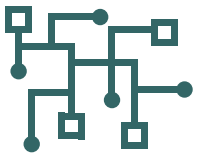


ISE

Industrial and Systems Engineering

PHYSICS-INFORMED KRIGING: A PHYSICS-INFORMED GAUSSIAN PROCESS REGRESSION METHOD FOR DATA-MODEL CONVERGENCE

XIU YANG[1], GUZEL D. TARTAKOVSKY[2], AND ALEXANDRE M. TARTAKOVSKY[3]

[1]Department of Industrial and Systems Engineering, Lehigh University

[2]Hydrology Group, Pacific Northwest National Laboratory

[3]Advanced Computing, Mathematics and Data Division, Pacific Northwest National Laboratory

ISE Technical Report 19T-026

LEHIGH UNIVERSITY

PHYSICS-INFORMED KRIGING: A PHYSICS-INFORMED GAUSSIAN PROCESS REGRESSION METHOD FOR DATA-MODEL CONVERGENCE*

XIU YANG†, GUZEL D. TARTAKOVSKY ‡, AND ALEXANDRE M. TARTAKOVSKY §

Abstract. In this work, we propose a new Gaussian process regression (GPR) method: physics-informed Kriging (PhIK). In the standard data-driven Kriging, the unknown function of interest is usually treated as a Gaussian process with assumed stationary covariance with hyperparameters estimated from data. In PhIK, we compute the mean and covariance function from realizations of available stochastic models, e.g., from realizations of governing stochastic partial differential equations solutions. Such constructed Gaussian process generally is non-stationary and does not assume a specific form of the covariance function. Our approach avoids the costly optimization step in data-driven GPR methods to identify the hyperparameters. More importantly, we prove that the physical constraints in the form of a deterministic linear operator are guaranteed in the resulting prediction. We also provide an error estimate in preserving the physical constraints when errors are included in the stochastic model realizations. To reduce the computational cost of obtaining stochastic model realizations, we propose a multilevel Monte Carlo estimate of the mean and covariance functions. Further, we present an active learning algorithm that guides the selection of additional observation locations. The efficiency and accuracy of PhIK are demonstrated for reconstructing a partially known modified Branin function and learning a conservative tracer distribution from sparse concentration measurements.

Key words. physics-informed, Gaussian process regression, active learning, error bound.

AMS subject classifications. 65C60, 42C05, 41A10

1. Introduction.

Gaussian process regression (GPR), also known as *Kriging* in geostatistics, is a widely used method in applied mathematics, statistics and machine learning for constructing surrogate models, interpolation, classification, supervised learning, and active learning [16, 41, 43]. GPR constructs a statistical model of a partially observed function (of time and/or space) assuming this function is a realization of a Gaussian process (GP). GP is uniquely described by its mean and covariance function. In the standard (here referred to as *data-driven*) GP, prescribed forms of mean and covariance functions are assumed, and the hyperparameters (e.g., variance and correlation length) are computed from data via negative log-marginal likelihood function minimization. There are several variants of GPR, including simple, ordinary, and universal Kriging [25]. GPR is also closely related to kernel machines in machine learning, but it includes more information as it provides the uncertainty estimate [48].

In the ordinary Kriging, the data are modeled as a GP with constant mean and a prescribed form of the stationary covariance function (also known as *kernel*). The stationarity assumption reduces the number of hyperparameters and model complexity.

* **Funding:** This work was supported by the U.S. Department of Energy (DOE), Office of Science, Office of Advanced Scientific Computing Research (ASCR) as part of the Multifaceted Mathematics for Complex Systems and Uncertainty Quantification in Advection-Diffusion-Reaction Systems projects. A portion of the research described in this paper was conducted under the Laboratory Directed Research and Development Program at Pacific Northwest National Laboratory (PNNL). PNNL is operated by Battelle for the DOE under Contract DE-AC05-76RL01830.

†Department of Industrial and Systems Engineering, Lehigh University, Bethlehem, PA 18015 (Corresponding author: xiy518@lehigh.edu)

‡Hydrology Group, Pacific Northwest National Laboratory, Richland, WA 99352 (guzel.tartakovsky@pnnl.gov)

§Advanced Computing, Mathematics and Data Division, Pacific Northwest National Laboratory, Richland, WA 99352 (alexandre.tartakovsky@pnnl.gov)

However, many fields are not stationary. Furthermore, even if the process is stationary, there are many choices of the covariance functions with different smoothness properties. In the universal Kriging, the assumption of constant mean is relaxed by modeling it as a polynomial [2], which increases the number of unknown parameters and may lead to non-convex optimization problems. Usually, there are not enough data to get an accurate estimate of the true statistics.

In this work, we incorporate physical knowledge in Kriging by computing mean and covariance function from a partially know physics model. Therefore, we call this method the *physics-informed Kriging*, or *PhIK*. We assume that in addition to data (e.g., hydraulic head), partial physical knowledge is available in the form of partial differential differential equations (e.g., the Darcy law governing the hydraulic head) with unknown boundary, initial condition, and/or space-dependent coefficient (e.g., the hydraulic conductivity). The standard treatment of partial differential equations (PDEs) with unknown parameters is to model the unknown parameters as random variables and solve the resulting stochastic partial differential equations (SPDEs). In general, the covariance of the state variable of SPDEs defined on a bounded domain with non-periodic boundary conditions is non-stationary [44, 23]. Therefore, modeling the measurements of states of such equations with a stationary covariance should lead to significant errors. Some progress has been made to incorporate physical knowledge in kernels, for example, [42, 38] computed kernels for linear and weakly nonlinear (allowing accurate linearization) ordinary and partial differential equations by substituting a GPR approximation of the state variables in a governing equation and obtaining a system of equations for the kernel hyperparameters. For complex linear systems, computing kernel in such a way can become prohibitively expensive, while for strongly nonlinear systems, it may not be possible at all.

Here we propose to compute the mean and covariance function for modeling the state measurements by solving the governing SPDEs. Computational tools, including commercial and open source packages, have achieved a significant degree of maturity for many science applications (e.g., climate modeling, hydrology, aerospace engineering, electrical engineering, etc.), and they can be run in parallel in the "Mote Carlo (MC) mode" to compute mean and covariances to model scientific data. This could be achieved by treating unknown parameters as random parameters or random fields, which is a common approach in uncertainty quantification (UQ) and sensitivity analysis [29, 50, 22, 52, 61, 11]. Here, we propose to use MC or other sampling techniques for constructing a GP model (i.e., to estimate the mean and covariance function) as a way to integrate physical knowledge in GPR. In addition to making GPR prediction more accurate in terms of preserving some physical constraints, this approach removes the need for assuming a specific form of the kernel and solving a costly optimization problem for its hyperparameters. A similar idea is adopted in the ensemble Kalman filter (EnKF) [15] for data assimilation in time-dependent problems, where the covariance matrix of the PDF of the state vector is represented by an ensemble of the model outputs.

The cost of estimating mean and covariance depends on the size and complexity of the physical model. We propose to reduce this cost by using multilevel Monte Carlo (MLMC) [19]. Traditionally, MLMC has been used to approximate the mean and one-point moments by combining a relatively few high-resolution simulations with a (larger) number of coarse resolution simulations to compute moments with the desired accuracy. We extend MLMC for approximating covariance function, a two-point second moment. Then, we provide error estimates for PhIK and MLMC-based PhIK describing how well physics constraints are preserved. Finally, we apply PhIK

for active learning (i.e., choosing additional measurement locations) using the mean squared error (MSE) of the PhIK prediction.

This work is organized as follows: Section 2 introduces PhIK, the MLMC method for covariance estimates and the error estimates for PhIK and active learning. Section 3 provides two numerical examples to demonstrate the efficiency of the proposed method. Conclusions are presented in Section 4.

2. Methodology. This section begins by reviewing the general GPR framework [16] and the Kriging method based on the assumption of stationary GP [1]. Next, we introduce the PhIK and MLMC-based PhIK. Finally, we present an active learning algorithm based on PhIK.

2.1. GPR framework. We denote the observation locations as $\boldsymbol{X} = \{\boldsymbol{x}^{(i)}\}_{i=1}^{N}$ ($\boldsymbol{x}^{(i)}$ are d-dimensional vectors in $D \subseteq \mathbb{R}^d$) and the observed state values at these locations as $\boldsymbol{y} = (y^{(1)}, y^{(2)}, \cdots, y^{(N)})^{\mathsf{T}}$ ($y^{(i)} \in \mathbb{R}$). For simplicity, we assume that $y^{(i)}$ are scalars. We aim to predict y at any new location $\boldsymbol{x}^* \in D$. The GPR method assumes that the observation vector $\boldsymbol{y}$ is a realization of the following N-dimensional random vector that satisfies multivariate Gaussian distribution:

$$\boldsymbol{Y} = \left(Y(\boldsymbol{x}^{(1)}), Y(\boldsymbol{x}^{(2)}), \cdots, Y(\boldsymbol{x}^{(N)})\right)^{\mathsf{T}},$$

where $Y(\boldsymbol{x}^{(i)})$ is the concise notation of $Y(\boldsymbol{x}^{(i)}; \omega)$, and $Y(\boldsymbol{x}^{(i)}, \omega)$ is a Gaussian random variable defined on a probability space $(\Omega, \mathcal{F}, P)$ with $\omega \in \Omega$. Of note, $\boldsymbol{x}^{(i)}$ can be considered as parameters for the GP $Y : D \times \Omega \to \mathbb{R}$, such that $Y(\boldsymbol{x}^{(i)}) : \Omega \to \mathbb{R}$ is a Gaussian random variable for any $\boldsymbol{x}^{(i)}$ in the set D. Usually, $Y(\boldsymbol{x})$ is denoted as

$$Y(\boldsymbol{x}) \sim \mathcal{GP}\left(\mu(\boldsymbol{x}), k(\boldsymbol{x}, \boldsymbol{x}')\right), \tag{2.1}$$

where $\mu : D \to \mathbb{R}$ and $k : D \times D \to \mathbb{R}$ are the mean and covariance functions:

$$\mu(\boldsymbol{x}) = \mathbb{E}\left\{Y(\boldsymbol{x})\right\} \tag{2.2}$$

$$k(\boldsymbol{x}, \boldsymbol{x}') = \mathrm{Cov}\left\{Y(\boldsymbol{x}), Y(\boldsymbol{x}')\right\} = \mathbb{E}\left\{(Y(\boldsymbol{x}) - \mu(\boldsymbol{x}))(Y(\boldsymbol{x}') - \mu(\boldsymbol{x}'))\right\}. \tag{2.3}$$

The variance of $Y(\boldsymbol{x})$ is $k(\boldsymbol{x}, \boldsymbol{x})$, and its standard deviation is $\sigma(\boldsymbol{x}) = \sqrt{k(\boldsymbol{x}, \boldsymbol{x})}$. The covariance matrix of random vector $\boldsymbol{Y}$ is defined as

$$\mathbf{C} = \begin{pmatrix} k(\boldsymbol{x}^{(1)}, \boldsymbol{x}^{(1)}) & \cdots & k(\boldsymbol{x}^{(1)}, \boldsymbol{x}^{(N)}) \\ \vdots & \ddots & \vdots \\ k(\boldsymbol{x}^{(N)}, \boldsymbol{x}^{(1)}) & \cdots & k(\boldsymbol{x}^{(N)}, \boldsymbol{x}^{(N)}) \end{pmatrix}. \tag{2.4}$$

The prediction at location $\boldsymbol{x}^*$ is given as

$$\hat{y}(\boldsymbol{x}^*) = \mu(\boldsymbol{x}^*) + \boldsymbol{c}^{\mathsf{T}}\mathbf{C}^{-1}(\boldsymbol{y} - \boldsymbol{\mu}), \tag{2.5}$$

where $\boldsymbol{\mu} = (\mu(\boldsymbol{x}^{(1)}), \cdots, \mu(\boldsymbol{x}^{(N)}))^{\mathsf{T}}$, and $\boldsymbol{c}$ is a vector of covariance between the observed data and the prediction:

$$\boldsymbol{c} = \boldsymbol{c}(\boldsymbol{x}^*) = \left(k(\boldsymbol{x}^{(1)}, \boldsymbol{x}^*), k(\boldsymbol{x}^{(2)}, \boldsymbol{x}^*), \cdots, k(\boldsymbol{x}^{(N)}, \boldsymbol{x}^*)\right)^{\mathsf{T}}. \tag{2.6}$$

The MSE of this prediction is

$$\hat{s}^2(\boldsymbol{x}^*) = \sigma^2(\boldsymbol{x}^*) - \boldsymbol{c}^{\mathsf{T}}\mathbf{C}^{-1}\boldsymbol{c}. \tag{2.7}$$

Here, MSE is defined as $\hat{s}^2(\boldsymbol{x}^*) = \mathbb{E}\left\{(\hat{y}(\boldsymbol{x}^*) - Y(\boldsymbol{x}^*))^2\right\}$, and $\hat{s}(\boldsymbol{x}^*)$ is the root mean squared error (RMSE). Of note, Eq. (2.7) disregards a small term presenting the uncertainty in the estimated mean (see Eq. (3.1) in [16]). Here, prediction and MSE can be derived from the maximum likelihood estimate (MLE) method [16]. There are also other routes to obtain $\hat{y}$ and $\hat{s}$. For example, the Bayesian framework requires maximizing a log-marginal likelihood, and the result is a posterior distribution $y(\boldsymbol{x}^*) \sim \mathcal{N}(\hat{y}(\boldsymbol{x}^*), \hat{s}^2(\boldsymbol{x}^*))$ (see e.g., [39]). Moreover, to account for the observation noise, one can assume that the noise is independent and identically distributed (i.i.d.) Gaussian random variables with zero mean and variance δ^2, and replace $\mathbf{C}$ with $\mathbf{C} + \delta^2\mathbf{I}$. In this study, we assume that y can be described by a physical model (e.g., a system of PDEs), and noiseless measurements of y are available. We use the physical model realizations to estimate $k(\boldsymbol{x}, \boldsymbol{x}')$ and $\mathbf{C}$. It is straightforward to extend our method to noisy observation cases. Moreover, we assume that $\mathbf{C}$ is invertible. If computed $\mathbf{C}$ is not invertible, following the common GPR approach, one can always add a small regularization term $\alpha\mathbf{I}$ (α is a small positive real number) to $\mathbf{C}$ such that it becomes full rank. Adding the regularization term is equivalent to assuming there is a measurement noise.

2.2. Stationary GPR. In the widely used ordinary Kriging method, a stationary GP is assumed. In this case, μ is set as a constant $\mu(\boldsymbol{x}) \equiv \mu$. Then, the mean of $\boldsymbol{Y}$ is a constant vector $\mathbf{1}\mu$, where $\mathbf{1}$ is an $N \times 1$ column vector of ones. Next, it is assumed that $k(\boldsymbol{x}, \boldsymbol{x}') = k(\boldsymbol{\tau})$, where $\boldsymbol{\tau} = \boldsymbol{x} - \boldsymbol{x}'$, and $\sigma^2(\boldsymbol{x}) = k(\boldsymbol{x}, \boldsymbol{x}) = k(\mathbf{0}) = \sigma^2$ is a constant. To satisfy these conditions, l_i ($i = 1, \cdots, d$), the correlation length of y in the i direction, also must also be a constant. Popular forms of kernels include polynomial, exponential, Gaussian, and Matérn functions. For example, the Gaussian kernel can be written as $k(\boldsymbol{\tau}) = \sigma^2 \exp\Big(-\frac{1}{2}\|\boldsymbol{x} - \boldsymbol{x}'\|_w^2\Big)$, where the weighted norm is defined as $\|\boldsymbol{x} - \boldsymbol{x}'\|_w^2 = \sum_{i=1}^{d}\Big(\frac{x_i - x_i'}{l_i}\Big)^2$.

