorporating lower-fidelity data from $T_{2,1}$ improves the final prediction.

Eight specific physical experiment runs $T_{2, \text{exp}}$ (i.e., runs 4, 15, 17, 21, 23, 25, 26, and 28 as in Qian and Wu's work) are reserved to form the dataset for model validation. All 28 $T_{2,1}$ and $T_{2,2}$ runs as well as the remaining 20 $T_{2, \text{exp}}$ runs are used in the training of the multi-model fusion. Fig. 6(a), (c) and (e) show the error-bar plots of the resulting predictions of the eight reserved validation points after our proposed nonhierarchical multi-model fusions, against the experimentally observed steady-state outlet air

temperatures. The 95% PI for each point is shown in the figure as well. The predictions of our three approaches are fairly close to the observed values and with very small PIs. Most validation points are almost located on the $y = x$ (green dash) line, except for the fifth observation from the left side deviating a little in each approach.

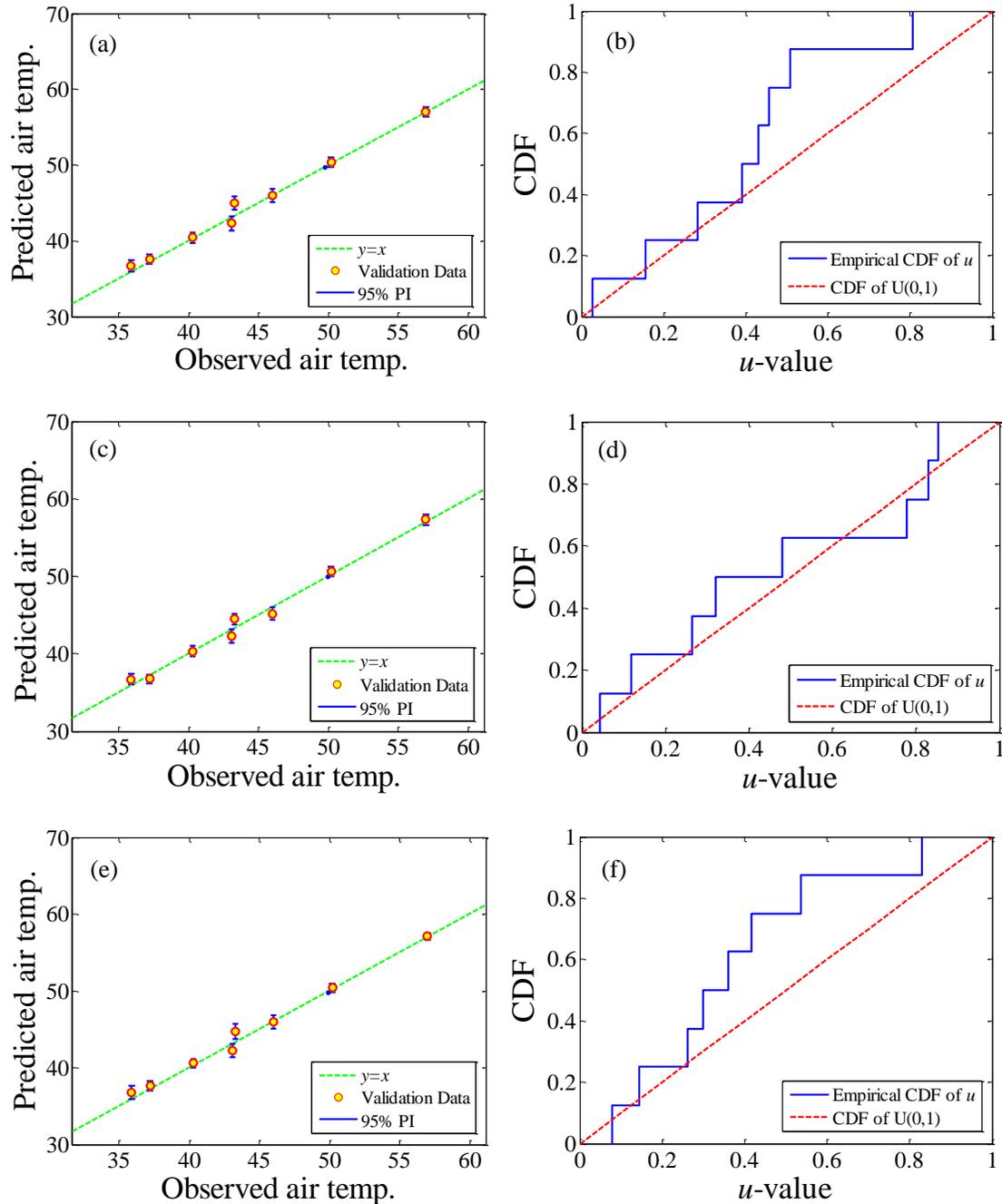


Fig. 6 Observed versus predicted steady-state outlet air temperatures and 95% PIs (left column), as well as their u -pooling metric plots (right column) using WS approach

(upper), PC-DIT approach (middle), and PC-CSC approach (lower).

