ORNL REPORT

ORNL/TM-2012/499
Unlimited Release
Printed October 2012

An adaptive sparse-grid high-order stochastic collocation method for Bayesian inference in groundwater reactive transport modeling

G. Zhang, D. Lu, M. Ye, M. Gunzburger and C. G. Webster

Prepared by
Oak Ridge National Laboratory
One Bethel Valley Road, Oak Ridge, Tennessee 37831

The Oak Ridge National Laboratory is operated by UT-Battelle, LLC,
for the United States Department of Energy under Contract DE-AC05-00OR22725.

ORNL/TM-2012/499

Computer Science and Mathematics Division

AN ADAPTIVE SPARSE-GRID HIGH-ORDER STOCHASTIC COLLOCATION METHOD FOR BAYESIAN INFERENCE IN GROUNDWATER REACTIVE TRANSPORT MODELING

G. Zhang [*] D. Lu [†] M. Ye [‡] M. Gunzburger [§] C. G. Webster [¶]

Date Published: October 2012

Prepared by
OAK RIDGE NATIONAL LABORATORY
Oak Ridge, Tennessee 37831-6283
managed by
UT-BATTELLE, LLC
for the
U.S. DEPARTMENT OF ENERGY
under contract DE-AC05-00OR22725

[*]Computer Science and Mathematics Division, Oak Ridge National Laboratory, One Bethel Valley Road, P.O. Box 2008, MS-6367, Oak Ridge, TN 37831-6367 (zhangg@ornl.gov)

[†]Department of Scientific Computing, 400 Dirac Science Library, Florida State University, Tallahassee, FL 32306–4120 (dl07f@my.fsu.edu)

[‡]Department of Scientific Computing, 400 Dirac Science Library, Florida State University, Tallahassee, FL 32306–4120 (mye@fsu.edu)

[§]Department of Scientific Computing, 400 Dirac Science Library, Florida State University, Tallahassee, FL 32306–4120 (mgunzburger@fsu.edu)

[¶]Computer Science and Mathematics Division, Oak Ridge National Laboratory, One Bethel Valley Road, P.O. Box 2008, MS-6164, Oak Ridge, TN 37831-6164 (webstercg@ornl.gov)

CONTENTS

LIST OF FIGURES v

LIST OF TABLES vi

ABSTRACT 1

ACKNOWLEDGEMENTS 1

1 INTRODUCTION 2

2 PROBLEM DEFINITION 5

3 METHODOLOGY 6
3.1 Global optimization for determining the prior domain 6
3.2 Adaptive sparse-grid high-order stochastic collocation method 7
3.3 Algorithm for constructing the surrogate PPDF 15

4 APPLICATION IN GROUNDWATER REACTIVE TRANSPORT MODELING ... 18
4.1 Case 1: Multi-species reactive transport equations 18
4.2 Case 2: Reactive transport of uranium (VI) in column experiment 24

5 CONCLUSION 30

Appendix A BUILDING THE SURROGATE SYSTEM FOR A 2-D FUNCTION WITH MULTIPLE SIGNIFICANT MODES 31

LIST OF FIGURES

1 One dimensional 3-level hierarchical bases: linear basis (left), quadratic basis (middle), and cubic basis (right). 10

2 A comparison of a 3-level isotropic sparse grid (b) and the corresponding full tensor-product grid (c) based on Newton-Cotes points. The sparse grid (b) consists of 10 coarse sub-grids above the dashed line in (a)(grids shown in black), each of which is a coarse tensor-product grid with $i_1 + i_2 \leq 3$. The sparse grid has only 29 points. The full tensor-product grid (c), constructed by the all 16 sub-grids in (a) (grids shown in both black and gray) with $\max(i_1, i_2) \leq 3$, has 81 points. . . 12

3 A 6-level adaptive sparse grid for interpolating a one-dimensional function $\eta(\theta) = \exp[-(\theta - 0.4)^2/0.0625^2]$ on $[0, 1]$ with the error tolerance of 0.01. The resulting adaptive sparse grid has only 21 points (black points) whereas the full grid has 65 points (black and gray points). 14