Given a stationary covariance function, the covariance matrix $\mathbf{C}$ of $\boldsymbol{Y}$ can be written as $\mathbf{C} = \sigma^2\boldsymbol{\Psi}$, where $\psi_{ij} = k(\boldsymbol{x}^{(i)}, \boldsymbol{x}^{(j)})/\sigma^2$. In the MLE framework, the estimators of μ and σ^2, denoted as $\hat{\mu}$ and $\hat{\sigma}^2$, are

$$\hat{\mu} = \frac{\mathbf{1}^\mathsf{T}\boldsymbol{\Psi}^{-1}\boldsymbol{y}}{\mathbf{1}^\mathsf{T}\boldsymbol{\Psi}^{-1}\mathbf{1}}, \qquad \hat{\sigma}^2 = \frac{(\boldsymbol{y} - \mathbf{1}\mu)^\mathsf{T}\boldsymbol{\Psi}^{-1}(\boldsymbol{y} - \mathbf{1}\mu)}{N}. \tag{2.8}$$

The hyperparameters l_i are estimated by maximizing the concentrated ln-likelihood function: $L_c = -\frac{n}{2}\ln(\hat{\sigma}^2) - \frac{1}{2}\ln|\boldsymbol{\Psi}|$. The prediction of y at location $\boldsymbol{x}^*$ is

$$\hat{y}(\boldsymbol{x}^*) = \hat{\mu} + \boldsymbol{\psi}^\mathsf{T}\boldsymbol{\Psi}^{-1}(\boldsymbol{y} - \mathbf{1}\hat{\mu}), \tag{2.9}$$

where $\boldsymbol{\psi}$ is a vector of correlations between the observed data and the prediction,

$$\boldsymbol{\psi} = \boldsymbol{\psi}(\boldsymbol{x}^*) = \frac{1}{\sigma^2}\left(k(\boldsymbol{x}^{(1)} - \boldsymbol{x}^*), k(\boldsymbol{x}^{(2)} - \boldsymbol{x}^*), \cdots, k(\boldsymbol{x}^{(N)} - \boldsymbol{x}^*)\right)^\mathsf{T},$$

and MSE of the prediction is

$$\hat{s}^2(\boldsymbol{x}^*) = \hat{\sigma}^2\left(1 - \psi^\mathsf{T}\boldsymbol{\Psi}^{-1}\boldsymbol{\psi}\right). \tag{2.10}$$

A more general approach is to assume a non-stationary covariance function, which is done by modifying a stationary covariance function that potentially increases the

number of hyperparameters [32, 35, 7]. However, these methods still need to assume a specific form of the correlation functions according to experience. The key computational challenge in the data-driven GPR is the optimization step of maximizing the (marginal) likelihood. In many practical cases, this is a non-convex optimization problem, and the condition number of $\mathbf{C}$ or $\mathbf{\Psi}$ can be quite large. A more fundamental challenge in the data-driven GPR is that it does not explicitly account for physical constraints and requires a large amount of data to accurately model the physics. The PhIK introduced in the next section aims to address both of these challenges.

2.3. PhIK. PhIK takes advantage of the existing domain knowledge in the form of realizations of a stochastic model of the observed system. As such, there is no need to assume a specific form of the correlation functions and solve an optimization problem for the hyperparameters. This idea is motivated by many physical and engineered problems, where approximate numerical or analytical physics-based models are available. These models typically include random parameters or random processes/fields to reflect the lack of understanding (of physical laws) or knowledge (of the coefficients, parameters, etc.) of the real system. Then, MC simulations are conducted to generate an ensemble of state variables, from which the statistics of these state variables, e.g., mean and standard deviation, are estimated. This ensemble can be considered as a collection of (approximate) realizations of the random field Y that we want to identify. Therefore, we can estimate the mean of random field Y and covariance matrix of random vector $\boldsymbol{Y}$ from the MC simulations instead of inferring them from the observations.

Specifically, assume that we have M realizations of $Y(\boldsymbol{x})$ ($\boldsymbol{x} \in D$) denoted as $\{Y^m(\boldsymbol{x})\}_{m=1}^M$. The mean of Y can be estimated as

$$\mu(\boldsymbol{x}) \approx \mu_{_{MC}}(\boldsymbol{x}) = \frac{1}{M}\sum_{m=1}^{M} Y^m(\boldsymbol{x}). \tag{2.11}$$

Similarly, the covariance function is approximated as

$$k(\boldsymbol{x},\boldsymbol{x}') \approx k_{_{MC}}(\boldsymbol{x},\boldsymbol{x}') = \frac{1}{M-1}\sum_{m=1}^{M} \left(Y^m(\boldsymbol{x}) - \mu_{_{MC}}(\boldsymbol{x})\right)\left(Y^m(\boldsymbol{x}') - \mu_{_{MC}}(\boldsymbol{x}')\right). \tag{2.12}$$

Thus, the covariance matrix of $\boldsymbol{Y}$ can be estimated as

$$\mathbf{C} \approx \mathbf{C}_{_{MC}} = \frac{1}{M-1}\sum_{m=1}^{M} \left(\boldsymbol{Y}^m - \boldsymbol{\mu}_{_{MC}}\right)\left(\boldsymbol{Y}^m - \boldsymbol{\mu}_{_{MC}}\right)^{\mathsf{T}}, \tag{2.13}$$

where $\boldsymbol{Y}^m = (Y^m(\boldsymbol{x}^{(1)}), \cdots, Y^m(\boldsymbol{x}^{(N)}))^{\mathsf{T}}$, $\boldsymbol{\mu}_{_{MC}} = (\mu_{_{MC}}(\boldsymbol{x}^{(1)}), \cdots, \mu_{_{MC}}(\boldsymbol{x}^{(N)}))^{\mathsf{T}}$. In Eqs. (2.11) and (2.13), we approximate μ and $\mathbf{C}$ using the ensemble instead of MLE as in the data-driven GPR. When $\mathbf{C}_{_{MC}}$ is invertible, the prediction at location $\boldsymbol{x}^*$, is

$$\hat{y}(\boldsymbol{x}^*) = \mu_{_{MC}}(\boldsymbol{x}^*) + \boldsymbol{c}_{_{MC}}^{\mathsf{T}}\mathbf{C}_{_{MC}}^{-1}(\boldsymbol{y} - \boldsymbol{\mu}_{_{MC}}), \tag{2.14}$$

where $\boldsymbol{c}_{_{MC}} = (k_{_{MC}}(\boldsymbol{x}^{(1)}, \boldsymbol{x}^*), \cdots, k_{_{MC}}(\boldsymbol{x}^{(N)}, \boldsymbol{x}^*))$. The MSE of this prediction is

$$\hat{s}^2(\boldsymbol{x}^*) = \hat{\sigma}_{_{MC}}^2(\boldsymbol{x}^*) - \boldsymbol{c}_{_{MC}}^{\mathsf{T}}\mathbf{C}_{_{MC}}^{-1}\boldsymbol{c}_{_{MC}}, \tag{2.15}$$

where $\hat{\sigma}_{_{MC}}^2(\boldsymbol{x}^*) = k_{_{MC}}(\boldsymbol{x}^*, \boldsymbol{x}^*)$ is the variance of data set $\{Y^m(\boldsymbol{x}^*)\}_{m=1}^M$.

PhIK has several advantages:

- It does not need to assume stationarity of the GP.
- It does not need to assume a specific form of the covariance relation. Thus, the form of the resulting GP is more flexible.
- It does not need to solve the optimization problem to identify hyperparameters, which can be a challenging problem often suffering from the ill-conditioned covariance matrix.
- It incorporates physical constraints via the mean and covariance function.

Next, we present a theorem that details how well PhIK prediction preserves linear physical constraints.

THEOREM 2.1. *Assume that a stochastic model $u(\boldsymbol{x};\omega)$ defined on $\mathbb{R}^d \times \Omega$ satisfies $\|\mathcal{L}u(\boldsymbol{x};\omega) - g(\boldsymbol{x};\omega)\| \leq \epsilon$ for any $\omega \in \Omega$, where $\mathcal{L}$ is a deterministic bounded linear operator, $g(\boldsymbol{x};\omega)$ is a well-defined function on $\mathbb{R}^d \times \Omega$, and $\|\cdot\|$ is a specific norm of a function defined on $\mathbb{R}^d$. $\{Y^m(\boldsymbol{x})\}_{m=1}^M$ are a finite number of realizations of $u(\boldsymbol{x};\omega)$, i.e., $Y^m(\boldsymbol{x}) = u(\boldsymbol{x};\omega^m)$. Then, the prediction $\hat{y}(\boldsymbol{x})$ from PhIK satisfies*

$$\begin{aligned}\|\mathcal{L}\hat{y}(\boldsymbol{x}) - \overline{g(\boldsymbol{x})}\| \leq \epsilon + \left[2\epsilon\sqrt{\frac{M}{M-1}} + \sigma\left(g(\boldsymbol{x};\omega^m)\right)\right] \cdot \\ \left\|\mathbf{C}_{MC}^{-1}(\boldsymbol{y} - \boldsymbol{\mu}_{MC})\right\|_\infty \sum_{i=1}^N \sigma(Y^m(\boldsymbol{x}^{(i)})),\end{aligned} \tag{2.16}$$

where $\sigma(Y^m(\boldsymbol{x}^{(i)}))$ is the standard deviation of data set $\{Y^m(\boldsymbol{x}^{(i)})\}_{m=1}^M$ for each fixed $\boldsymbol{x}^{(i)}$, $\overline{g(\boldsymbol{x})} = \frac{1}{M}\sum_{m=1}^M g(\boldsymbol{x};\omega^m)$, and $\sigma(g(\boldsymbol{x};\omega^m)) = \left(\frac{1}{M-1}\sum_{m=1}^M \|g(\boldsymbol{x};\omega^m) - \overline{g(\boldsymbol{x})}\|^2\right)^{\frac{1}{2}}$.

We present the proof of these two theorems in Appendix A.

This theorem holds for various norms, e.g., L_2 norm, L_∞ norm, and H^1 norm. In practice, the realizations $Y^m(\boldsymbol{x})$ are obtained by numerical simulations and are subject to numerical errors, model errors, etc. Thus, the theorem includes ϵ in the upper bound. It also indicates that the standard deviation of ensemble member Y^m at all observation locations $\boldsymbol{x}^{(i)}$ affects the upper bound of $\|\mathcal{L}(\hat{y}(\boldsymbol{x})) - \overline{g(\boldsymbol{x})}\|$. If the variance of $Y^m(\boldsymbol{x}^{(i)})$ is small at every $\boldsymbol{x}^{(i)}$, e.g., when the physical model is less uncertain, the resulting prediction $\hat{y}(\boldsymbol{x})$ will not violate the linear constraint much, i.e., $\|\mathcal{L}\hat{y}(\boldsymbol{x}) - \overline{g(\boldsymbol{x})}\|$ is small. Moreover, if $g(\boldsymbol{x};\omega)$ is a deterministic function $g(\boldsymbol{x})$, then $\sigma(g(\boldsymbol{x};\omega^m)) = 0$ in the upper bound (see Eq. (2.16)), and we have the following corollary: Another important factor for the error bound is $\max_i |\tilde{a}_i|$, i.e., $\|\mathbf{C}_{MC}^{-1}(\boldsymbol{y} - \boldsymbol{\mu}_{MC})\|_\infty$. The following corollary exploits the relation between this term and $\mathbf{C}_{MC}$ structure.

COROLLARY 2.1. *Given the conditions in Theorem 2.1, we have*

$$\begin{aligned}\|\mathcal{L}\hat{y}(\boldsymbol{x}) - \overline{g(\boldsymbol{x})}\| \leq \epsilon + \left[2\epsilon\sqrt{\frac{M}{M-1}} + \sigma\left(g(\boldsymbol{x};\omega^m)\right)\right] \cdot \\ \left\|\mathbf{C}_{MC}^{-1}\right\|_2 \|\boldsymbol{y} - \boldsymbol{\mu}_{MC}\|_2 \sum_{i=1}^N \sigma(Y^m(\boldsymbol{x}^{(i)})).\end{aligned}$$

Proof.

$$\|\mathbf{C}_{MC}^{-1}(\boldsymbol{y}-\boldsymbol{\mu}_{MC})\|_\infty \le \|\mathbf{C}_{MC}^{-1}(\boldsymbol{y}-\boldsymbol{\mu}_{MC})\|_2 \le \|\mathbf{C}_{MC}^{-1}\|_2\|\boldsymbol{y}-\boldsymbol{\mu}_{MC}\|_2. \qquad \square$$

This corollary indicates that the upper bound is affected by the difference between the physical model output and the observation, i.e., $\|\boldsymbol{y}-\boldsymbol{\mu}_{MC}\|_2$, and the reciprocal of the smallest eigenvalue of $\mathbf{C}_{MC}$, i.e., $\|\mathbf{C}_{MC}^{-1}\|_2$. The former depends on the physical model's accuracy, and the latter is affected by the model and parametric uncertainty, the GP model properties, and/or observation locations. For example, if the correlation length is large and the observations cluster, $\|\mathbf{C}_{MC}^{-1}\|_2$ can be very large.

In addition, the following corollary describes a special case.