Table 4: Metrics for engineering application example

Approach	WS	PC-DIT	PC-CSC	Qian and Wu's
RMSE	0.7402	0.6884	0.6925	/
u-pooling	0.1210	0.0706	0.1410	/
SRMSE	0.0177	0.0163	0.0169	0.020

The RMSE and u-pooling metrics for our proposed multi-model fusion approaches are computed in Table 4. The close values of RMSE indicates that the approaches work equally well in the mean prediction of the steady-state outlet air temperature, while the PC-DIT approach excels a lot in terms of the overall prediction performance with much smaller u-pooling metric than the other two approaches. The u-pooling metric plots of our proposed approaches are shown in Fig. 6(b), (d), and (f).

Finally, in order to compare the prediction accuracy of our proposed approaches with that of Qian and Wu's approach [19], the standardized root mean square errors (SRMSEs) for our three approaches are also provided in Table 4 for this application example. It is found that the multi-model fusion can efficiently improve prediction performance because the SRMSEs of all our three proposed approaches are smaller than that of Qian and Wu's approach. This result comes as no surprise, because our proposed multi-model fusion approaches integrate additional data (i.e., simulation data from $T_{2,1}$) for response predictions. Although less accurate, the computer model $T_{2,1}$ still can provide some useful information to guide the prediction in a better manner. This benefit will be more distinct when the less accurate responses come from the input sites where no more accurate data are available. In general, the proposed multi-model fusion approaches that use more data tend to perform better, unless the additional data are totally wrong and heavily mislead the prediction. However, in most circumstances, the modelers can easily distinguish the seriously biased models by prior knowledge and/or preliminary data analyses and then exclude them in the model fusion.

5. CONCLUSIONS

In this work, three different nonhierarchical multi-model fusion approaches are proposed, based on SRP modeling, to integrate the low-fidelity data from multiple alternative or competing simulation models, as well as the high-fidelity data from either experimental or industry-standard simulation observations, for building an accurate and yet computationally efficient predictive model. Each proposed approach imposes different assumptions and structures to capture the relationships between the simulation models and the high fidelity observations. The WS approach achieves final prediction as a weighted sum of simulation models and a single residual discrepancy function; the other two PC approaches assign the model discrepancy associated with each simulation model, PC-DIT assuming independency between the model discrepancy and the true physics, while PC-CSC adopting a fully correlated multi-response SRP structure with a common spatial correlation function for the simulation models and their discrepancies. To illustrate and compare the effectiveness of different proposed approaches, three numerical examples with different model fidelity characteristics are tested, as well as one real engineering application concerning the steady-state thermodynamics of the fluidized-bed process. For validating the accuracy of the multi-model fusion approaches, two model validation metrics are computed, with RMSE assessing the accuracy of the mean prediction and u-pooling measuring the overall model prediction performance considering the uncertainties in both model predictions and experimental observations at multiple validation sites.

The proposed nonhierarchical approaches are more flexible than the existing hierarchical multi-fidelity approaches to handle various kinds of sophisticated scenarios for multi-model fusion, such as clearly different model fidelities, similar model fidelities, and range-dependent model fidelities. Although there are slight differences between the three approaches in the mean prediction or MSE, in general, all of them perform equivalently well with close metric values as demonstrated in the numerical examples.

From the comparative study of our approaches with the existing approach in the real engineering application, the incorporation of more data (even though low-fidelity) in multi-model fusion improves the prediction performance. Note that although we refer to our proposed approaches as “nonhierarchical”, they can also be used in situations with hierarchical levels of model fidelity. However, they would not utilize any knowledge of the fidelity levels when fusing the data from multiple-fidelity models.

One additional observation about the WS approach is that it is sensitive to the prior ranges of hyperparameters for their estimations via the MLE method. It may suffer from inferior prediction in the situation with range-dependent model fidelity because its global-sense weight parameter will induce a relatively large but smooth residual discrepancy. However, this can be mitigated by assigning a smaller prior upper bound for the roughness parameter of discrepancy function.

The proposed nonhierarchical multi-model fusion approaches are widely applicable to many examples and phenomena from structural optimization, fluid analysis, electrical design, financial prediction, and other fields. Nevertheless there are still some avenues for future research that can be envisioned for our proposed approaches. First, Kennedy and O’Hagan’s modular Bayesian method [5] or Higdon et al.’s full Bayesian method [6] can be applied in our approaches to incorporate the calibration of unknown model parameter in the simulation. Second, following the idea from Goldstein and Rougier [50], and Xiu et al. [25], one could extend our approaches to the situation in which the simulation models and/or experiment have different input variables because of the model physics reduction or approximation. Third, one could incorporate additional information (if available), such as the prior knowledge of the fidelities of different models and/or multiple responses from models and experiments (i.e., vectorial responses), to further improve the predictive capability of our proposed approaches.

ACKNOWLEDGEMENTS

The grant support from the National Science Foundation (CMMI-1233403) and the China Scholarship Council is greatly acknowledged. The views expressed in this work are those of authors and do not necessarily reflect the views of the sponsors.