4 The algorithm for constructing the surrogate system of the PPDF. 17

5 The serial-parallel reaction network in Case 1. 19

6 The marginal PPDFs for the six unknown parameters in Case 1 are estimated using the conventional MCMC (C-MCMC) with 60,000 model executions (red-solid lines), the linear, quadratic, and cubic surrogate systems with 9226, 4375, 3765 model executions (dashed lines), respectively. The true parameter values are plotted in black-solid lines. Take the conventional MCMC results as reference, the estimations by the surrogate systems are accurate enough but with computational cost greatly reduced. 21

7 The marginal PPDFs for the six unknown parameters in Case 1 are estimated using the conventional MCMC (C-MCMC) with 60,000 model executions (red-solid lines) and 9226, 4375, 3765 model executions (dashed lines) which correspond to those used by the linear, quadratic, and cubic surrogate systems,respectively in Figure 6. The true parameter values are plotted in black-solid lines. 22

8 The errors of the surrogate systems based on linear, quadratic, and cubic hierarchical basis functions in Case 1. 23

9 Left graph shows the true solution of $c_3(x, t)$ at $t = 40$ (black dots), the 95% credible intervals estimated using conventional MCMC (red-solid lines) and using the surrogate systems with linear, quadratic, and cubic hierarchical basis (dashed lines) in Case 1. Right graph shows the PDFs of a specific predicted quantity of $c_3(x = 32, t = 40)$ estimated using the conventional MCMC (red-solid line),and the linear, quadratic, and cubic surrogate systems (dashed lines) in Case 1. The true solution is plotted in black-solid line and the red-solid lines are taken as reference. The surrogate-based MCMC simulations require 1853, 1032, 793 model executions for the linear, quadratic and cubic surrogate system respectively, while the conventional MCMC requires 57,000 model executions 25

10 (a): The 8-level isotropic sparse grid with 6017 points; (b): the adaptive sparse grid with 1072 points for the linear surrogate system; (c): the adaptive sparse grid with 705 points for the quadratic surrogate system; and (d): the adaptive sparse grid with 341 points for the cubic surrogate system in Case 2. 27

11 Left graph shows the true breakthrough curve of Experiment 4 in Case 2 (black dots), the 95% credible intervals estimated using the conventional MCMC (red-solid lines) and using the cubic surrogate system (pink-dashed lines). Right graph shows the PDFs of a specific predicted quantity at Pore Volume of 2.49 estimated using the conventional MCMC (red-solid line) with 58,200 model executions and the cubic surrogate system (pink-dashed line) with 411 model executions. The true solution is plotted in black-solid line and the red-solid lines are taken as reference. 28
12 The 1D and 2D marginal PPDFs of the three unknown parameters in Case 2 are estimated using the conventional MCMC with 60,000 model executions (red-solid lines and contours) and the cubic surrogate system with 6748 model executions (pink-dashed lines and contours). The true parameter values are plotted in black-solid lines or black dots. Take the conventional MCMC results as reference, the estimations by the cubic surrogate system is accurate enough but with computational cost greatly reduced. 29
A1 (a): Surface of the target function $\eta(\boldsymbol{\theta})$ where modes $M_3(\boldsymbol{\theta})$ and $M_4(\boldsymbol{\theta})$ are too small to be displayed. (b): Surface of $\log(\eta(\boldsymbol{\theta}))$ that has four modes displayed. (c): Plain view of $\log(\eta(\boldsymbol{\theta}))$. Since modes $M_1(\boldsymbol{\theta})$ and $M_2(\boldsymbol{\theta})$ (black dots) are more significant than $M_3(\boldsymbol{\theta})$ and $M_4(\boldsymbol{\theta})$ (green dots), our algorithm only build the surrogate system consisting of the two components for $M_1(\boldsymbol{\theta})$ and $M_2(\boldsymbol{\theta})$. The prior domains of the two components are marked by the black boxes around the two black dots. (d): The sparse grids for the two components of the surrogate system. The one for $M_1(\boldsymbol{\theta})$ has 341 points and the other for $M_2(\boldsymbol{\theta})$ has 145 points.. 34