COROLLARY 2.2. *In Theorem 2.1, if g is a deterministic function, i.e., $g(\boldsymbol{x};\omega) = g(\boldsymbol{x})$, and $\mathcal{L}u(\boldsymbol{x};\omega) = g(\boldsymbol{x})$ for any $\omega \in \Omega$, then $\mathcal{L}\hat{y}(\boldsymbol{x}) = g(\boldsymbol{x})$.*

Proof. Because $\mathcal{L}Y^m(\boldsymbol{x}) = g(\boldsymbol{x})$ and $\mathcal{L}\mu_{MC}(\boldsymbol{x}) = \overline{g(\boldsymbol{x})} = g(\boldsymbol{x})$, we have

$$\begin{aligned}\mathcal{L}k_{MC}(\boldsymbol{x},\boldsymbol{x}^{(i)}) &= \mathcal{L}\Big[\frac{1}{M-1}\sum_{m=1}^{M}\Big(Y^m(\boldsymbol{x})-\mu_{MC}(\boldsymbol{x})\Big)\Big(Y^m(\boldsymbol{x}^{(i)})-\mu_{MC}(\boldsymbol{x}^{(i)})\Big)\Big]\\ &= \frac{1}{M-1}\sum_{m=1}^{M}\mathcal{L}\Big(Y^m(\boldsymbol{x})-\mu_{MC}(\boldsymbol{x})\Big)\Big(Y^m(\boldsymbol{x}^{(i)})-\mu_{MC}(\boldsymbol{x}^{(i)})\Big) = 0.\end{aligned}$$

Therefore,

$$\mathcal{L}\hat{y}(\boldsymbol{x}) = \mathcal{L}\Big(\mu_{MC}(\boldsymbol{x}) + \sum_{i=1}^{N}\tilde{a}_i k_{MC}(\boldsymbol{x},\boldsymbol{x}^{(i)})\Big) = \mathcal{L}\mu_{MC}(\boldsymbol{x}) = g(\boldsymbol{x}). \qquad \square$$

For example, if $u(\boldsymbol{x};\omega)$ satisfies the Dirichlet boundary condition $u(\boldsymbol{x};\omega) = g(\boldsymbol{x})$, $\boldsymbol{x} \in \partial D_D$ for any $\omega \in \Omega$, then $\hat{y}(\boldsymbol{x}) = g(\boldsymbol{x}), \boldsymbol{x} \in \partial D_D$. Similarly, if $u(\boldsymbol{x};\omega)$ satisfies the Neumann boundary condition, $\partial u(\boldsymbol{x};\omega)/\partial \boldsymbol{n} = 0, \boldsymbol{x} \in \partial D_N$, then $\partial \hat{y}(\boldsymbol{x};\omega)/\partial \boldsymbol{n} = 0, \boldsymbol{x} \in \partial D_N$. Another example is $\mathcal{L}u = \nabla \cdot u$. Then if u satisfies $\nabla \cdot u(\boldsymbol{x};\omega) = 0$ for any $\omega \in \Omega$, $\hat{y}(\boldsymbol{x})$ is also a divergence-free field. In general cases, i.e., g is a random function and $\epsilon \neq 0$, the upper bound in Theorem 2.1 describes how well the physical constraint is preserved.

In this work, we choose to use the MC method to compute the mean and covariance because of its robustness. Empricially, the choice of M is simular to the standard MC method, e.g., in our numerical examples, the order of magnitude of M is $O(100)$. Also, as detailed in the aforementioned theorems using GP statistics estimated by MC method, PhIK predictions preserve physical constraints in the form of deterministic linear operators. However, when the system's number of degrees of freedom is large (e.g., a high-resolution 3D model), the cost of estimating these statistics can be very large, which is the main drawback of the PhIK method. To solve this problem, other sampling methods, including quasi-Monte Carlo [31], probabilistic collocation [51], Analysis Of Variance (ANOVA) [54], and compressive sensing [55], as well as mode reduction methods, e.g., the moment equation method [45], can be used for estimating state statistics. Depending on the applications, these methods could be significantly more efficient than MC. It is not difficult to show that conclusions similar to Theorem 2.1 hold if $\mu(\boldsymbol{x})$ and $k(\boldsymbol{x},\boldsymbol{x}')$ are approximated using a *linear combination* of realizations $\{Y^m(\boldsymbol{x})\}_{m=1}^M$, where these realizations are based on a different sampling strategy. However, extending these theorems to mode reduction methods (where de-

terministic equations for $\mu(\boldsymbol{x})$ and $k(\boldsymbol{x}, \boldsymbol{x}^{(i)})$ are derived) is less obvious and requires further investigation.

2.4. Estimating statistics using MLMC. The MC method requires a sufficiently large ensemble of Y to accurately estimate the mean and covariance matrix, which, in some applications, can be unpractical to obtain with high accuracy. To address this issue, instead of using aforementioned different sampling strategies, we replace the MC approximation of $\mu(\boldsymbol{x})$ and $\mathbf{C}$ in Eqs. (2.11) and (2.13) with MLMC ones. For simplicity, we demonstrate the idea via two-level MLMC. We use Y_L^m $(m = 1, ..., M_L)$ and Y_H^m $(m = 1, ..., M_H)$ to denote M_L low-accuracy and M_H high-accuracy realizations of the stochastic model for the system. We assume that Y_L^m and Y_H^m are realizations of the GP $Y_L : D_L \times \Omega \to \mathbb{R}$ and $Y_H : D_H \times \Omega \to \mathbb{R}$, respectively, and $Y_H = Y$ is the GP we want to identify. We also denote $\overline{Y}(\boldsymbol{x}) = Y_H(\boldsymbol{x}) - Y_L(\boldsymbol{x})$. For example, $D_L \subset \mathbb{R}^d$ and $D_H = D \subseteq \mathbb{R}^d$ can be coarse and fine grids in numerical simulations, respectively. Thus, Y_L and Y_H are low- and high-resolution random processes. In this case, when computing $\overline{Y}$, we interpolate Y_L from D_L to D_H. To simplify notations, we use Y_L to denote both the low-resolution random process on D_L and the interpolated random process from D_L to D_H in the MLMC formula. The mean of $Y_H(\boldsymbol{x})$ is estimated as

$$\mathbb{E}\left\{Y_H(\boldsymbol{x})\right\} = \mu(\boldsymbol{x}) \approx \mu_{_{MLMC}}(\boldsymbol{x}) = \frac{1}{M_L}\sum_{m=1}^{M_L} Y_L^m(\boldsymbol{x}) + \frac{1}{M_H}\sum_{m=1}^{M_H} \overline{Y}^m(\boldsymbol{x}), \tag{2.17}$$

which is the standard MLMC estimate of the mean [19]. In the past, MLMC was used only to estimate single point statistics, e.g., [3, 5, 6]. Here, we propose an MLMC estimate of the covariance function of $Y_H(\boldsymbol{x})$ based on the following relationship:

$$\begin{aligned} \mathrm{Cov}\left\{Y_H(\boldsymbol{x}), Y_H(\boldsymbol{x}')\right\} &= \mathrm{Cov}\left\{Y_L(\boldsymbol{x}) + \overline{Y}(\boldsymbol{x}), Y_L(\boldsymbol{x}') + \overline{Y}(\boldsymbol{x}')\right\} \\ &= \mathrm{Cov}\left\{Y_L(\boldsymbol{x}), Y_L(\boldsymbol{x}')\right\} + \mathrm{Cov}\left\{Y_L(\boldsymbol{x}), \overline{Y}(\boldsymbol{x}')\right\} \\ &\quad + \mathrm{Cov}\left\{\overline{Y}(\boldsymbol{x}), Y_L(\boldsymbol{x}')\right\} + \mathrm{Cov}\left\{\overline{Y}(\boldsymbol{x}), \overline{Y}(\boldsymbol{x}')\right\}. \end{aligned} \tag{2.18}$$

Because Y_L and $\overline{Y}$ are sampled independently in MLMC, we have

$$\mathrm{Cov}\left\{Y_L(\boldsymbol{x}), \overline{Y}(\boldsymbol{x}')\right\} = \mathrm{Cov}\left\{\overline{Y}(\boldsymbol{x}), Y_L(\boldsymbol{x}')\right\} = 0.$$

Thus,

$$\mathrm{Cov}\left\{Y_H(\boldsymbol{x}), Y_H(\boldsymbol{x}')\right\} = \mathrm{Cov}\left\{Y_L(\boldsymbol{x}), Y_L(\boldsymbol{x}')\right\} + \mathrm{Cov}\left\{\overline{Y}(\boldsymbol{x}), \overline{Y}(\boldsymbol{x}')\right\}, \tag{2.19}$$

and the unbiased MLMC approximation of the covariance is

$$\begin{aligned} &\mathrm{Cov}\left\{Y_H(\boldsymbol{x}), Y_H(\boldsymbol{x}')\right\} \approx k_{_{MLMC}}(\boldsymbol{x}, \boldsymbol{x}') \\ =&\frac{1}{M_L - 1}\sum_{m=1}^{M_L}\left(Y_L^m(\boldsymbol{x}) - \frac{1}{M_L}\sum_{m=1}^{M_L} Y_L^m(\boldsymbol{x})\right)\left(Y_L^m(\boldsymbol{x}') - \frac{1}{M_L}\sum_{m=1}^{M_L} Y_L^m(\boldsymbol{x}')\right) \\ &+ \frac{1}{M_H - 1}\sum_{m=1}^{M_H}\left(\overline{Y}^m(\boldsymbol{x}) - \frac{1}{M_H}\sum_{m=1}^{M_H} \overline{Y}^m(\boldsymbol{x})\right)\left(\overline{Y}^m(\boldsymbol{x}') - \frac{1}{M_H}\sum_{m=1}^{M_H} \overline{Y}^m(\boldsymbol{x}')\right). \end{aligned} \tag{2.20}$$

Finally, the MLMC-based PhIK model takes the form

$$\hat{y}(\boldsymbol{x}^*) = \mu_{_{MLMC}}(\boldsymbol{x}^*) + \boldsymbol{c}_{_{MLMC}}^{\mathsf{T}} \mathbf{C}_{_{MLMC}}^{-1}(\boldsymbol{y} - \boldsymbol{\mu}_{_{MLMC}}), \tag{2.21}$$

where $\boldsymbol{\mu}_{MLMC} = \left(\mu_{MLMC}(\boldsymbol{x}^{(1)}), \cdots, \mu_{MLMC}(\boldsymbol{x}^{(N)})\right)^{\mathsf{T}}$. The matrix $\mathbf{C}_{MLMC}$ and vector $\boldsymbol{c}_{MLMC}$ are approximations of $\mathbf{C}$ and $\boldsymbol{c}$ in Eq. (2.5) using Eq. (2.20). The MSE of this prediction is

$$\hat{s}^2(\boldsymbol{x}^*) = \sigma^2_{MLMC}(\boldsymbol{x}^*) - \boldsymbol{c}^{\mathsf{T}}_{MLMC}\mathbf{C}^{-1}_{MLMC}\boldsymbol{c}_{MLMC}, \tag{2.22}$$

where $\sigma^2_{MLMC}(\boldsymbol{x}^*)$ is computed from Eq. (2.20) by replacing $\boldsymbol{x}$ and $\boldsymbol{x}'$ with $\boldsymbol{x}^*$. The following corollary is a straightforward extension of Theorem 2.1 for PhIK with the mean and covariance obtained from MLMC.

COROLLARY 2.3. *Assume that* $\{Y_H^m(\boldsymbol{x})\}_{m=1}^{M_H}$ *and* $\{Y_L^m(\boldsymbol{x})\}_{m=1}^{M_L}$ *are finite ensembles of approximated realizations of stochastic models* $u_{H}(\boldsymbol{x};\omega)$ *and* $u_{L}(\boldsymbol{x};\omega)$*, where* $\|\mathcal{L}u_{H}(\boldsymbol{x};\omega) - g(\boldsymbol{x};\omega)\| < \epsilon_{H}$ *and* $\|\mathcal{L}u_{L}(\boldsymbol{x};\omega) - g(\boldsymbol{x};\omega)\| < \epsilon_{L}$ *for any* $\omega \in \Omega$*, and* $\mathcal{L}$*,* $\|\cdot\|$*,* $g(\boldsymbol{x};\omega)$*, and* $\overline{g(\boldsymbol{x})}$ *are given in Theorem 2.1. The MLMC-based PhIK prediction* $\hat{y}(\boldsymbol{x})$ *satisfies*

$$\|\mathcal{L}\hat{y}(\boldsymbol{x}) - \overline{g(\boldsymbol{x})}\| \leq C_{H}\epsilon_{H} + C_{L}\epsilon_{L} + \sigma(g(\boldsymbol{x};\omega^m))\sum_{i=1}^{N}\tilde{a}_i\sigma(Y_L^m(\boldsymbol{x}^{(i)})), \tag{2.23}$$

where

$$C_{H} = 1 + 2\sum_{i=1}^{N}\tilde{a}_i\sqrt{\frac{M_H}{M_H-1}}\sigma(\overline{Y}^m(\boldsymbol{x}^{(i)})),$$
$$C_{L} = 2 + 2\sum_{i=1}^{N}\tilde{a}_i\left(\sqrt{\frac{M_L}{M_L-1}}\sigma(Y_L^m(\boldsymbol{x}^{(i)})) + \sqrt{\frac{M_H}{M_H-1}}\sigma(\overline{Y}^m(\boldsymbol{x}^{(i)}))\right),$$

and $\tilde{a}_i$ *is the* i*-th entry of* $\mathbf{C}^{-1}_{MLMC}(\boldsymbol{y} - \boldsymbol{\mu}_{MLMC})$ *bounded by* $\|\mathbf{C}^{-1}_{MLMC}\|_2\|\boldsymbol{y} - \boldsymbol{\mu}_{MLMC}\|_2$*. Here,* $\sigma(g(\boldsymbol{x};\omega^m))$ *is defined in Theorem 2.1,* $\sigma(Y_L^m(\boldsymbol{x}^{(i)}))$ *and* $\sigma(\overline{Y}^m(\boldsymbol{x}^{(i)}))$ *are standard deviation of data sets* $\{Y_L^m(\boldsymbol{x}^{(i)})\}_{m=1}^{M_L}$ *and* $\{\overline{Y}^m(\boldsymbol{x}^{(i)})\}_{m=1}^{M_H}$*, respectively.*

We present the proof in Appendix B.

It is uncomplicated to extend the two-level MC to a general L-level MLMC. We present the following theorem for the L-level ($L > 2$) MLMC-based PhIK error bounds. The proof of this theorem immediately follows from Theorem 2.1 and Collorary 2.3.

THEOREM 2.2. *Assume that* $\{Y_l^m(\boldsymbol{x})\}_{m=1}^{M_l}, l = 1, \cdots, L$ *are finite ensembles of realizations of stochastic models* $u_l(\boldsymbol{x};\omega), l = 1, \cdots, L$*. Denote* $\overline{Y}_l = Y_l - Y_{l-1}$ *for* $l = 2, \cdots, L$ *and* $\overline{Y}_1 = Y_1$*. The MLMC-based PhIK prediction* $\hat{y}(\boldsymbol{x})$ *can be given as*

$$\hat{y}(\boldsymbol{x}) = \mu_{MLMC}(\boldsymbol{x}) + \sum_{i=1}^{N}\tilde{a}_i k_{MLMC}(\boldsymbol{x}, \boldsymbol{x}^{(i)}), \tag{2.24}$$

where

$$\mu_{MLMC}(\boldsymbol{x}) = \sum_{l=0}^{L}\frac{1}{M_l}\sum_{m=1}^{M_L}\overline{Y}_l(\boldsymbol{x});$$

$$k_{MLMC}(\boldsymbol{x}, \boldsymbol{x}') = \sum_{l=0}^{L}\frac{1}{M_l - 1}\sum_{m=1}^{M_L}\left(\overline{Y}_l^m(\boldsymbol{x}) - \frac{1}{M_l}\sum_{m=1}^{M_L}\overline{Y}_l^m(\boldsymbol{x})\right)\left(\overline{Y}_l^m(\boldsymbol{x}') - \frac{1}{M_l}\sum_{m=1}^{M_L}\overline{Y}_l^m(\boldsymbol{x}')\right);$$

and $\tilde{a}_i = (\mathbf{C}_{MLMC}^{-1}(\boldsymbol{y} - \boldsymbol{\mu}_{MLMC}))_i$, $\boldsymbol{\mu}_{MLMC} = (\mu_{MLMC}(\boldsymbol{x}^{(1)}), \cdots, \mu_{MLMC}(\boldsymbol{x}^{(N)}))^\mathsf{T}$, $(\mathbf{C}_{MLMC})_{ij} = k_{MLMC}(\boldsymbol{x}^{(i)}, \boldsymbol{x}^{(j)})$. *Let* $\mathcal{L}$, $g(\boldsymbol{x};\omega)$, $\overline{g(\boldsymbol{x})}$, *and* $\|\cdot\|$ *be given as in Theorem 2.1.*

1) If $g(\boldsymbol{x};\omega)$ *is a deterministic function, i.e.,* $g(\boldsymbol{x};\omega) = g(\boldsymbol{x})$, *and* u_l *satisfies* $\mathcal{L}u_l(\boldsymbol{x};\omega) = g(\boldsymbol{x})$ *for any* $\omega \in \Omega$ *and for* $l = 1, \cdots, L$, *then* $\mathcal{L}\hat{y}(\boldsymbol{x}) = g(\boldsymbol{x})$.