APPENDIX A: β , \mathbf{H} and \mathbf{V}_a for Hyperparameters Estimation

(a) Approach 1: WS

$$\beta = \begin{bmatrix} \beta^{m\{1\}} \\ \vdots \\ \beta^{m\{Q\}} \\ \hline \beta^\delta \end{bmatrix}, \quad \mathbf{H} = \begin{bmatrix} \mathbf{H}^m(\mathbf{X}^{m\{1\}})^T & \cdots & \mathbf{0} & \vdots & \mathbf{0} \\ \vdots & \ddots & \vdots & & \vdots \\ \mathbf{0} & \cdots & \mathbf{H}^m(\mathbf{X}^{m\{Q\}})^T & & \mathbf{0} \\ \hline \rho^{\{1\}}\mathbf{H}^m(\mathbf{X}^e)^T & \cdots & \rho^{\{Q\}}\mathbf{H}^m(\mathbf{X}^e)^T & & \mathbf{H}^\delta(\mathbf{X}^e)^T \end{bmatrix}, \quad (\text{A1})$$

$$\mathbf{V}_a = \begin{bmatrix} e_1^T \Sigma^m e_1 R^m(\mathbf{X}^{m\{1\}}, \mathbf{X}^{m\{1\}}) & \cdots & e_1^T \Sigma^m e_Q R^m(\mathbf{X}^{m\{1\}}, \mathbf{X}^{m\{Q\}}) & \vdots & e_1^T \Sigma^m \rho R^m(\mathbf{X}^{m\{1\}}, \mathbf{X}^e) \\ \vdots & \ddots & \vdots & & \vdots \\ e_Q^T \Sigma^m e_1 R^m(\mathbf{X}^{m\{Q\}}, \mathbf{X}^{m\{1\}}) & \cdots & e_Q^T \Sigma^m e_Q R^m(\mathbf{X}^{m\{Q\}}, \mathbf{X}^{m\{Q\}}) & & e_Q^T \Sigma^m \rho R^m(\mathbf{X}^{m\{Q\}}, \mathbf{X}^e) \\ \hline \rho^T \Sigma^m e_1 R^m(\mathbf{X}^e, \mathbf{X}^{m\{1\}}) & \cdots & \rho^T \Sigma^m e_Q R^m(\mathbf{X}^e, \mathbf{X}^{m\{Q\}}) & & \rho^T \Sigma^m \rho R^m(\mathbf{X}^e, \mathbf{X}^e) \\ & & & & + \sigma_\delta^2 R^\delta(\mathbf{X}^e, \mathbf{X}^e) + \lambda \mathbf{I} \end{bmatrix}. \quad (\text{A2})$$

Note: We are using a compact notation here such that $\mathbf{H}^m(\mathbf{X}^{m\{i\}})^T$ denotes the matrix $[\mathbf{h}^m(\mathbf{x}_1^{m\{i\}}), \mathbf{h}^m(\mathbf{x}_2^{m\{i\}}), \dots, \mathbf{h}^m(\mathbf{x}_{M_i}^{m\{i\}})]^T$ for all $i \in [1, 2, \dots, Q]$, and similarly for other terms as well.

(b) Approach 2(a): PC-DIT

$$\beta = \begin{bmatrix} \beta^{\delta\{1\}} \\ \vdots \\ \beta^{\delta\{Q\}} \\ \hline \beta^t \end{bmatrix}, \quad \mathbf{H} = \begin{bmatrix} -\mathbf{H}^\delta(\mathbf{X}^{m\{1\}})^T & \cdots & \mathbf{0} & \vdots & \mathbf{H}^t(\mathbf{X}^{m\{1\}})^T \\ \vdots & \ddots & \vdots & & \vdots \\ \mathbf{0} & \cdots & -\mathbf{H}^\delta(\mathbf{X}^{m\{Q\}})^T & & \mathbf{H}^t(\mathbf{X}^{m\{Q\}})^T \\ \hline \mathbf{0} & \cdots & \mathbf{0} & & \mathbf{H}^t(\mathbf{X}^e)^T \end{bmatrix}, \quad (\text{A3})$$

$$\mathbf{V}_d = \begin{bmatrix} \Sigma_{1,1}^\delta R^\delta(\mathbf{X}^{m\{1\}}, \mathbf{X}^{m\{1\}}) & \cdots & \Sigma_{1,Q}^\delta R^\delta(\mathbf{X}^{m\{1\}}, \mathbf{X}^{m\{Q\}}) & \vdots & \mathbf{0} \\ \vdots & \ddots & \vdots & \vdots & \vdots \\ \Sigma_{Q,1}^\delta R^\delta(\mathbf{X}^{m\{Q\}}, \mathbf{X}^{m\{1\}}) & \cdots & \Sigma_{Q,Q}^\delta R^\delta(\mathbf{X}^{m\{Q\}}, \mathbf{X}^{m\{Q\}}) & \vdots & \mathbf{0} \\ \hline \mathbf{0} & \cdots & \mathbf{0} & \lambda \mathbf{I} & \mathbf{0} \end{bmatrix} + \sigma_t^2 R^t(\mathbf{X}^{all}, \mathbf{X}^{all}), \quad (\text{A4})$$