LIST OF TABLES

1 True parameter values used in Case 1. 19
2 The true values, the initial searching domain Γ and the prior domain $\Gamma_{\widehat{\boldsymbol{\theta}}_1}$ of the six unknown parameters in Case 1. 20
3 The 95% credible intervals of the six unknown parameters obtained using the conventional MCMC and the surrogate systems with linear, quadratic, and cubic basis functions in Case 1. 24
4 Surface complexation reactions and parameters of the true model in Case 2. Total site density used in this model is 1.3 M/L and the summation of site fraction in the last column is 1.0. 25
5 The true values, the initial searching domain Γ and the prior domain $\Gamma_{\widehat{\boldsymbol{\theta}}_1}$ of the three unknown parameters in Case 2. 26

ABSTRACT

Although Bayesian analysis has become vital to the quantification of prediction uncertainty in groundwater modeling, its application has been hindered due to the computational cost associated with numerous model executions needed for exploring the posterior probability density function (PPDF) of model parameters. This is particularly the case when the PPDF is estimated using Markov Chain Monte Carlo (MCMC) sampling. In this study, we develop a new approach that improves computational efficiency of Bayesian inference by constructing a surrogate system based on an adaptive sparse-grid high-order stochastic collocation (aSG-hSC) method. Unlike previous works using first-order hierarchical basis, we utilize a compactly supported higher-order hierarchical basis to construct the surrogate system, resulting in a significant reduction in the number of computational simulations required. In addition, we use hierarchical surplus as an error indicator to determine adaptive sparse grids. This allows local refinement in the uncertain domain and/or anisotropic detection with respect to the random model parameters, which further improves computational efficiency. Finally, we incorporate a global optimization technique and propose an iterative algorithm for building the surrogate system for the PPDF with multiple significant modes. Once the surrogate system is determined, the PPDF can be evaluated by sampling the surrogate system directly with very little computational cost. The developed method is evaluated first using a simple analytical density function with multiple modes and then using two synthetic groundwater reactive transport models. The groundwater models represent different levels of complexity; the first example involves coupled linear reactions and the second example simulates nonlinear uranium surface complexation. The results show that the aSG-hSC is an effective and efficient tool for Bayesian inference in groundwater modeling in comparison with conventional MCMC simulations. The computational efficiency is expected to be more beneficial to more computational expensive groundwater problems.

ACKNOWLEDGEMENTS

The first author was supported by the US Air Force Office of Scientific Research under grant number FA9550-11-1-0149. The first author was also supported by the Advanced Simulation Computing Research (ASCR), Department of Energy, through the Householder Fellowship at ORNL. The second author was supported by DOE-SBR grant DE-SC0002687. The third author was supported by NSF-EAR grant 0911074, DOE-SBR grant DE-SC0002687, ORAU/ORAL High Performance Computing Grant. The fourth author was supported by the US Air Force Office of Scientific Research under grant number FA9550-11-1-0149. The fifth author was supported by the US Air Force Office of Scientific Research under grant #1854-V521-12. The fifth author was also sponsored by the Director's Strategic Hire Funds through the Laboratory Directed Research and Development (LDRD) Program of Oak Ridge National Laboratory (ORNL). The ORNL is operated by UT-Battelle, LLC, for the United States Department of Energy under Contract DE-AC05-00OR22725.

1 INTRODUCTION

Groundwater models are vital tools for predicting the effects of future anthropomorphic and/or natural occurrences in the subsurface environment. Because it is essential for dealing with realistic experimental data and assessing the reliability of predictions based on numerical simulations, the development of novel uncertainty quantification (UQ) methodologies are necessary to facilitate science-informed decision-making for water resource management. A crucial, yet often complicated, ingredient that all UQ approaches must i3corporate is a proper description of the uncertainty in the system parameters, in the external environment, and in predictions. The Bayesian inference is one of the more widely utilized approaches for quantifying such uncertainty [1, 3, 4, 8, 20–23, 33, 34], wherein the uncertainty in the model parameters and predictions are modeled as random variables. The Bayesian method can incorporate any type of prior information and measurement errors in the posterior probability density function (PPDF), rendering the Bayesian inference more informative, and therefore, more valuable for risk analysis and decision processes.