2) If Y_l *satisfies* $\|\mathcal{L}Y_l(\boldsymbol{x};\omega) - g(\boldsymbol{x};\omega)\| \le \epsilon_l$ *for* $l = 1, \cdots, L$, *then*

$$\|\mathcal{L}\hat{y}(\boldsymbol{x})\| \le \sum_{l=1}^{L} C_l\epsilon_l + \sigma(g(\boldsymbol{x};\omega^m))\sum_{i=1}^{N}\tilde{a}_i\sigma(Y_L^m(\boldsymbol{x}^{(i)})), \tag{2.25}$$

where

$$C_l = \begin{cases} 1 + 2\sum_{i=1}^{N}\tilde{a}_i\sqrt{\dfrac{M_l}{M_l-1}}\sigma(\overline{Y}_l^m(\boldsymbol{x}^{(i)})), & l = L; \\ 2 + 2\sum_{i=1}^{N}\tilde{a}_i\left(\sqrt{\dfrac{M_l}{M_l-1}}\sigma(\overline{Y}_l^m(\boldsymbol{x}^{(i)})) + \sqrt{\dfrac{M_{l+1}}{M_{l+1}-1}}\sigma(\overline{Y}_{l+1}^m(\boldsymbol{x}^{(i)}))\right), & 1 \le l < L. \end{cases}$$

Moreover, $\tilde{a}_i$ *is bounded by* $\|\mathbf{C}_{MLMC}^{-1}\|_2\|\boldsymbol{y} - \boldsymbol{\mu}_{MLMC}\|_2$.

Of note, MLMC estimates for variance and higher-order single point (i.e., a fixed $\boldsymbol{x} \in \mathbb{R}^d$) statistical moments was proposed in [3, 5, 6]. We note that the covariance estimate Eq. (2.20) proposed herein also can be used to estimate variance by setting $\boldsymbol{x}' = \boldsymbol{x}$. The systematic convergence analysis of the MLMC can be found in [19, 3, 8, 5, 6]. Other multifidelity methods, e.g., [18, 62], also can be used as long as they compute the mean and covariance efficiently.

Moreover, in standard MC, if only a small number of Y realizations $\{Y^m\}_{m=1}^M$ is available to approximate $\mathbf{C}$ with $\mathbf{C}_{MC}$ in Eq. (2.13), in addition to having a large statistical error, the matrix $\mathbf{C}_{MC}$ is not full rank if $N \ge M$. This is because the size of $\mathbf{C}_{MC}$ is $N \times N$, but its rank is, at most, $M - 1$. This is common in practical problems where N is large and M is small due to the computational cost. Therefore, $\mathbf{C}_{MC}$ is not invertible, and it is necessary to add a matrix, e.g., $\alpha\mathbf{I}$, to stabilize the algorithm, where $\mathbf{I}$ is the identity matrix and α is a small number. Even when the observation noise is included as $\mathbf{C}_{MC} + \delta^2\mathbf{I}$, a small ensemble $\{Y^m\}$ will result in a large condition number of the resulting matrix because of the rank deficit if the noise δ is not large enough. Multifidelity methods such as MLMC can help to alleviate the ill-conditioning issue by incorporating a sufficiently large low-fidelity (low-resolution) ensemble.

2.5. Active learning. In this context, *active learning* (e.g., [9, 24, 46, 10]) is a process of identifying locations for additional observations that minimize the prediction error and reduce MSE or uncertainty. In the GPR framework, a natural way is to add observations at the locations corresponding to local maxima in $s^2(\boldsymbol{x})$, e.g., [16, 37]. Then, we can make a new prediction $\hat{y}(\boldsymbol{x})$ for $\boldsymbol{x} \in D$ and compute a new $\hat{s}^2(\boldsymbol{x})$ to select the next location for additional observation (see Algorithm 2.1). Such treatment differs from other sensor placement methods based on deterministic approximation of unknown fields (e.g., [60, 56]). This selection criterion is based on the statistical interpretation of the interpolation.

Notably, Algorithm 2.1 is a greedy algorithm to identify additional observation locations when some observations are affordable. It cannot guarantee to identify the optimal new observation locations. More sophisticated algorithms can be

Algorithm 2.1 Active learning based on GPR

1: Specify the locations $\boldsymbol{X}$, corresponding observation $\boldsymbol{y}$, and the maximum number of observation $N_{\max}$ affordable. The number of available observations is denoted as N.
2: **while** $N_{\max} > N$ **do**
3: Compute the MSE $\hat{s}^2(\boldsymbol{x})$ of MLE prediction $\hat{y}(\boldsymbol{x})$ for $\boldsymbol{x} \in D$.
4: Locate the location $\boldsymbol{x}_m$ for the maximum of $\hat{s}^2(\boldsymbol{x})$ for $\boldsymbol{x} \in D$.
5: Obtain observation y_m at $\boldsymbol{x}_m$ and set $\boldsymbol{X} = \{\boldsymbol{X}, \boldsymbol{x}_m\}, \boldsymbol{y} = (\boldsymbol{y}^\mathsf{T}, y_m)^\mathsf{T}, N = N + 1$.
6: **end while**
7: Construct the MLE prediction of $\hat{y}(\boldsymbol{x})$ on D using $\boldsymbol{X}$ and $\boldsymbol{y}$.

found in literature, e.g., [24, 26], and PhIK is complementary to these methods because it provides the GP. Also, it is not necessary that the new observations are added one by one. Roughly speaking, if there are several maxima of $\hat{s}^2(\boldsymbol{x})$ and they are not clustered (to avoid potential ill-conditioning of $\mathbf{C}$ in some cases), the observations at these locations can be added simultaneously. In this work, we add new observations one by one in the numerical examples for demonstration purposes. The efficiency of the active learning algorithm depends on the correlation $\mathrm{Cor}\{Y(\boldsymbol{x}), Y(\boldsymbol{x}')\} = \mathrm{Cov}\{Y(\boldsymbol{x}), Y(\boldsymbol{x}')\}/(\sigma(Y(\boldsymbol{x}))\sigma(Y(\boldsymbol{x}')))$. Intuitively, if the correlation is large, then adding a new observation will provide information in a large neighborhood of this location, reducing the MSE in a large region. An extreme example is that when $\mathrm{Cor}\{Y(\boldsymbol{x}), Y(\boldsymbol{x}')\} \equiv 1$ (e.g., the correlation length of the GP is infinite), only one observation is needed to reconstruct the field. On the other hand, if the correlation is small (e.g., the correlation length of the GP is small), an observation can only influence a small neighborhood, which will require more observations to reduce the uncertainty in the prediction of the entire domain. An extreme example of this scenario is $\mathrm{Cor}\{Y(\boldsymbol{x}), Y(\boldsymbol{x}')\} \equiv 0$. Unless we have observations everywhere, the MSE in the prediction at the locations with no observations is unchanged no matter how many observations we have because at these locations $\boldsymbol{c} = \mathbf{0}$ in Eq. (2.7).

There is a large body of literature in statistics and machine learning on the *learning curve* that describes the average MSE over D as a function of N, the number of available observations, e.g., [58, 28, 49, 40]. Both noisy and noiseless scenarios have been studied, and we refer interested readers to the aforementioned literatures.

3. Numerical examples. We present two numerical examples to demonstrate the performance of PhIK. Both numerical examples are two-dimensional in physical space. In the first example, we use the MC-based PhIK introduced in Section 2.3, and in the second example, we employ the MLMC-based PhIK presented in Section 2.4. We compare PhIK with the ordinary Kriging (in the following, we refer to the ordinary Kriging as Kriging). In the Kriging method, we tested the Gaussian kernel and Matérn kernel. We do not observe significant difference in the results and only report solutions obtained with the Gaussian kernel.

3.1. Branin function. We consider the following modified Branin function [16]:

$$f(x, y) = a(\bar{y} - b\bar{x}^2 + c\bar{x} - r)^2 + g(1 - p)\cos(\bar{x}) + g + qx, \tag{3.1}$$

where

$$\bar{x} = 15x - 5,\ \bar{y} = 15y,\ (x, y) \in D = [0, 1] \times [0, 1],$$

and

$$a = 1,\ b = 5.1/(4\pi^2),\ c = 5/\pi,\ r = 6,\ g = 10,\ p = 1/(8\pi),\ q = 5.$$

The contour of f and eight randomly chosen observation locations are presented in Figure 3.1. The function f is evaluated on a 41×41 uniform grid, and we denote the resulting discrete field (a 41×41 matrix) as $\boldsymbol{F}$. We will compare reconstruction of $\boldsymbol{F}$ by different methods, and we denote the reconstructed field as $\boldsymbol{F}_r$.

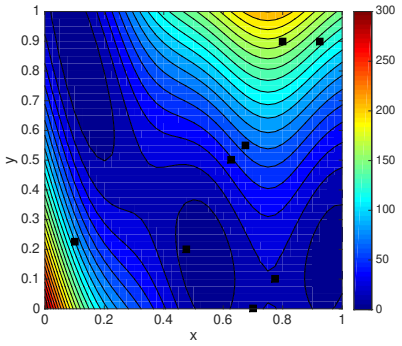
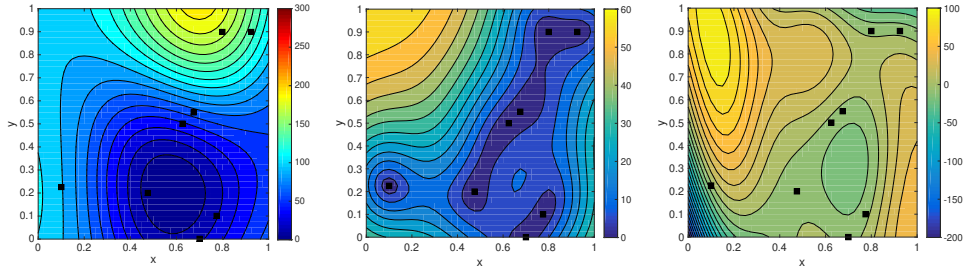


FIG. 3.1. *Contours of modified Branin function (on* 41×41 *uniform grids) and locations of eight observations (black squares).*

3.1.1. Field reconstruction. We first use Kriging to reconstruct $\boldsymbol{F}$ based on the eight observation data sets. Figure 3.2(a) presents the reconstructed field $\boldsymbol{F}_r$ by Kriging, and Figure 3.2(b) depicts the RMSE of this reconstruction, which shows the error from the statistical point of view. The difference $\boldsymbol{F}_r - \boldsymbol{F}$ is shown in Figure 3.2(c), which quantifies the $\hat{y}$ deviation from the ground truth. Apparently, this reconstruction deviates considerably from $\boldsymbol{F}$ in Figure 3.1, especially in the region $[0, 0.5] \times [0.5, 1]$. This is consistent with Figure 3.2(b) as the RMSE is large in this region. This is because there is no observation in this region. We later show that adding observations guided by active learning increases the reconstruction accuracy.



(a) (b) (c)

FIG. 3.2. *Reconstruction of the modified Branin function by Kriging: (a) reconstructed field* $\boldsymbol{F}_r$*; (b) RMSE* $\hat{s}$ *of the reconstruction; (c) difference* $\boldsymbol{F}_r - \boldsymbol{F}$*.*

Next, we assume that based on "domain knowledge", $f(x, y)$ is partially known, i.e., its form is known, but the coefficients b and q are unknown. Then, we treat these coefficients as random fields $\hat{b}$ and $\hat{q}$, which indicates that the field f is described by

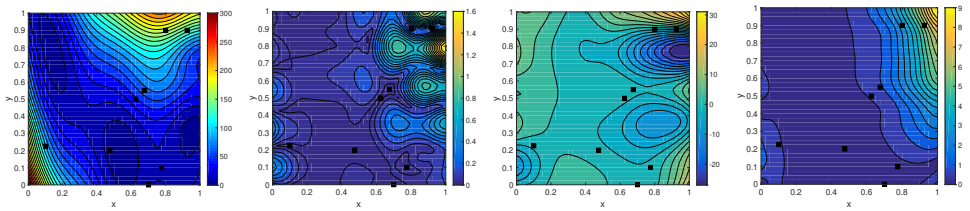
a random function $\hat{f} : D \times \Omega \to \mathbb{R}$:

$$\hat{f}(x,y;\omega) = a(\bar{y} - \hat{b}(x,y;\omega)\bar{x}^2 + c\bar{x} - r)^2 + g(1-p)\cos(\bar{x}) + \hat{g} + \hat{q}(x,y;\omega)x, \tag{3.2}$$

$$\hat{b}(x,y;\omega) = b\Big\{0.9 + \frac{0.2}{\pi}\sum_{i=1}^{3}\Big[\frac{1}{4i-1}\sin((2i-0.5)\pi x)\xi_{2i-1}(\omega) + \frac{1}{4i+1}\sin((2i+0.5)\pi y)\xi_{2i}(\omega)\Big]\Big\},$$

$$\hat{q}(x,y;\omega) = q\Big\{1.0 + \frac{0.6}{\pi}\sum_{i=1}^{3}\Big[\frac{1}{4i-3}\cos((2i-1.5)\pi x)\xi_{2i+5}(\omega) + \frac{1}{4i-1}\cos((2i-0.5)\pi y)\xi_{2i+6}(\omega)\Big]\Big\},$$

and $\{\xi_i(\omega)\}_{i=1}^{12}$ are i.i.d. Gaussian random variables with zero mean and unit variance. Further, we allow for the model error by setting $\hat{g} = 20$ different from $g = 10$ in Eq (3.2). We use this partial knowledge to compute the mean and covariance function of $\hat{f}$ by generating $M = 1000$ samples of $\xi_i(\omega)$ and evaluating $\hat{f}$ on the 41×41 uniform grid for each sample of $\xi_i(\omega)$. We denote these realizations of $\hat{f}$ as $\{\hat{\boldsymbol{F}}^m\}_{m=1}^M$. Figure 3.3 presents the reconstructed field $\boldsymbol{F}_r$, RMSE, and the difference from the exact field $\boldsymbol{F}$. These results are much better than those found by Kriging as both the reconstruction error and the RMSE are much smaller. More significantly, the RMSE in PhIK is much smaller than Kiriging in the $[0, 0.5] \times [0.5, 1]$ subdomain with no observations. This is because in PhIK, the covariance matrix is computed by the ensembles of physics-based model. Figure 3.3(d) shows σ_{MC}, the standard deviation of $\{\hat{\boldsymbol{F}}^m\}_{m=1}^M$, i.e., ensemble of "physics-based model" in this case. Note that σ_{MC} is a measure of uncertainty in the physical model $\hat{f}$. Figure 3.3 demonstrates that σ_{MC} has a similar pattern as RMSE (which is a measure of uncertainty in PhIK), but larger magnitude. It demonstrates that PhIK reduces uncertainty by conditioning the prediction of f on observations.