$$\mathbf{X}^{all} = \begin{bmatrix} \mathbf{X}^{m\{1\}} \\ \vdots \\ \mathbf{X}^{m\{Q\}} \\ \hline \mathbf{X}^e \end{bmatrix}, \quad R^t(\mathbf{X}^{all}, \mathbf{X}^{all}) = \begin{bmatrix} R^t(\mathbf{X}^{m\{1\}}, \mathbf{X}^{m\{1\}}) & \cdots & R^t(\mathbf{X}^{m\{1\}}, \mathbf{X}^{m\{Q\}}) & \vdots & R^t(\mathbf{X}^{m\{1\}}, \mathbf{X}^e) \\ \vdots & \ddots & \vdots & \vdots & \vdots \\ R^t(\mathbf{X}^{m\{Q\}}, \mathbf{X}^{m\{1\}}) & \cdots & R^t(\mathbf{X}^{m\{Q\}}, \mathbf{X}^{m\{Q\}}) & \vdots & R^t(\mathbf{X}^{m\{Q\}}, \mathbf{X}^e) \\ \hline R^t(\mathbf{X}^e, \mathbf{X}^{m\{1\}}) & \cdots & R^t(\mathbf{X}^e, \mathbf{X}^{m\{Q\}}) & \vdots & R^t(\mathbf{X}^e, \mathbf{X}^e) \end{bmatrix}. \quad (\text{A5})$$

(c) Approach 2(b): PC-CSC

$$\boldsymbol{\beta} = \begin{bmatrix} \boldsymbol{\beta}^{m\{1\}} \\ \vdots \\ \boldsymbol{\beta}^{m\{Q\}} \\ \hline \boldsymbol{\beta}^{\delta\{1\}} \end{bmatrix}, \quad \mathbf{H} = \begin{bmatrix} \mathbf{H}^m(\mathbf{X}^{m\{1\}})^T & \cdots & \mathbf{0} & \vdots & \mathbf{0} \\ \vdots & \ddots & \vdots & \vdots & \vdots \\ \mathbf{0} & \cdots & \mathbf{H}^m(\mathbf{X}^{m\{Q\}})^T & \vdots & \mathbf{0} \\ \hline \mathbf{H}^m(\mathbf{X}^e)^T & \cdots & \mathbf{0} & \mathbf{H}^\delta(\mathbf{X}^e)^T & \mathbf{0} \end{bmatrix}, \quad (\text{A6})$$

$$\mathbf{V}_d = \begin{bmatrix} \mathbf{V}_{1,1} & \mathbf{V}_{2,1}^T \\ \mathbf{V}_{2,1} & \mathbf{V}_{2,2} \end{bmatrix}, \quad \mathbf{V}_{1,1} = \begin{bmatrix} \Sigma_{1,1}^a R^c(\mathbf{X}^{m\{1\}}, \mathbf{X}^{m\{1\}}) & \cdots & \Sigma_{1,Q}^a R^c(\mathbf{X}^{m\{1\}}, \mathbf{X}^{m\{Q\}}) \\ \vdots & \ddots & \vdots \\ \Sigma_{Q,1}^a R^c(\mathbf{X}^{m\{Q\}}, \mathbf{X}^{m\{1\}}) & \cdots & \Sigma_{Q,Q}^a R^c(\mathbf{X}^{m\{Q\}}, \mathbf{X}^{m\{Q\}}) \end{bmatrix}, \quad (\text{A7})$$

$$\mathbf{V}_{2,1} = \left[\left(\Sigma_{1,1}^a + \Sigma_{Q+1,1}^a \right) R^c(\mathbf{X}^e, \mathbf{X}^{m\{1\}}) \quad \cdots \quad \left(\Sigma_{1,Q}^a + \Sigma_{Q+1,Q}^a \right) R^c(\mathbf{X}^e, \mathbf{X}^{m\{Q\}}) \right], \quad (\text{A8})$$

$$\mathbf{V}_{2,2} = \left(\Sigma_{1,1}^a + 2\Sigma_{1,Q+1}^a + \Sigma_{Q+1,Q+1}^a \right) R^c(\mathbf{X}^e, \mathbf{X}^e) + \lambda \mathbf{I}. \quad (\text{A9})$$

APPENDIX B: \mathbf{H}_p , \mathbf{T}_p and \mathbf{V}_p for Experiment Prediction

For notational simplicity we omit the hat operator “ $\hat{\cdot}$ ” for MLEs.