To quantify the prediction uncertainty using the Bayesian method, one needs to simulate the PPDF of model parameters. However, except in special cases in which the analytical expression can be derived, the PPDF is usually known up to a multiplicative constant and needs to be estimated numerically using sampling techniques. The probability density function (PDF) of model predictions are typically estimated by virtue of the the following numerical procedure: (1) sample realizations of the uncertain parameters from their PPDF, (2) for each set of sampled parameters, evaluate the model prediction and (3) estimate the PDF of the model predictions by, e.g., constructing histograms from the ensemble of simulations. For the first step, the Markov Chain Monte Carlo (MCMC) method has proven to be a powerful approach to generate parameter samples from the PPDF [9,15,21,31,32]. This approach is based on constructing Markov chains that have the desired PPDF as its limiting distribution. The states of the chains after a large number of steps are then used as samples of the PPDF. Due to the non-intrusive implementation of the MCMC approach, it remains very popular approach for calculation ensembles of random solutions to groundwater problems. Many algorithms have been developed to improve the computational efficiency, such as delayed rejection and adaptive Metropolis (DRAM) sampling [13] and differential evolution adaptive Metropolis (DREAM) sampling [31, 32]. These methods focus on reducing the needed number of model executions required by the MCMC method by either increasing the acceptance rate or decreasing the number of samples in the burn-in period before convergence.

However, even with a high acceptance rate and a short burn-in period, the MCMC approach still requires a very large number of parameter samples to achieve a prescribed accuracy for the estimation of the PPDF due to the slow convergence rate of the Monte Carlo sampling. The total computational burden is not significantly reduced because one model execution is still needed to generate one parameter sample. Furthermore, since most groundwater problems are described by partial differential equations (PDEs), estimating the PPDF and generating the ensemble of predictions with MCMC require a substantial number of numerical approximations of the PDEs, i.e. forward model executions. As a consequence, the MCMC approach is very computationally expensive, especially for problems of modeling groundwater reactive transport [35]. We mention here that the

number of forward model executions practically makes up the total computational cost because all other MCMC calculations are simple algebraic operations and negligible when compared to the cost of the forward model execution, e.g. solving PDEs.

Recently, surrogate modeling has been used to improve computational efficiency for model calibration and optimization [29]. The sparse-grid stochastic collocation (SG-SC) method [24, 25] has received considerable attention due to its efficiency in uncertainty quantification (UQ), especially for problems with a moderately large number of model parameters. The SG-SC method was used to quantify forward uncertainty propagation in subsurface transport modeling [18, 19]. In [34] and [20], the SG-SC method was used to build surrogate systems for geophysical models, after which the surrogate systems were employed to conduct Bayesian inference. However, the surrogate systems used in these efforts were constructed using linear (first-order) hierarchical polynomial functions, and increased number of interpolation points, i.e., more model executions, are needed to obtain the prescribed interpolation accuracy. On the other hand, the surface of the PPDF may have multiple significant modes where the significance of each mode is measured by the value of the PPDF at its peak, i.e., the height of its peak. In this case, existing algorithms cannot succeed in capturing all the significant modes or may succeed only with significantly increased computational effort.

In this paper, a new approach is proposed to improve the efficiency of Bayesian inference using an adaptive sparse-grid high-order stochastic collocation (aSG-hSC) method. The general idea is to construct a surrogate system with multiple components for the PPDF, each of which is a surrogate model based on the aSG-hSC method to approximate one significant mode of the PPDF. Subsequently, the surrogate system is employed to replace the true PPDF in the MCMC simulation. Our method is advantageous for several reasons. First, we construct the surrogate system with a sparse-grid interpolation with high-order hierarchical polynomial basis, as in [5, 11], using e.g., quadratic or cubic functions. Due to their increased accuracy compared to the linear hierarchical basis, the number of model executions needed for constructing the surrogate system can be greatly reduced. This high-order approach is not a trivial extension from the linear hierarchical technique and, to our knowledge, this is the first attempt to introduce the high-order hierarchical basis (coupled with the SG-SC approach) into the Bayesian framework. Second, instead of building the approximate PPDF using isotropic sparse-grid interpolation [2, 25] or dimension-adaptive sparse-grid interpolation [24], we use a locally adaptive sparse-grid interpolation [11]. This technique utilizes the hierarchical surplus (discussed in Section 3.2) as an error indicator to detect the non-smooth and/or important regions in the parameter space and adaptively place more points in this area. This results in further computational gains and guarantees that a user-defined accuracy of the surrogate system is realized.