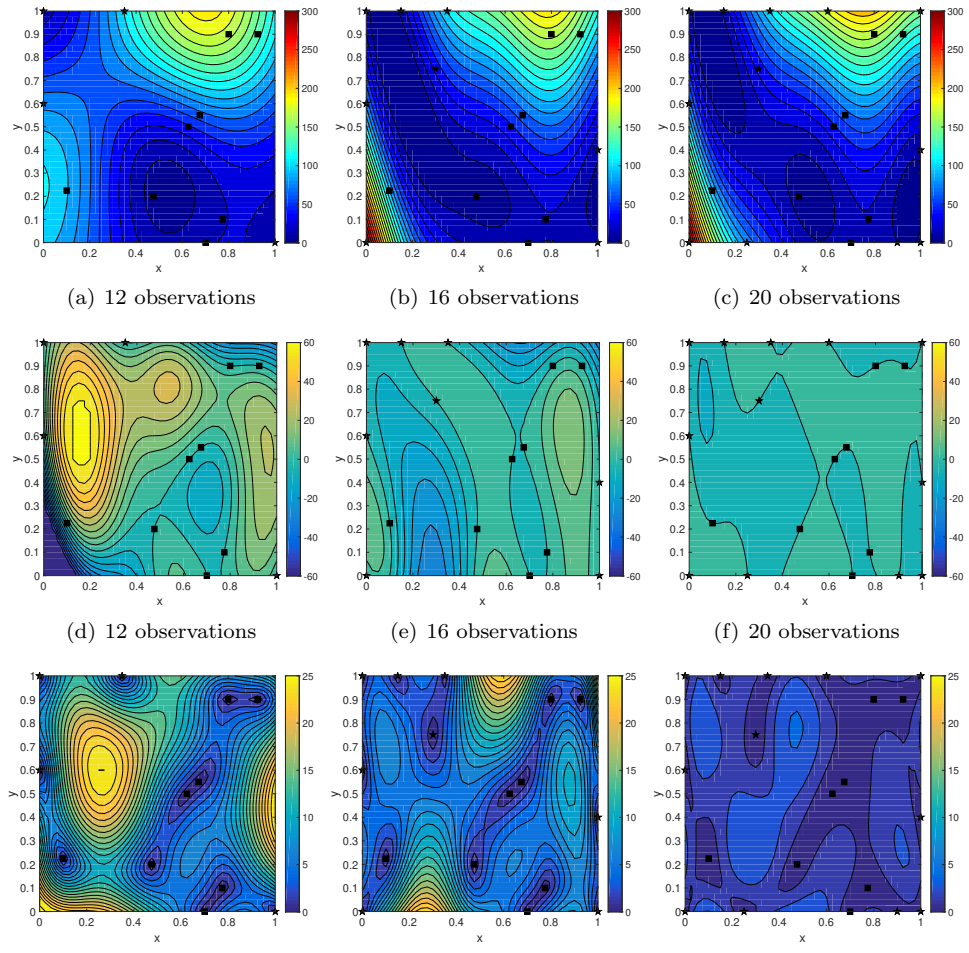


(a) (b) (c) (d)

FIG. 3.3. *Reconstruction of the modified Branin function by PhIK: (a) reconstructed field $\boldsymbol{F}_r$; (b) RMSE $\hat{s}$; (c) $\boldsymbol{F}_r - \boldsymbol{F}$; (d) standard deviation of the ensemble $\hat{\boldsymbol{F}}^m$, i.e., σ_{MC}.*

3.1.2. Active learning. After obtaining $\hat{s}^2$, we use Algorithm 2.1 to perform active learning by adding one by one new observations of f at (x, y) where $\hat{s}^2$ has maximum. Figure 3.4 displays locations of additional observations and resulting Kriging prediction. In this figure, the first row is the field $\boldsymbol{F}_r$ reconstructed by Kriging, the

second row shows corresponding errors $\boldsymbol{F}_r - \boldsymbol{F}$, and the third row presents the $\boldsymbol{F}_r$ RMSE. The three columns correspond to results with 12, 16, and 20 observations. The initial eight observations are marked as squares, and added observations are marked as stars. As expected, the reconstruction accuracy increases as more observations are added, and the uncertainty in the reconstruction decreases (indicated in the third row). Notably, the active learning algorithm "places" most observation points on ∂D where the variance of f is largest. This illustrates that the GPR is more accurate for interpolation than extrapolation, and most original observations are within the domain. As such, the results are extrapolated toward the boundary.



(a) 12 observations (b) 16 observations (c) 20 observations

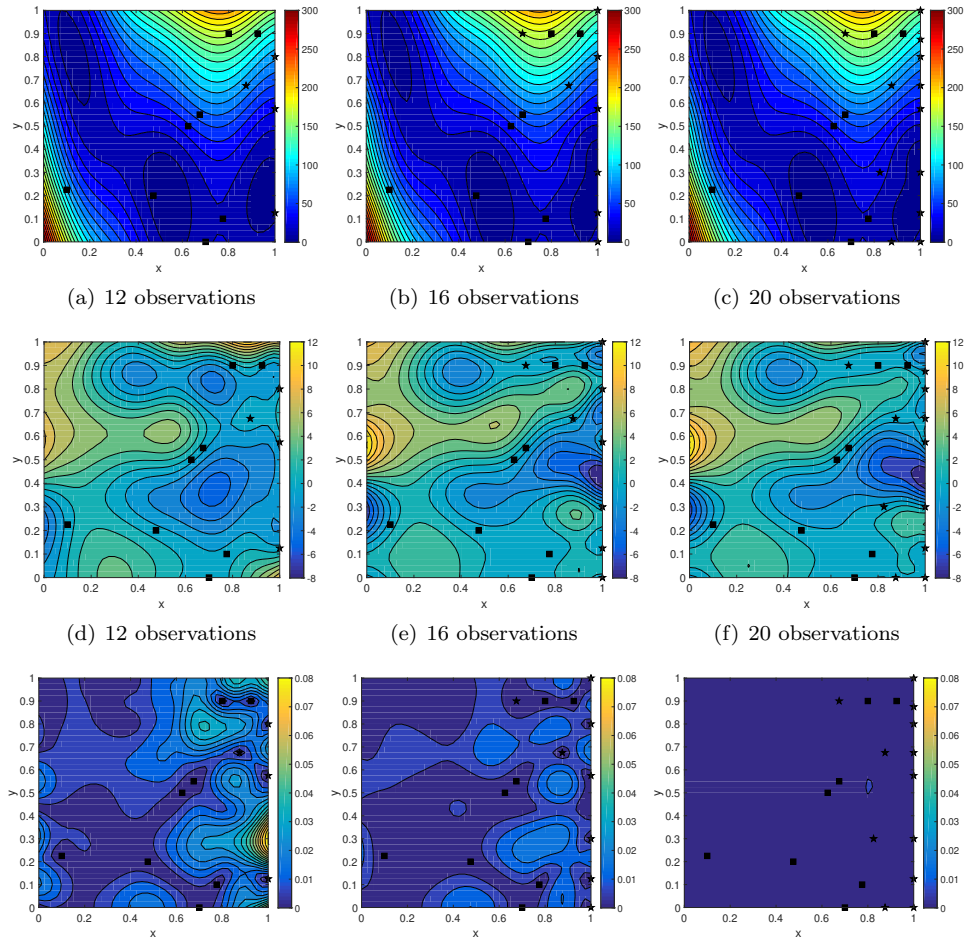
(d) 12 observations (e) 16 observations (f) 20 observations

(g) 12 observations (h) 16 observations (i) 20 observations

FIG. 3.4. *Reconstruction of the modified Branin function by Kriging via active learning. Black squares are the locations of the original eight observation, and stars are newly added observations. First row: reconstructed field* $\boldsymbol{F}_r$*; second row:* $\boldsymbol{F}_r - \boldsymbol{F}$*; third row: RMSE* $\hat{s}$*.*

Next, we use PhIK combined with active learning. Figure 3.5 shows the results. The first row shows $\boldsymbol{F}_r$, estimated by PhIK, the second row includes $\boldsymbol{F}_r - \boldsymbol{F}$, and the third row presents $\hat{s}$. The three columns correspond to results with 12, 16, and 20 observations, respectively. The initial eight observations are marked as squares, and

added observations are marked as stars. The accuracy of the reconstruction in these three columns is close as shown in the second row, and all three are much better than the results obtained with Kriging in Figure 3.4. On the other hand, the third row in Figure 3.5 demonstrates that the uncertainty in the reconstruction decreases as more observations are available. This indicates that decrease of $\hat{s}$ does not necessarily lead to a reduction in the difference between $\boldsymbol{F}_r$ and $\boldsymbol{F}$.



(a) 12 observations (b) 16 observations (c) 20 observations

(d) 12 observations (e) 16 observations (f) 20 observations

(g) 12 observations (h) 16 observations (i) 20 observations

FIG. 3.5. *Reconstruction of the modified Branin function by PhIK via active learning. Black squares mark the locations of the original eight observations, and stars are newly added observations. First row: reconstructed field* $\boldsymbol{F}_r$*; second row:* $\boldsymbol{F}_r - \boldsymbol{F}$*; third row: RMSE* $\hat{s}$*.*

Figure 3.6 shows the relative error $\|\boldsymbol{F}_r - \boldsymbol{F}\|_F / \|\boldsymbol{F}\|_F$ ($\|\cdot\|_F$ is the Frobenius norm) in Kriging and PhIK as a function of the observation numbers, where the first eight are the "original" observations and the rest are added according to the active learning algorithm. With the original eight observations, the PhIK result (about 8% error) is much better than the Kriging (more than 50% error). As more observations are added by the active learning algorithm, the error of Kriging decreases almost linearly to approximately 4% (20 observations). The error of PhIK reduces from 8% to 4% (10 observations). Adding additional observations does little to improve the accuracy.

For 20 observations, the accuracy of Kriging and PhIK is approximately the same. In both Kriging and PhIK, the accuracy of regression generally increases with the number of observation points. In Kriging, the accuracy increases because the accuracy of the mean and covariance estimates increase with the number of observation points. In PhIK, the mean and covariance are decided by the ensemble $\hat{\boldsymbol{F}}^m$ only. Thus, they are unchanged as more observations are made available. Of note, f is not a realization of $\hat{f}$, and the PhIK prediction can be considered as an approximation of the "projection" of f on the linear space spanned by $\{\hat{\boldsymbol{F}}^m\}_{m=1}^M$. This approximation relies on the observation data and the covariance matrix of $\{\hat{\boldsymbol{F}}^m\}_{m=1}^M$. In this example, 12 observations are sufficient to obtain a very accurate projection, therefore adding more observations only leads to small improvements in the f prediction. We refer interested readers to literature on reproducing kernel Hilbert space for approximation accuracy e.g., [59, 4]. Here, we choose the random model $\hat{f}$ to model incomplete knowledge. A more accurate stochastic model (i.e., a smaller distance between f and the linear space spanned by $\{\hat{\boldsymbol{F}}^m\}_{m=1}^M$) is expected to result in the more accurate PhIK prediction.

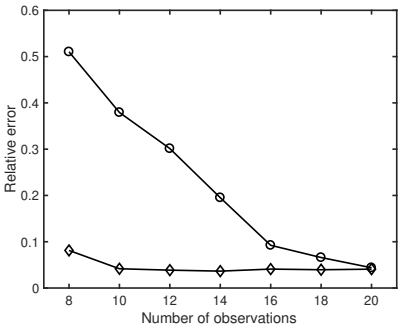


FIG. 3.6. *Relative error of reconstructed modified Branin function* $\|\boldsymbol{F}_r - \boldsymbol{F}\|_F / \|\boldsymbol{F}\|_F$ *using Kriging ("∘") and PhIK ("◇") with different numbers of total observations via active learning.*

3.2. Solute transport in heterogeneous porous media. In the second example, we consider steady-state flow and advection and dispersion of conservative tracer with concentration $C_e(\boldsymbol{x}, t)$ in a heterogeneous porous medium with known initial and boundary conditions and the unknown hydraulic conductivity $K(\boldsymbol{x})$. We assume that measurements of $C_e(\boldsymbol{x}, t)$ are available at several locations at different times. The flow and transport in porous media can be described by conservation laws, including a combination of the continuity equation and Darcy law:

$$
\begin{cases}
\nabla \cdot [K(\boldsymbol{x};\omega)\nabla h(\boldsymbol{x};\omega)] = 0, & \boldsymbol{x} \in D, \\
\dfrac{\partial h(\boldsymbol{x};\omega)}{\partial \boldsymbol{n}} = 0, & x_2 = 0 \quad \text{or} \quad x_2 = L_2, \\
h(x_1 = 0, x_2;\omega) = H_1 \quad \text{and} \quad h(x_1 = L_1, x_2;\omega) = H_2,
\end{cases} \tag{3.3}
$$

where $D = [0, L_1] \times [0, L_2] = [0, 256] \times [0, 128]$, the unknown conductivity is modeled as the random log-normally distributed field $K(\boldsymbol{x};\omega) = \exp(Z(\boldsymbol{x};\omega))$ with the known exponential covariance function $\mathrm{Cov}\{Z(\boldsymbol{x}), Z(\boldsymbol{x}')\} = \sigma_Z^2 \exp(-|\boldsymbol{x} - \boldsymbol{x}'|/l_z)$ with the variance $\sigma_Z^2 = 2$, correlation length $l_z = 5$, $h(\boldsymbol{x};\omega)$ is the hydraulic head, and $\omega \in \Omega$.

The solute transport is is governed by the advection-dispersion equation [14, 27]:

$$
\begin{cases}
\dfrac{\partial C(\boldsymbol{x},t;\omega)}{\partial t} + \nabla\cdot(\boldsymbol{v}(\boldsymbol{x};\omega)C(\boldsymbol{x},t;\omega)) = \\
\qquad\qquad \nabla\cdot\left[\left(\dfrac{D_w}{\tau} + \boldsymbol{\alpha}\|\boldsymbol{v}(\boldsymbol{x};\omega)\|_2\right)\nabla C(\boldsymbol{x},t;\omega)\right], \qquad \boldsymbol{x} \text{ in } D, \\
C(\boldsymbol{x},t=0;\omega) = \delta(\boldsymbol{x}-\boldsymbol{x}^*), \\
\dfrac{\partial C(\boldsymbol{x};\omega)}{\partial \boldsymbol{n}} = 0, \qquad x_2 = 0 \quad \text{or} \quad x_2 = L_2 \quad \text{or} \quad x_1 = L_1, \\
C(x_1 = 0, x_2;\omega) = 0,
\end{cases}
\tag{3.4}
$$

where $C(\boldsymbol{x},t;\omega)$ is the solute concentration defined on $D\times[0,T]\times\Omega$, the solute is instantaneously injected at $\boldsymbol{x}^* = (50, 64)$, $\boldsymbol{v}(\boldsymbol{x};\omega) = -K(\boldsymbol{x};\omega)\nabla h(\boldsymbol{x};\omega)/\phi$ is the average pore velocity, ϕ is the porosity, D_w is the diffusion coefficient, τ is the tortuosity, and $\boldsymbol{\alpha}$ is the dispersivity tensor with the diagonal components α_L and α_T. In the present work, the transport parameters are set to $\phi = 0.317$, $\tau = \phi^{1/3}$, $D_w = 2.5\times10^{-5}$ m^2/s, $\alpha_L = 5$ m, and $\alpha_T = 0.5$ m.