(a) Approach 1: WS

$$\begin{aligned} \mathbf{H}_p &= \begin{bmatrix} \rho^{\{1\}} \mathbf{H}^m(\mathbf{X}^p)^T & \cdots & \rho^{\{Q\}} \mathbf{H}^m(\mathbf{X}^p)^T & \vdots & \mathbf{H}^\delta(\mathbf{X}^p)^T \end{bmatrix} \\ &= \begin{bmatrix} \boldsymbol{\rho}^T \otimes \mathbf{H}^m(\mathbf{X}^p)^T & \vdots & \mathbf{H}^\delta(\mathbf{X}^p)^T \end{bmatrix}, \end{aligned} \quad (\text{B1})$$

$$\mathbf{T}_p = \left[\begin{array}{c|c} \boldsymbol{\rho}^T \boldsymbol{\Sigma}^m e_1 R^m(\mathbf{X}^p, \mathbf{X}^{m(1)}) & \cdots & \boldsymbol{\rho}^T \boldsymbol{\Sigma}^m e_Q R^m(\mathbf{X}^p, \mathbf{X}^{m(Q)}) \\ \hline & & \boldsymbol{\rho}^T \boldsymbol{\Sigma}^m \boldsymbol{\rho} R^m(\mathbf{X}^p, \mathbf{X}^e) \\ & & + \sigma_\delta^2 R^\delta(\mathbf{X}^p, \mathbf{X}^e) \end{array} \right], \quad (\text{B2})$$

$$\mathbf{V}_p = \boldsymbol{\rho}^T \boldsymbol{\Sigma}^m \boldsymbol{\rho} R^m(\mathbf{X}^p, \mathbf{X}^p) + \sigma_\delta^2 R^\delta(\mathbf{X}^p, \mathbf{X}^p) + \lambda \mathbf{I}, \quad (\text{B3})$$

where \otimes denotes the Kronecker product.

(b) Approach 2(a): PC-DIT

$$\mathbf{H}_p = \left[\begin{array}{c|c} \mathbf{0} & \cdots & \mathbf{0} \\ \hline & & \mathbf{H}^t(\mathbf{X}^p)^T \end{array} \right], \quad (\text{B4})$$

$$\mathbf{T}_p = \left[\begin{array}{c|c} \sigma_i^2 R^t(\mathbf{X}^p, \mathbf{X}^{m(1)}) & \cdots & \sigma_i^2 R^t(\mathbf{X}^p, \mathbf{X}^{m(Q)}) \\ \hline & & \sigma_i^2 R^t(\mathbf{X}^p, \mathbf{X}^e) \end{array} \right], \quad (\text{B5})$$

$$\mathbf{V}_p = \sigma_i^2 R^t(\mathbf{X}^p, \mathbf{X}^p) + \lambda \mathbf{I}. \quad (\text{B6})$$

(c) Approach 2(b): PC-CSC

$$\mathbf{H}_p = \left[\begin{array}{c|c} \mathbf{H}^m(\mathbf{X}^p)^T & \cdots & \mathbf{0} \\ \hline & & \mathbf{H}^\delta(\mathbf{X}^p)^T \end{array} \right], \quad (\text{B7})$$

$$\mathbf{T}_p = \left[\begin{array}{c|c} (\boldsymbol{\Sigma}_{1,1}^a + \boldsymbol{\Sigma}_{Q+1,1}^a) R^c(\mathbf{X}^p, \mathbf{X}^{m(1)}) & \cdots & (\boldsymbol{\Sigma}_{1,Q}^a + \boldsymbol{\Sigma}_{Q+1,Q}^a) R^c(\mathbf{X}^p, \mathbf{X}^{m(Q)}) \\ \hline & & (\boldsymbol{\Sigma}_{1,1}^a + 2\boldsymbol{\Sigma}_{1,Q+1}^a + \boldsymbol{\Sigma}_{Q+1,Q+1}^a) R^c(\mathbf{X}^p, \mathbf{X}^e) \end{array} \right], \quad (\text{B8})$$

$$\mathbf{V}_p = (\boldsymbol{\Sigma}_{1,1}^a + 2\boldsymbol{\Sigma}_{1,Q+1}^a + \boldsymbol{\Sigma}_{Q+1,Q+1}^a) R^c(\mathbf{X}^p, \mathbf{X}^p) + \lambda \mathbf{I}. \quad (\text{B9})$$

APPENDIX C: $\tilde{\mathbf{H}}_p$, $\tilde{\mathbf{T}}_p$ and $\tilde{\mathbf{V}}_p$ for Discrepancy Prediction

(a) Approach 1: WS

$$\tilde{\mathbf{H}}_p = \left[\begin{array}{c|c} \mathbf{0} & \cdots & \mathbf{0} \\ \hline & & \mathbf{H}^\delta(\mathbf{X}^p)^T \end{array} \right], \quad (\text{C1})$$

$$\tilde{\mathbf{T}}_p = \left[\begin{array}{c|c} \mathbf{0} & \cdots & \mathbf{0} \\ \hline & & \sigma_\delta^2 R^\delta(\mathbf{X}^p, \mathbf{X}^e) \end{array} \right], \quad (\text{C2})$$

$$\tilde{\mathbf{V}}_p = \sigma_\delta^2 R^\delta(\mathbf{X}^p, \mathbf{X}^p). \quad (\text{C3})$$