We generate $M = 1000$ realizations of $Z(\boldsymbol{x})$ using the SGSIM (sequential Gaussian simulation) code [13] and solve the governing equations for each realization of $K(\boldsymbol{x}) = \exp(Z(\boldsymbol{x}))$ using the finite volume code STOMP (subsurface transport over multiple phases) [47] with the grid size 1m $\times$ 1m. Both, PhIK and Kriging independently regress data each time the concentration data are available. Here, we show the results of PhIK and Kriging at $t = 8$ days. The ground truth is generated as one of the 1000 solutions of the governing equations and is shown in Figure 3.7 with observation locations. We assume that six uniformly spaced observations are available near the domain boundary, and nine randomly placed observations are given within the domain D. Because Kriging is known to be less accurate for extrapolation (as illustrated in the first numerical example), it is common to collect data near the boundary of the domain of interest in practice, e.g., [11].

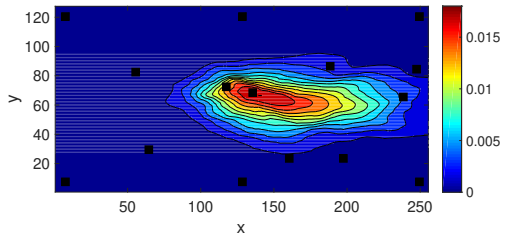
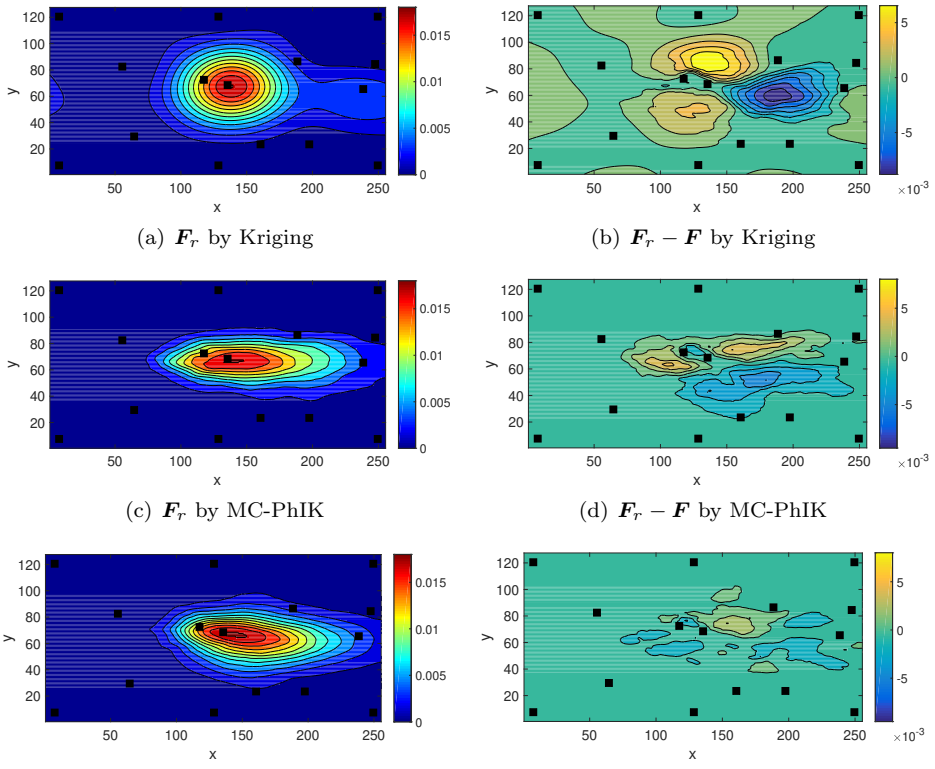


FIG. 3.7. *Ground truth of the solute concentration when $t = 8$ days and observation locations (black squares).*

3.2.1. Field reconstruction. We use the matrix $\boldsymbol{F}$ to denote the ground truth. We first use Kriging to reconstruct $\boldsymbol{F}$ using 15 observations. Figures 3.8(a) and (b) present the reconstructed field $\boldsymbol{F}_r$ and the error $\boldsymbol{F}_r - \boldsymbol{F}$. We can see that Kriging performs poorly as the relative error $\|\boldsymbol{F}_r - \boldsymbol{F}\|_F/\|\boldsymbol{F}\|_F$ is more than 50%. Next, we assume that only 10 simulations (i.e., $M_H = 10$) with grid size 1×1 are available and use them in the MC-based PhIK to reconstruct $\boldsymbol{F}$. Specifically, the mean and covariance matrix are computed from Eqs. (2.11) and (2.13) using ensembles $\{\hat{\boldsymbol{F}}_H^m\}_{m=1}^{10}$ (simulations with grid size 1×1). Figure 3.8(c) and (d) present $\boldsymbol{F}_r$ and $\boldsymbol{F}_r - \boldsymbol{F}$, respectively. These results are better than the Kriging as the relative error is less than

30%. Finally, we assume that additional 500 coarse-resolution simulations are available with grid size 4m × 4m and use MLMC Eqs. (2.17) and (2.20), to approximate the mean and covariance matrix. The reconstructed field and the difference from the ground truth are presented in Figure 3.8(e) and (f), respectively. The coarse simulations significantly improve prediction as the relative error reduces to approximately 14%.



(a) $\boldsymbol{F}_r$ by Kriging (b) $\boldsymbol{F}_r - \boldsymbol{F}$ by Kriging

(c) $\boldsymbol{F}_r$ by MC-PhIK (d) $\boldsymbol{F}_r - \boldsymbol{F}$ by MC-PhIK

(e) $\boldsymbol{F}_r$ by MLMC-PhIK (f) $\boldsymbol{F}_r - \boldsymbol{F}$ by MLMC-PhIK

FIG. 3.8. *Reconstructed solute concentration field* $\boldsymbol{F}_r$ *by Kriging, MC-based PhIK with* 10 *high-resolution simulations, MLMC-based PhIK with* 10 *high-resolution (grid size* 1×1*) simulations and* 500 *low-resolution (grid size* 4×4*) simulations, and their difference from the exact field* $\boldsymbol{F}_r - \boldsymbol{F}$*. Black squares are the observations.*

Next, we study how the MLMC-based PhIK's accuracy depends on the number of high-resolution simulations M_H for the fixed number of low-resolution simulations $M_L = 500$. Figure 3.9 shows how the MLMC-based PhIK error $\|\boldsymbol{F}_r - \boldsymbol{F}\|_F / \|\boldsymbol{F}\|_F$ decreases with increasing M_H. For comparison, we also compute error in the MC-based PhIK for the same number of M_H. It is clear that MC-based PhIK is less accurate than MLMC-based PhIK, especially for small M_H. Also, the smaller error in MLMC-based PhIK is achieved with a smaller computational cost than that of MC-based PhIK. In this example, the number of degrees of freedom in the low-resolution simulation is 1/16 of that in the high-resolution simulation. For an implicit scheme for the dispersion operator and an explicit scheme for the advection operator, according to the CFL condition, the time step in a low-resolution simulation is approximately four times larger than the time step in a high-resolution simulation. Therefore, the

computational cost of a high-resolution simulation is at least 64 times that of a low-resolution simulation and the cost of 500 low-resolution simulations is less than eight high-resolution ones. Thus, for the considered problem, the MLMC-based PhIK using 10 high-resolution and 500 low-resolution simulations is less costly than MC-based PhIK with 18 high-resolution simulations, while its accuracy is better than the latter with 90 high-resolution simulations (as shown in Figure 3.9). Equally important, the matrix $\mathbf{C}_{MC}$ (size 15×15) computed from Eq. (2.13) with only 10 high-resolution simulations is not full rank, so we must add a regularization term $\alpha\mathbf{I}$ to $\mathbf{C}_{MC}$. As discussed in Section 2.4, MLMC with additional low-resolution simulations eliminates the rank deficiency caused by insufficient number of realizations. Moreover, we denote the cost

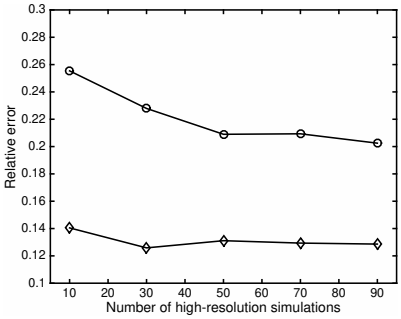


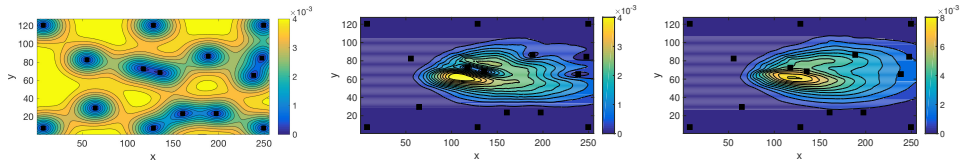
FIG. 3.9. *Relative error of solute concentration* $\|\boldsymbol{F}_r - \boldsymbol{F}\|_F/\|\boldsymbol{F}\|_F$ *by PhIK using different numbers of high-resolution simulations (grid size* 1×1*) only ("∘") and* 500 *low-resolution simulations (grid size* 4×4*) in addition to different numbers of high-resolution simulations ("⋄").*

of each single low-fidelty simulation as C_L and the cost of each high-fidelity simulation as C_H. The total cost of simulations for MC-based PhIK is $C_H M_L$, while the total cost of simulations for MLMC-based PhIK is $C_H M_H + C_L M_L$. The ratio (MLMC cost/MC cost) is $M_H/M_L + C_L/C_H$. In this specific case, $M_H/M_L = 10/500 = 0.02$, and $C_L/C_H = 1/64 = 0.015625$.

3.2.2. Active learning. We now compare the performance of the active learning algorithm based on Kriging and MLMC-PhIK with ensembles $\{\hat{\boldsymbol{F}}_H^m\}_{m=1}^{10}$ and $\{\hat{\boldsymbol{F}}_L^m\}_{m=1}^{500}$ (i.e., $M_H = 10, M_L = 500$). Because we demonstrated that MLMC-PhIK is more accurate and less costly than MC-PhIK, we do not use the latter in this comparison.

Figures 3.10(a) and (b) show $\hat{s}$ for Kriging, and MLMC-PhIK, both using the initial 15 observations (locations are denoted by squares). Note that $\hat{s}$ in MLMC-PhIK is much smaller than that in Kriging and the locations of local maxima differ. Figure 3.10(c) depicts the standard deviation of concentration σ_{MLMC} computed from MLMC ensembles using Eq. (2.20) (i.e., the standard deviation not conditioned on observations). Figures 3.10(a), (b) and (c) reveals that PhIK has smaller uncertainty than Kriging and the MLMC ensembles.

Next, we use Algorithm 2.1 in combination with Kriging (Figure 3.11) and MLMC-based PhIK (Figure 3.12) to add new observations one by one. In these figures, the initial 15 observation locations are marked as squares and new locations are marked as stars. Figure 3.11 shows the Kriging predictions and corresponding $\boldsymbol{F}_r - \boldsymbol{F}$ error and RMSE obtained via Kriging with 18 and 24 observations. Figure 3.12 presents the same information for MLMC-based PhIK. MLMC-based PhIK consistently outperforms Kriging as quantitatively confirmed by the comparison in Figure 3.13. For



(a) $\hat{s}$ of $\boldsymbol{F}_r$ by Kriging (b) $\hat{s}$ of $\boldsymbol{F}_r$ by MLMC-PhIK (c) Std. of emsembles

FIG. 3.10. *Solute concentration: (a) RMSE of* $\boldsymbol{F}_r$ *by Kriging using* 15 *observations; (b) RMSE of* $\boldsymbol{F}_r$ *by MLMC-based PhIK using* 15 *observations; (c) standard deviation of ensembles estimated by MLMC.*

both methods, the error and uncertainty decrease with an increasing number of observations. However, there are significant differences in the results. In Kriging, most new points are added near the boundary, while in MLMC-based PhIK, new measurements are added inside the domain close to the plume center. This is because the error in Kriging is dominated by the extrapolation error at the boundary. In MLMC-based PhIK, the boundary conditions in the physical model provide sufficient information near the boundaries. Consequently, the active learning algorithm explores more information around the plume. As a result, PhIK achieves higher accuracy than Kriging with a smaller number of observations

4. Conclusion. In this work, we propose the PhIK method, where the mean and covariance function in the GP model are computed from a partially known physical model of the states. We also propose a novel MLMC estimate of the covariance function that, in combination with the standard MLMC estimate of the mean, leads to significant cost reduction in estimating statistics compared to the standard MC method. The resulting statistics in PhIK is non-stationary as can be expected for states of many physical systems due to nonhomogenous initial conditions, boundary conditions, etc. This is different from the standard "data-driven" Kriging, where the mean and kernel are estimated from data only and usually requires an assumption of stationarity. In addition, PhIK avoids the need for estimating hyperparameters in the covariance function, which can be a costly optimization problem.

We prove that PhIK preserves the physical knowledge if it is in the form of a deterministic linear operator. We also provide an upper error bound in the PhIK prediction in the presence of numerical errors. These theoretical results indicate that the accuracy of PhIK prediction depends on the physical model's accuracy ($\|\boldsymbol{y}-\boldsymbol{\mu}\|_2$), numerical error ($\epsilon$) the physical model's stochastic properties, and the selection of observation locations ($\|\mathbf{C}^{-1}\|_2$). We demonstrate that an active learning algorithm in combination with PhIK suggests very different locations for new observations than the data-driven Kriging and results in significantly more accurate predictions with reduced uncertainty. Other Kriging methods, e.g., university Kriging, may perform better than ordinary Kriging. However, such methods require non-stationary mean or kernel with larger numbers of hyperparameters, which adds to the difficulty of the optimization problem in identifying these hyperparameters.

Our method allows model and data convergence without solving complex optimization problems. Moreover, this method is nonintrusive as it can utilize existing domain codes to compute mean and covariance functions for GPR. This differs from other "physics-informed" GPR methods, e.g., [20, 42, 37, 38], where physical laws are used to derive equations for the covariance function, which, in general, must be solved

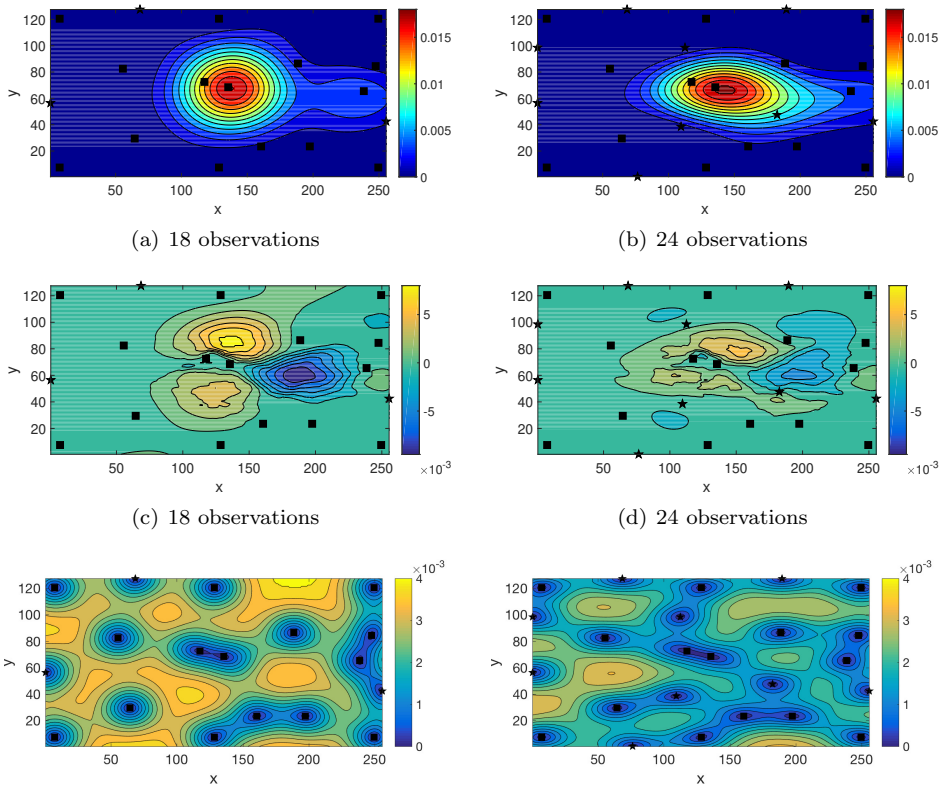
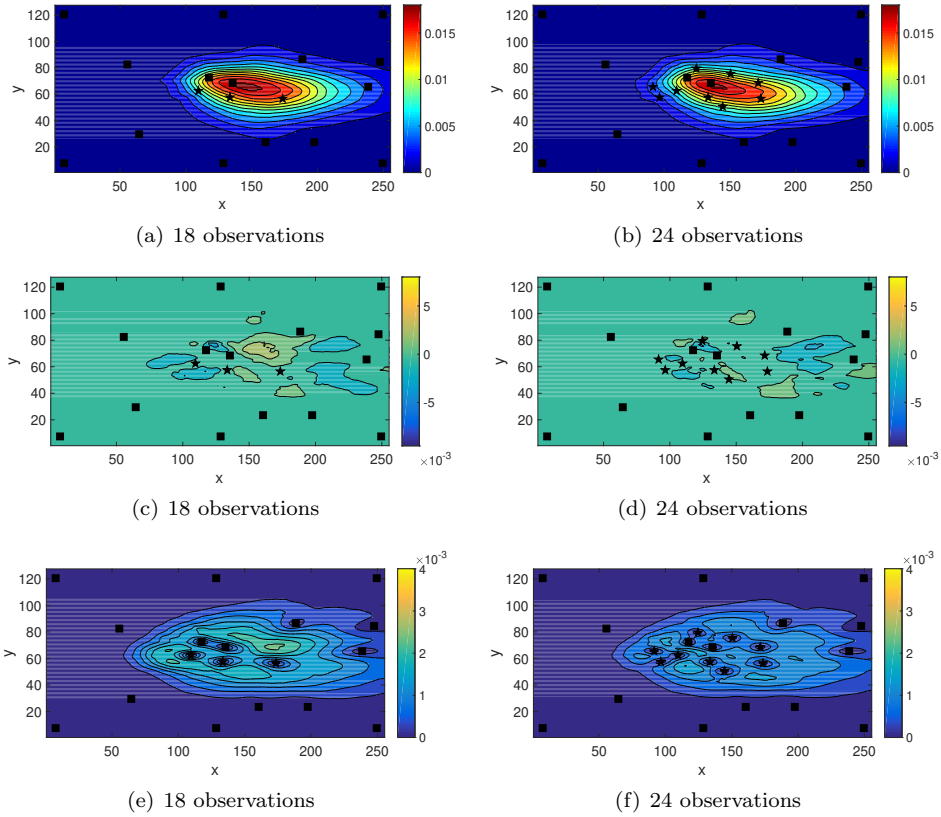


FIG. 3.11. *Reconstruction of the solute concentration by Kriging via active learning. Black squares mark the locations of the original eight observations, and stars are newly added observations. First row: reconstructed field* $\boldsymbol{F}_r$*; second row:* $\boldsymbol{F}_r - \boldsymbol{F}$*; third row:* $\hat{s}$*.*

numerically. PhIK is especially suitable for problems with very costly observations and partially known physics models, including climate, oceanography, hydrology. Such applications are governed by conservation laws, but the parameters, source/sink terms, and stresses in these conservation laws are often unknown and could be modeled as random processes.

We note that merging model and data can be categorized as an multi-resolution modeling (MRM) task, where a real system is represented as a set of models of different resolutions at different abstraction levels from the viewpoint of simulation objectives [12, 21, 36]. In our cases, the data is considered as a high-fidelity "model" and the stochastic model is treated as a low-fidelity model. Moreover, MLMC is a multi-fidelity method that reduces the cost of estimating moments using simulations with different fidelity. Here the fidelity does not only refer to the simulation resolution of a single model (this is what we utilize in the second demonstration example), but also the accuracy of multiple models [33], e.g., some low-fidelity model may neglect less important forces, average out fast processes. Multi-fidelity methods were shown to be efficient for computing one-point statistics (e.g., variance) for quantifying uncertainty [30, 17, 34]. In our work, we use MLMC to reduce the cost of estimating mean and and two-point statistics (i.e., covariance).



(a) 18 observations (b) 24 observations

(c) 18 observations (d) 24 observations

(e) 18 observations (f) 24 observations

FIG. 3.12. *Reconstruction of the solute concentration by MLMC-PhIK via active learning. Black squares mark the locations of the original eight observations, and stars are newly added observations. First row: reconstructed field* $\boldsymbol{F}_r$*; second row:* $\boldsymbol{F}_r - \boldsymbol{F}$*; third row:* $\hat{s}$*.*

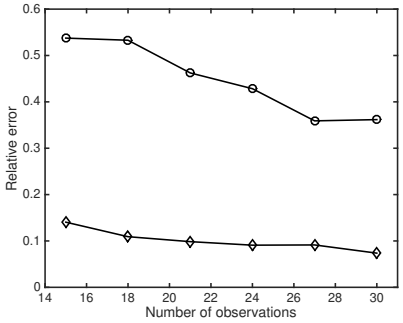


FIG. 3.13. *Relative error of reconstructed solute concentration* $\|\boldsymbol{F}_r - \boldsymbol{F}\|_F / \|\boldsymbol{F}\|_F$ *of Kriging ("∘") and MLMC-based PhIK ("⋄") using different numbers of total observations via active learning.*

Finally, it is worth repeating that the accuracy of PhIK prediction depends on the accuracy of the stochastic physical model. In other words, the PhIK accuracy depends on the distance between the exact solution and the linear space spanned by the simulation ensemble. The accuracy may be improved by adding correction terms, e.g, [53] and the cost of simulations may be further reduced by using other

multi-fidelity approaches [57].

Appendix A. Proof of Theorems 2.1.

Proof. The Kriging prediction Eq. (2.5) can be rewritten as the following function form:

$$\hat{y}(\boldsymbol{x}) = \mu(\boldsymbol{x}) + \sum_{i=1}^{N} a_i k(\boldsymbol{x}, \boldsymbol{x}^{(i)}), \tag{A.1}$$

where $\boldsymbol{x} \in D$, a_i is the i-th entry of $\mathbf{C}^{-1}(\boldsymbol{y} - \boldsymbol{\mu})$. Similarly, the PhIK prediction can be written as

$$\hat{y}(\boldsymbol{x}) = \mu_{MC}(\boldsymbol{x}) + \sum_{i=1}^{N} \tilde{a}_i k_{MC}(\boldsymbol{x}, \boldsymbol{x}^{(i)}), \tag{A.2}$$

where $\tilde{a}_i$ is the i-th entry of $\mathbf{C}_{MC}^{-1}(\boldsymbol{y} - \boldsymbol{\mu}_{MC})$. We have

$$\begin{aligned}\|\mathcal{L}\mu_{MC}(\boldsymbol{x}) - \overline{g(\boldsymbol{x})}\| &= \left\| \frac{1}{M}\sum_{m=1}^{M} \mathcal{L}Y^m(\boldsymbol{x}) - \frac{1}{M}\sum_{m=1}^{M} g(\boldsymbol{x};\omega^m) \right\| \\ &\leq \frac{1}{M}\sum_{m=1}^{M} \|\mathcal{L}Y^m(\boldsymbol{x}) - g(\boldsymbol{x};\omega^m)\| \leq \epsilon.\end{aligned}$$

Also,

(A.3)

$$\begin{aligned}
&\|\mathcal{L}k_{MC}(\boldsymbol{x}, \boldsymbol{x}^{(i)})\| \\
=&\left\| \frac{1}{M-1}\sum_{m=1}^{M} \Big(Y^m(\boldsymbol{x}^{(i)}) - \mu_{MC}(\boldsymbol{x}^{(i)})\Big)\, \mathcal{L}\Big(Y^m(\boldsymbol{x}) - \mu_{MC}(\boldsymbol{x})\Big) \right\| \\
\leq&\frac{1}{M-1}\sum_{m=1}^{M} \Big|Y^m(\boldsymbol{x}^{(i)}) - \mu_{MC}(\boldsymbol{x}^{(i)})\Big| \left\|\mathcal{L}\Big(Y^m(\boldsymbol{x}) - \mu_{MC}(\boldsymbol{x})\Big)\right\| \\
\leq&\frac{1}{M-1}\sum_{m=1}^{M} \Big|Y^m(\boldsymbol{x}^{(i)}) - \mu_{MC}(\boldsymbol{x}^{(i)})\Big| \cdot \\
&\left\{\left\|\mathcal{L}Y^m(\boldsymbol{x}) - g(\boldsymbol{x};\omega^m) - \Big(\mathcal{L}\mu_{MC}(\boldsymbol{x}) - \overline{g(\boldsymbol{x})}\Big)\right\| + \|g(\boldsymbol{x};\omega^m) - \overline{g(\boldsymbol{x})}\|\right\} \\
\leq&\frac{2\epsilon}{M-1}\left(M\sum_{m=1}^{M}\Big|Y^m(\boldsymbol{x}^{(i)}) - \mu_{MC}(\boldsymbol{x}^{(i)})\Big|^2\right)^{\frac{1}{2}} \\
&+\frac{1}{M-1}\left(\sum_{m=1}^{M}\Big|Y^m(\boldsymbol{x}^{(i)}) - \mu_{MC}(\boldsymbol{x}^{(i)})\Big|^2\right)^{\frac{1}{2}}\left(\sum_{m=1}^{M}\|g(\boldsymbol{x};\omega^m) - \overline{g(\boldsymbol{x})}\|^2\right)^{\frac{1}{2}} \\
=&2\epsilon\sqrt{\frac{M}{M-1}}\left(\frac{1}{M-1}\sum_{m=1}^{M}\Big|Y^m(\boldsymbol{x}^{(i)}) - \mu_{MC}(\boldsymbol{x}^{(i)})\Big|^2\right)^{\frac{1}{2}} \\
&+\left(\frac{1}{M-1}\sum_{m=1}^{M}\Big|Y^m(\boldsymbol{x}^{(i)}) - \mu_{MC}(\boldsymbol{x}^{(i)})\Big|^2\right)^{\frac{1}{2}}\left(\frac{1}{M-1}\sum_{m=1}^{M}\|g(\boldsymbol{x};\omega^m) - \overline{g(\boldsymbol{x})}\|^2\right)^{\frac{1}{2}} \\
=&\left(2\epsilon\sqrt{\frac{M}{M-1}} + \sigma(g(\boldsymbol{x};\omega^m))\right)\sigma(Y^m(\boldsymbol{x}^{(i)})).
\end{aligned}$$

Thus, according to Eq. (A.2):

$$\begin{aligned}\|\mathcal{L}\hat{y}(\boldsymbol{x}) - \overline{g(\boldsymbol{x})}\| &\leq \epsilon + \left[2\epsilon\sqrt{\frac{M}{M-1}} + \sigma(g(\boldsymbol{x};\omega^m))\right]\sum_{i=1}^{N}|\tilde{a}_i|\sigma(Y^m(\boldsymbol{x}^i)) \\ &\leq \epsilon + \left[2\epsilon\sqrt{\frac{M}{M-1}} + \sigma(g(\boldsymbol{x};\omega^m))\right]\max_{1\leq i\leq N}|\tilde{a}_i|\sum_{i=1}^{N}\sigma(Y^m(\boldsymbol{x}^i))\end{aligned}$$

Because $\max_i |\tilde{a}_i| = \left\|\mathbf{C}_{MC}^{-1}(\boldsymbol{y} - \boldsymbol{\mu}_{MC})\right\|_\infty$, the conclusion holds. ∎

Appendix B. Proof of Corollary 2.3.