(b) Approach 2(a): PC-DIT

$$\tilde{\mathbf{H}}_p = \begin{bmatrix} \mathbf{H}^\delta(\mathbf{X}^p)^T & \cdots & \mathbf{0} & \mathbf{0} \\ \vdots & \ddots & \vdots & \vdots \\ \mathbf{0} & \cdots & \mathbf{H}^\delta(\mathbf{X}^p)^T & \mathbf{0} \end{bmatrix}, \quad (\text{C4})$$

$$\tilde{\mathbf{T}}_p = \begin{bmatrix} -\Sigma_{1,1}^\delta R^\delta(\mathbf{X}^p, \mathbf{X}^{m\{1\}}) & \cdots & -\Sigma_{1,Q}^\delta R^\delta(\mathbf{X}^p, \mathbf{X}^{m\{Q\}}) & \mathbf{0} \\ \vdots & \ddots & \vdots & \vdots \\ -\Sigma_{Q,1}^\delta R^\delta(\mathbf{X}^p, \mathbf{X}^{m\{1\}}) & \cdots & -\Sigma_{Q,Q}^\delta R^\delta(\mathbf{X}^p, \mathbf{X}^{m\{Q\}}) & \mathbf{0} \end{bmatrix}, \quad (\text{C5})$$

$$\tilde{\mathbf{V}}_p = \begin{bmatrix} \Sigma_{1,1}^\delta R^\delta(\mathbf{X}^p, \mathbf{X}^p) & \cdots & \Sigma_{1,Q}^\delta R^\delta(\mathbf{X}^p, \mathbf{X}^p) \\ \vdots & \ddots & \vdots \\ \Sigma_{Q,1}^\delta R^\delta(\mathbf{X}^p, \mathbf{X}^p) & \cdots & \Sigma_{Q,Q}^\delta R^\delta(\mathbf{X}^p, \mathbf{X}^p) \end{bmatrix}. \quad (\text{C6})$$

(c) Approach 2(b): PC-CSC

$$\tilde{\mathbf{H}}_p = \begin{bmatrix} \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{H}^\delta(\mathbf{X}^p)^T \\ \mathbf{H}^m(\mathbf{X}^p)^T & -\mathbf{H}^m(\mathbf{X}^p)^T & \cdots & \mathbf{0} & \mathbf{H}^\delta(\mathbf{X}^p)^T \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \mathbf{H}^m(\mathbf{X}^p)^T & \mathbf{0} & \cdots & -\mathbf{H}^m(\mathbf{X}^p)^T & \mathbf{H}^\delta(\mathbf{X}^p)^T \end{bmatrix}, \quad (\text{C7})$$

$$\tilde{\mathbf{T}}_p = \begin{bmatrix} \Sigma_{Q+1,1}^a R^c(\mathbf{X}^p, \mathbf{X}^{m\{1\}}) & \cdots & \Sigma_{Q+1,Q}^a R^c(\mathbf{X}^p, \mathbf{X}^{m\{Q\}}) & (\Sigma_{Q+1,1}^a + \Sigma_{Q+1,Q+1}^a) R^c(\mathbf{X}^p, \mathbf{X}^e) \\ \vdots & \ddots & \vdots & \vdots \\ \Sigma_{2Q,1}^a R^c(\mathbf{X}^p, \mathbf{X}^{m\{1\}}) & \cdots & \Sigma_{2Q,Q}^a R^c(\mathbf{X}^p, \mathbf{X}^{m\{Q\}}) & (\Sigma_{2Q,1}^a + \Sigma_{2Q,Q+1}^a) R^c(\mathbf{X}^p, \mathbf{X}^e) \end{bmatrix}, \quad (\text{C8})$$

$$\tilde{\mathbf{V}}_p = \begin{bmatrix} \Sigma_{Q+1,Q+1}^a R^c(\mathbf{X}^p, \mathbf{X}^p) & \cdots & \Sigma_{Q+1,2Q}^a R^c(\mathbf{X}^p, \mathbf{X}^p) \\ \vdots & \ddots & \vdots \\ \Sigma_{2Q,Q+1}^a R^c(\mathbf{X}^p, \mathbf{X}^p) & \cdots & \Sigma_{2Q,2Q}^a R^c(\mathbf{X}^p, \mathbf{X}^p) \end{bmatrix}. \quad (\text{C9})$$

APPENDIX D: Mathematical Expressions for Numerical Examples

(1) Numerical Example 1:

$$y'(x) = \sin(x) + 0.2x + (x-5)^2/16 + 0.5, \quad (\text{D1})$$

$$y^e(x) = y'(x) + 0.2\varepsilon, \quad \varepsilon \sim \mathcal{N}(0,1), \quad (\text{D2})$$

$$y^{m(1)}(x) = (x-0.5)(x-4)(x-9)/20 + 2, \quad (\text{D3})$$

$$y^{m(2)}(x) = \sin(x) + 0.2x + 0.5, \quad (\text{D4})$$

(for all responses, $x \in [0, 10]$).

(2) Numerical Example 2:

$$y'(x) = \sin(x), \quad (\text{D5})$$

$$y^e(x) = y'(x) + 0.1\varepsilon, \quad \varepsilon \sim \mathcal{N}(0,1), \quad (\text{D6})$$

$$y^{m(1)}(x) = \sin(x) + 0.1(x-\pi)^2, \quad (\text{D7})$$

$$y^{m(2)}(x) = 1.2\sin(x) + 0.1(x-\pi)^2 - 0.2, \quad (\text{D8})$$

(for all responses, $x \in [0, 2\pi]$).

(3) Numerical Example 3:

$$y'(x) = 2\sin(\pi x/5), \quad (\text{D9})$$

$$y^e(x) = y'(x) + 0.2\varepsilon, \quad \varepsilon \sim \mathcal{N}(0,1), \quad (\text{D10})$$

$$y^{m(1)}(x) = x(x-5)(x-12)/30, \quad (\text{D11})$$

$$y^{m(2)}(x) = (x+2)(x-5)(x-10)/30, \quad (\text{D12})$$

(for all responses, $x \in [0, 10]$).

REFERENCES

1. Reese CS, Wilson AG, Hamada M, Martz HF, Ryan KJ. Integrated analysis of computer and physical experiments. *Technometrics* 2004. **46**(2): 153-164.
2. Pacheco JE, Amon CH, Finger S. Bayesian surrogates applied to conceptual stages of the engineering design process. *Journal of Mechanical Design* 2003. **125**(4): 664-672.
3. El-Beltagy MA, Wright WA. Gaussian processes for model fusion. *Artificial Neural Networks—ICANN 2001*, Vienna, Austria, 2001; 376-383.
4. Osio IG, Amon CH. An engineering design methodology with multistage

- Bayesian surrogates and optimal sampling. *Research in Engineering Design* 1996. **8**(4): 189-206.
5. Kennedy MC, O'Hagan A. Bayesian calibration of computer models. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 2001. **63**(3): 425-464.
 6. Higdon D, Kennedy M, Cavendish JC, Cafeo JA, Ryne RD. Combining field data and computer simulations for calibration and prediction. *SIAM Journal on Scientific Computing* 2004. **26**(2): 448-466.
 7. Bayarri MJ, Berger JO, Paulo R, Sacks J, Cafeo JA, Cavendish J, Lin C-H, Tu J. A framework for validation of computer models. *Technometrics* 2007. **49**(2): 138-154.
 8. Liu F, Bayarri M, Berger J, Paulo R, Sacks J. A Bayesian analysis of the thermal challenge problem. *Computer Methods in Applied Mechanics and Engineering* 2008. **197**(29): 2457-2466.
 9. Chen W, Xiong Y, Tsui K-L, Wang S. A design-driven validation approach using Bayesian prediction models. *Journal of Mechanical Design* 2008. **130**(2): 021101.
 10. Jiang Z, Chen W, Fu Y, Yang R-J. Reliability-Based Design Optimization with Model Bias and Data Uncertainty. *SAE International Journal of Materials and Manufacturing* 2013. **6**(3): 1-15.
 11. Jiang Z, Li W, Apley DW, Chen W. A System Uncertainty Propagation Approach with Model Uncertainty Quantification in Multidisciplinary Design. *ASME 2014 International Design Engineering Technical Conferences & Computers and Information in Engineering Conference*, Buffalo, New York, 2014; 1-13.
 12. Wang S, Chen W, Tsui K-L. Bayesian validation of computer models. *Technometrics* 2009. **51**(4).
 13. Kennedy MC, O'Hagan A. Predicting the output from a complex computer code when fast approximations are available. *Biometrika* 2000. **87**(1): 1-13.
 14. Le Gratiet L. Bayesian analysis of hierarchical multifidelity codes. *SIAM/ASA Journal on Uncertainty Quantification* 2013. **1**(1): 244-269.
 15. Gratiet LL. Recursive co-kriging model for Design of Computer experiments with multiple levels of fidelity with an application to hydrodynamic. *International Journal for Uncertainty Quantification* 2014. **4**(5): 365-386.
 16. Forrester AI, Sobester A, Keane AJ. Multi-fidelity optimization via surrogate modelling. *Proceedings of the royal society A: mathematical, physical and engineering science* 2007. **463**(2088): 3251-3269.
 17. Huang D, Allen T, Notz W, Miller R. Sequential kriging optimization using multiple-fidelity evaluations. *Structural and Multidisciplinary Optimization* 2006. **32**(5): 369-382.
 18. Goh J, Bingham D, Holloway JP, Grosskopf MJ, Kuranz CC, Rutter E. Prediction and Computer Model Calibration Using Outputs From Multifidelity Simulators. *Technometrics* 2013. **55**(4): 501-512.