Proof.

$$\begin{aligned}\|\mathcal{L}\overline{Y}^m(\boldsymbol{x})\| &= \|\mathcal{L}Y_H^m(\boldsymbol{x}) - \mathcal{L}Y_L^m(\boldsymbol{x})\| \\ &= \|\mathcal{L}Y_H^m(\boldsymbol{x}) - g(\boldsymbol{x};\omega^m) - (\mathcal{L}Y_L^m(\boldsymbol{x}) - g(\boldsymbol{x};\omega^m))\| \\ &\leq \epsilon_{H} + \epsilon_{L}.\end{aligned}$$

We denote $\mu_{L}(\boldsymbol{x}) = \frac{1}{M_L}\sum_{m=1}^{M_L} Y_L^m(\boldsymbol{x})$, and $\overline{\mu}(\boldsymbol{x}) = \frac{1}{M_H}\sum_{m=1}^{M_H}\overline{Y}^m(\boldsymbol{x})$. According to Eq. (2.17), $\mu_{MLMC}(\boldsymbol{x}) = \mu_{L}(\boldsymbol{x}) + \overline{\mu}(\boldsymbol{x})$. By construction, $\|\mathcal{L}\mu_{L}(\boldsymbol{x}) - \overline{g(\boldsymbol{x})}\| \leq \epsilon_{L}$ and $\|\mathcal{L}\overline{\mu}(\boldsymbol{x})\| \leq \epsilon_{L} + \epsilon_{H}$. Thus,

$$\|\mathcal{L}\mu_{MLMC}(\boldsymbol{x}) - \overline{g(\boldsymbol{x})}\| = \left\|\mathcal{L}\mu_{L}(\boldsymbol{x}) - \overline{g(\boldsymbol{x})} + \mathcal{L}\overline{\mu}(\boldsymbol{x})\right\| \leq 2\epsilon_{L} + \epsilon_{H}.$$

Following the same procedure in Eq. (A.3), we have

$$\begin{aligned}&\left\|\frac{1}{M_L - 1}\sum_{m=1}^{M_L}\left(Y_L^m(\boldsymbol{x}^{(i)}) - \mu_{L}(\boldsymbol{x}^{(i)})\right)\mathcal{L}\left(Y_L^m(\boldsymbol{x}) - \mu_{L}(\boldsymbol{x})\right)\right\| \\ &\qquad\leq \left(2\epsilon_{L}\sqrt{\frac{M_L}{M_L-1}} + \sigma(g(\boldsymbol{x};\omega^m)\right)\sigma(Y_L^m(\boldsymbol{x}^{(i)})),\end{aligned}$$

and

$$\begin{aligned}&\left\|\frac{1}{M_H - 1}\sum_{m=1}^{M_H}\left(\overline{Y}^m(\boldsymbol{x}^{(i)}) - \overline{\mu}(\boldsymbol{x}^{(i)})\right)\mathcal{L}\left[\overline{Y}^m(\boldsymbol{x}) - \overline{\mu}(\boldsymbol{x})\right]\right\| \\ &\qquad\leq 2(\epsilon_{H} + \epsilon_{L})\sqrt{\frac{M_H}{M_H-1}}\sigma(\overline{Y}^m(\boldsymbol{x}^{(i)})).\end{aligned}$$

As such,

$$
\begin{aligned}
&\|\mathcal{L}\hat{y}(\boldsymbol{x}) - \overline{g(\boldsymbol{x})}\| \\
\leq &\epsilon_{_H} + 2\epsilon_{_L} + \left(2\epsilon_{_L}\sqrt{\frac{M_L}{M_L-1}} + \sigma(g(\boldsymbol{x};\omega^m))\right)\sum_{i=1}^{N}\tilde{a}_i\sigma(Y_L^m(\boldsymbol{x}^{(i)})) \\
&+ 2(\epsilon_{_H}+\epsilon_{_L})\sum_{i=1}^{N}\tilde{a}_i\sqrt{\frac{M_H}{M_H-1}}\sigma(\overline{Y}^m(\boldsymbol{x}^{(i)})) \\
= &\epsilon_{_H}\left(1 + 2\sum_{i=1}^{N}\tilde{a}_i\sqrt{\frac{M_H}{M_H-1}}\sigma(\overline{Y}^m(\boldsymbol{x}^{(i)}))\right) \\
&+ \epsilon_{_L}\left[2 + 2\sum_{i=1}^{N}\tilde{a}_i\left(\sqrt{\frac{M_L}{M_L-1}}\sigma(Y_L^m(\boldsymbol{x}^{(i)})) + \sqrt{\frac{M_H}{M_H-1}}\sigma(\overline{Y}^m(\boldsymbol{x}^{(i)}))\right)\right] \\
&+ \sigma(g(\boldsymbol{x};\omega^m))\sum_{i=1}^{N}\tilde{a}_i\sigma(Y_L^m(\boldsymbol{x}^{(i)})).
\end{aligned}
$$

The bound of $\tilde{a}_i$ is given in Corollary 2.1 by replacing $\mathbf{C}_{_{MC}}$ with $\mathbf{C}_{_{MLMC}}$. ∎

REFERENCES

[1] P. Abrahamsen, *A review of Gaussian random fields and correlation functions*, 1997.

[2] M. Armstrong, *Problems with universal kriging*, Journal of the International Association for Mathematical Geology, 16 (1984), pp. 101–108.

[3] A. Barth, C. Schwab, and N. Zollinger, *Multi-level Monte Carlo finite element method for elliptic PDEs with stochastic coefficients*, Numerische Mathematik, 119 (2011), pp. 123–161.

[4] A. Berlinet and C. Thomas-Agnan, *Reproducing kernel Hilbert spaces in probability and statistics*, Springer Science & Business Media, 2011.

[5] C. Bierig and A. Chernov, *Convergence analysis of multilevel Monte Carlo variance estimators and application for random obstacle problems*, Numerische Mathematik, 130 (2015), pp. 579–613.

[6] C. Bierig and A. Chernov, *Estimation of arbitrary order central statistical moments by the multilevel Monte Carlo method*, Stochastics and Partial Differential Equations Analysis and Computations, 4 (2016), pp. 3–40.

[7] S. Brahim-Belhouari and A. Bermak, *Gaussian process for nonstationary time series prediction*, Computational Statistics & Data Analysis, 47 (2004), pp. 705–712.

[8] K. A. Cliffe, M. B. Giles, R. Scheichl, and A. L. Teckentrup, *Multilevel Monte Carlo methods and applications to elliptic PDEs with random coefficients*, Computing and Visualization in Science, 14 (2011), p. 3.

[9] D. A. Cohn, Z. Ghahramani, and M. I. Jordan, *Active learning with statistical models*, Journal of Artificial Intelligence Research, 4 (1996), pp. 129–145.

[10] T. Collet and O. Pietquin, *Optimism in active learning with Gaussian processes*, in International Conference on Neural Information Processing, Springer, 2015, pp. 152–160.

[11] H. Dai, X. Chen, M. Ye, X. Song, and J. M. Zachara, *A geostatistics-informed hierarchical sensitivity analysis method for complex groundwater flow and transport modeling*, Water Resources Research, 53 (2017), pp. 4327–4343.

[12] P. K. Davis and J. H. Bigelow, *Experiments in multiresolution modeling (mrm)*, tech. report, RAND CORP SANTA MONICA CA, 1998.

[13] C. V. Deutsch and A. G. Journel, *GSLIB: Geostatistical Software Library and User's Guide*, Oxford University Press, 1992.

[14] S. Emmanuel and B. Berkowitz, *Mixing-induced precipitation and porosity evolution in porous media*, Advances in Water Resources, 28 (2005), pp. 337–344.

[15] G. Evensen, *The ensemble Kalman filter: Theoretical formulation and practical implementation*, Ocean Dynamics, 53 (2003), pp. 343–367.

[16] A. Forrester, A. Keane, et al., *Engineering Design via Surrogate Modelling: A Practical Guide*, John Wiley & Sons, 2008.
[17] G. Geraci, M. S. Eldred, and G. Iaccarino, *A multifidelity multilevel monte carlo method for uncertainty propagation in aerospace applications*, in 19th AIAA Non-Deterministic Approaches Conference, 2017, p. 1951.
[18] M. Giles, *Improved multilevel Monte Carlo convergence using the Milstein scheme*, in Monte Carlo and quasi-Monte Carlo Methods 2006, Springer, 2008, pp. 343–358.
[19] M. B. Giles, *Multilevel Monte Carlo path simulation*, Operations Research, 56 (2008), pp. 607–617.
[20] P. Hennig, M. A. Osborne, and M. Girolami, *Probabilistic numerics and uncertainty in computations*, Proceedings of the Royal Society London A, 471 (2015), p. 20150142.
[21] S.-Y. Hong and T. G. Kim, *Specification of multi-resolution modeling space for multi-resolution system simulation*, Simulation, 89 (2013), pp. 28–40.
[22] Z. Hou, M. Huang, L. R. Leung, G. Lin, and D. M. Ricciuto, *Sensitivity of surface flux simulations to hydrologic parameters based on an uncertainty quantification framework applied to the community land model*, Journal of Geophysical Research: Atmospheres, 117 (2012).
[23] K. D. Jarman and A. M. Tartakovsky, *A comparison of closures for stochastic advection-diffusion equations*, SIAM/ASA Journal on Uncertainty Quantification, 1 (2013), pp. 319–347.
[24] D. R. Jones, M. Schonlau, and W. J. Welch, *Efficient global optimization of expensive black-box functions*, Journal of Global optimization, 13 (1998), pp. 455–492.
[25] P. K. Kitanidis, *Introduction to Geostatistics: Applications in Hydrogeology*, Cambridge University Press, 1997.
[26] A. Krause, A. Singh, and C. Guestrin, *Near-optimal sensor placements in Gaussian processes: Theory, efficient algorithms and empirical studies*, J. Mach. Learn. Res., 9 (2008), pp. 235–284.
[27] G. Lin and A. M. Tartakovsky, *An efficient, high-order probabilistic collocation method on sparse grids for three-dimensional flow and solute transport in randomly heterogeneous porous media*, Advances in Water Resources, 32 (2009), pp. 712–722.
[28] C. A. Micchelli and G. Wahba, *Design problems for optimal surface interpolation.*, tech. report, Wisconsin Univ-Madison Dept of Statistics, 1979.
[29] J. M. Murphy, D. M. Sexton, D. N. Barnett, G. S. Jones, M. J. Webb, M. Collins, and D. A. Stainforth, *Quantification of modelling uncertainties in a large ensemble of climate change simulations*, Nature, 430 (2004), p. 768.
[30] L. W.-T. Ng and M. Eldred, *Multifidelity uncertainty quantification using non-intrusive polynomial chaos and stochastic collocation*, in 53rd AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference 20th AIAA/ASME/AHS Adaptive Structures Conference 14th AIAA, 2012, p. 1852.
[31] H. Niederreiter, *Random Number Generation and Quasi-Monte Carlo Methods*, vol. 63, SIAM, 1992.
[32] C. J. Paciorek and M. J. Schervish, *Nonstationary covariance functions for Gaussian process regression*, in Advances in Neural Information Processing Systems, 2004, pp. 273–280.
[33] B. Peherstorfer, K. Willcox, and M. Gunzburger, *Survey of multifidelity methods in uncertainty propagation, inference, and optimization*, (2016).
[34] B. Peherstorfer, K. Willcox, and M. Gunzburger, *Survey of multifidelity methods in uncertainty propagation, inference, and optimization*, Siam Review, 60 (2018), pp. 550–591.
[35] C. Plagemann, K. Kersting, and W. Burgard, *Nonstationary Gaussian process regression using point estimates of local smoothness*, in Joint European Conference on Machine Learning and Knowledge Discovery in Databases, Springer, 2008, pp. 204–219.
[36] L. Rabelo, K. Kim, T. W. Park, J. Pastrana, M. Marin, G. Lee, K. Nagadi, B. Ibrahim, and E. Gutierrez, *Multi resolution modeling*, in Proceedings of the 2015 Winter Simulation Conference, WSC '15, IEEE Press, 2015, pp. 2523–2534, http://dl.acm.org/citation.cfm?id=2888619.2888909.
[37] M. Raissi, P. Perdikaris, and G. E. Karniadakis, *Machine learning of linear differential equations using Gaussian processes*, Journal of Computational Physics, 348 (2017), pp. 683–693.
[38] M. Raissi, P. Perdikaris, and G. E. Karniadakis, *Numerical Gaussian processes for time-dependent and nonlinear partial differential equations*, SIAM Journal on Scientific Computing, 40 (2018), pp. A172–A198.
[39] C. E. Rasmussen, *Gaussian processes in machine learning*, in Advanced Lectures on Machine

Learning, Springer, 2004, pp. 63–71.
[40] K. Ritter, *Average-case Analysis of Numerical Problems*, Springer, 2007.
[41] J. Sacks, W. J. Welch, T. J. Mitchell, and H. P. Wynn, *Design and analysis of computer experiments*, Statistical Science, (1989), pp. 409–423.
[42] M. Schober, D. K. Duvenaud, and P. Hennig, *Probabilistic ODE solvers with Runge-Kutta means*, in Advances in Neural Information Processing Systems, 2014, pp. 739–747.
[43] M. L. Stein, *Interpolation of Spatial Data: Some Theory for Kriging*, Springer Science & Business Media, 2012.
[44] A. M. Tartakovsky, S. P. Neuman, and R. J. Lenhard, *Immiscible front evolution in randomly heterogeneous porous media*, Physics of Fluids (1994-present), 15 (2003), pp. 3331–3341.
[45] A. M. Tartakovsky, M. Panzeri, G. D. Tartakovsky, and A. Guadagnini, *Uncertainty quantification in scale-dependent models of flow in porous media*, Water Resources Research, 53 (2017), pp. 9392–9401.
[46] S. Tong and D. Koller, *Support vector machine active learning with applications to text classification*, Journal of Machine Learning Research, 2 (2001), pp. 45–66.
[47] M. D. White and M. Oostrom, *STOMP subsurface transport over multiple phases, version 4.0, users guide*, tech. report, PNNL-15782, Richland, WA, 2006.
[48] C. K. Williams and C. E. Rasmussen, *Gaussian processes for machine learning*, the MIT Press, 2 (2006), p. 4.
[49] C. K. Williams and F. Vivarelli, *Upper and lower bounds on the learning curve for Gaussian processes*, Machine Learning, 40 (2000), pp. 77–102.
[50] J. Witteveen, K. Duraisamy, and G. Iaccarino, *Uncertainty quantification and error estimation in scramjet simulation*, in 17th AIAA International Space Planes and Hypersonic Systems and Technologies Conference, 2011, p. 2283.
[51] D. Xiu and J. S. Hesthaven, *High-order collocation methods for differential equations with random inputs*, SIAM Journal on Scientific Computing, 27 (2005), pp. 1118–1139.
[52] B. Yang, Y. Qian, G. Lin, L. R. Leung, P. J. Rasch, G. J. Zhang, S. A. McFarlane, C. Zhao, Y. Zhang, H. Wang, et al., *Uncertainty quantification and parameter tuning in the cam5 zhang-mcfarlane convection scheme and impact of improved convection on the global circulation and climate*, Journal of Geophysical Research: Atmospheres, 118 (2013), pp. 395–415.
[53] X. Yang, D. Barajas-Solano, G. Tartakovsky, and A. M. Tartakovsky, *Physics-informed cokriging: A gaussian-process-regression-based multifidelity method for data-model convergence*, Journal of Computational Physics, 395 (2019), pp. 410–431.
[54] X. Yang, M. Choi, G. Lin, and G. E. Karniadakis, *Adaptive ANOVA decomposition of stochastic incompressible and compressible flows*, Journal of Computational Physics, 231 (2012), pp. 1587–1614.
[55] X. Yang and G. E. Karniadakis, *Reweighted ℓ_1 minimization method for stochastic elliptic differential equations*, Journal of Computational Physics, 248 (2013), pp. 87–108.
[56] X. Yang, D. Venturi, C. Chen, C. Chryssostomidis, and G. E. Karniadakis, *EOF-based constrained sensor placement and field reconstruction from noisy ocean measurements: Application to Nantucket Sound*, Journal of Geophysical Research: Oceans, 115 (2010), p. C12072.
[57] X. Yang, X. Zhu, and J. Li, *When bifidelity meets cokriging: An efficient physics-informed multifidelity method*, arXiv preprint arXiv:1812.02919, (2018).
[58] D. Ylvisaker, *Designs on random fields*, A Survey of Statistical Design and Linear Models, 37 (1975), pp. 593–607.
[59] M. Yuan, T. T. Cai, et al., *A reproducing kernel hilbert space approach to functional linear regression*, The Annals of Statistics, 38 (2010), pp. 3412–3444.
[60] Y. Zhang and J. G. Bellingham, *An efficient method of selecting ocean observing locations for capturing the leading modes and reconstructing the full field*, Journal of Geophysical Research: Oceans, 113 (2008), p. C04005.
[61] Z. Zhang, X. Yang, I. V. Oseledets, G. E. Karniadakis, and L. Daniel, *Enabling high-dimensional hierarchical uncertainty quantification by anova and tensor-train decomposition*, IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 34 (2014), pp. 63–76.
[62] X. Zhu, E. M. Linebarger, and D. Xiu, *Multi-fidelity stochastic collocation method for computation of statistical moments*, Journal of Computational Physics, 341 (2017), pp. 386–396.