19. Qian PZ, Wu CJ. Bayesian hierarchical modeling for integrating low-accuracy and high-accuracy experiments. *Technometrics* 2008. **50**(2): 192-204.
20. Qian ZG, Seepersad CC, Joseph VR, Allen JK, Wu CFJ. Building surrogate models based on detailed and approximate simulations. *Journal of Mechanical Design* 2006. **128**(4): 668-677.
21. Ghanem RG, Spanos PD, *Stochastic finite elements: a spectral approach*. Courier Corporation, 2003.
22. Arnst M, Ghanem R, Phipps E, Red-Horse J. Dimension reduction in stochastic modeling of coupled problems. *International Journal for Numerical Methods in Engineering* 2012. **92**(11): 940-968.
23. Eldred MS. Recent advances in non-intrusive polynomial chaos and stochastic collocation methods for uncertainty analysis and design. *The 50th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, Palm Springs, California, 2009; 1-37.
24. Ng LW-T, Eldred M. Multifidelity uncertainty quantification using nonintrusive polynomial chaos and stochastic collocation. *14th AIAA Non-Deterministic Approaches Conference*, Honolulu, Hawaii, April 23-26, 2012; AIAA-2012-1852.
25. Narayan A, Gittelsohn C, Xiu D. A Stochastic Collocation Algorithm with Multifidelity Models. *SIAM Journal on Scientific Computing* 2014. **36**(2): A495-A521.
26. Zhu X, Narayan A, Xiu D. Computational Aspects of Stochastic Collocation with Multifidelity Models. *SIAM/ASA Journal on Uncertainty Quantification* 2014. **2**(1): 444-463.
27. Grimmett G, Stirzaker D, *Probability and random processes*. Oxford university press, 2001.
28. Gallager RG, *Stochastic processes: theory for applications*. Cambridge University Press, 2013.
29. Santner TJ, Williams BJ, Notz W, *The design and analysis of computer experiments*. Springer: New York, 2003.
30. Rasmussen CE, Williams CK, *Gaussian processes for machine learning*. The MIT Press: Massachusetts, 2006.
31. Sacks J, Welch WJ, Mitchell TJ, Wynn HP. Design and analysis of computer experiments. *Statistical Science* 1989. **4**(4): 409-423.
32. Welch WJ, Buck RJ, Sacks J, Wynn HP, Mitchell TJ, Morris MD. Screening, predicting, and computer experiments. *Technometrics* 1992. **34**(1): 15-25.
33. Xiong Y, Chen W, Apley D, Ding X. A non-stationary covariance-based Kriging method for metamodelling in engineering design. *International Journal for Numerical Methods in Engineering* 2007. **71**(6): 733-756.
34. Cressie N, *Statistics for spatial data*. Wiley, 1993.
35. Huang L, Gao Z, Zhang D. Research on multi-fidelity aerodynamic optimization methods. *Chinese Journal of Aeronautics* 2013. **26**(2): 279-286.

36. Xiong Y, Chen W, Tsui K-L, Apley DW. A better understanding of model updating strategies in validating engineering models. *Computer methods in applied mechanics and engineering* 2009. **198**(15): 1327-1337.
37. Tuo R, Wu CJ, Yu D. Surrogate modeling of computer experiments with different mesh densities. *Technometrics* 2013. **56**(3): 372-380.
38. Jin R, Enhancements of metamodeling techniques in engineering design. 2004, Graduate College, University of Illinois at Chicago.
39. Kennedy MC, Anderson CW, Conti S, O'Hagan A. Case studies in Gaussian process modelling of computer codes. *Reliability Engineering & System Safety* 2006. **91**(10): 1301-1309.
40. Conti S, Gosling JP, Oakley J, O'hagan A. Gaussian process emulation of dynamic computer codes. *Biometrika* 2009. **96**(3): 663-676.
41. Conti S, O'Hagan A. Bayesian emulation of complex multi-output and dynamic computer models. *Journal of statistical planning and inference* 2010. **140**(3): 640-651.
42. Arendt PD, Apley DW, Chen W. Quantification of model uncertainty: Calibration, model discrepancy, and identifiability. *Journal of Mechanical Design* 2012. **134**(10): 100908.
43. Arendt PD, Apley DW, Chen W, Lamb D, Gorsich D. Improving identifiability in model calibration using multiple responses. *Journal of Mechanical Design* 2012. **134**(10): 100909.
44. Ferson S, Oberkampf WL, Ginzburg L. Model validation and predictive capability for the thermal challenge problem. *Computer Methods in Applied Mechanics and Engineering* 2008. **197**(29): 2408-2430.
45. Liu Y, Chen W, Arendt P, Huang H-Z. Toward a better understanding of model validation metrics. *Journal of Mechanical Design* 2011. **133**(7): 071005.
46. Li W, Chen W, Jiang Z, Lu Z, Liu Y. New validation metrics for models with multiple correlated responses. *Reliability Engineering & System Safety* 2014. **127**: 1-11.
47. Ferson S, Oberkampf WL. Validation of imprecise probability models. *International Journal of Reliability and Safety* 2009. **3**(1): 3-22.
48. Gramacy RB, Lee HK. Cases for the nugget in modeling computer experiments. *Statistics and Computing* 2012. **22**(3): 713-722.
49. Dewettinck K, Visscher AD, Deroo L, Huyghebaert A. Modeling the steady-state thermodynamic operation point of top-spray fluidized bed processing. *Journal of Food Engineering* 1999. **39**(2): 131-143.
50. Goldstein M, Rougier J. Probabilistic formulations for transferring inferences from mathematical models to physical systems. *SIAM journal on scientific computing* 2004. **26**(2): 467-487.