In addition, we also incorporate global optimization into the surrogate system and propose a new iterative algorithm to further reduce the computational cost and construct the surrogate system for a PPDF with multiple significant modes. It is well-known that in conventional MCMC methods the searching domain is usually defined to be relatively large due to the lack of information concerning the locations of the significant modes of the PPDF. Markov chains will move toward the the high-probability region of each significant mode by following the Metroplis-Hastings rule [9]. However, if we construct a surrogate system over this arbitrarily large searching domain, a

significant number of sparse grid points will be placed in the low-probability region of the PPDF as it tries to detect each significant mode. This may result in the computational cost for the surrogate system that may even exceed that of the conventional MCMC method, which in turn, renders the surrogate system useless. To overcome this challenge, our iterative algorithm can capture all the significant modes and build a surrogate for each of them. In each iteration, we utilize global optimization to detect the maximum of one significant mode of the PPDF and determine a bounded prior domain for the significant mode by estimating the Hessian matrix of the negative logarithm of the PPDF at the detected maximum. In this effort, we use the global optimization algorithm DIRECT, first proposed in [14], which is an improvement of the standard Lipschitzian approach that eliminates the need to specify a Lipschitz constant. We demonstrate that our algorithm can find all the modes whose significance is larger than a user-defined significance tolerance, and the resulting prior domain for each significant mode determined in this way is much smaller than the searching domain for the MCMC method.

The outline of this paper is as follows. In Section 2, we briefly introduce the Bayesian framework and the MCMC algorithm used in this study. In Section 3, we introduce the global optimization method and propose our aSG-hSC method for constructing the surrogate system. In Section 4, we apply our new approach to reactive transport problems and demonstrate its effectiveness and efficiency when compared to the conventional MCMC method.

2 PROBLEM DEFINITION

We consider the Bayesian inference problem for the general physical model

$$\boldsymbol{d} = \boldsymbol{f}(\boldsymbol{\theta}) + \boldsymbol{\varepsilon}, \tag{2.1}$$

where $\boldsymbol{d} = (d_1, \ldots, d_{N_d})$ is a vector of N_d measurement data, $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_{N_\theta})$ is a vector of N_θ model parameters, $\boldsymbol{f}(\boldsymbol{\theta})$ is the quantity of interest (QoI) of the forward model, e.g., discretized PDE solution, with N_θ inputs and N_d outputs and $\boldsymbol{\varepsilon}$ is the random measurement error. Here, we assume that $\boldsymbol{\varepsilon}$ satisfies the multivariate Gaussian distribution with zero mean and known covariance matrix $\boldsymbol{\Sigma}$, i.e., $\boldsymbol{\varepsilon} \sim N(\boldsymbol{0}, \boldsymbol{\Sigma})$. The PPDF $p(\boldsymbol{\theta}|\boldsymbol{d})$ of the model parameters $\boldsymbol{\theta}$, given the data $\boldsymbol{d}$, is defined with the aid of Bayes' theorem [4] as

$$p(\boldsymbol{\theta}|\boldsymbol{d}) = \frac{p(\boldsymbol{d}|\boldsymbol{\theta})p(\boldsymbol{\theta})}{\displaystyle\int p(\boldsymbol{d}|\boldsymbol{\theta})p(\boldsymbol{\theta})d\boldsymbol{\theta}}, \tag{2.2}$$

where $p(\boldsymbol{\theta})$ is the prior distribution and $p(\boldsymbol{d}|\boldsymbol{\theta})$ is the likelihood function given by

$$\begin{aligned} p(\boldsymbol{d}|\boldsymbol{\theta}) =& \frac{1}{(2\pi)^{N_d/2}|\boldsymbol{\Sigma}|^{1/2}} \\ & \cdot \exp\Big[-\frac{1}{2}(\boldsymbol{d} - \boldsymbol{f}(\boldsymbol{\theta}))^\top \boldsymbol{\Sigma}^{-1}(\boldsymbol{d} - \boldsymbol{f}(\boldsymbol{\theta}))\Big], \end{aligned} \tag{2.3}$$

with $|\boldsymbol{\Sigma}|$ the determinant of the covariance matrix. In practice, the likelihood function can be defined in any alternative form based on the assumption of the distribution of the measurement error. Note that the denominator of the Bayes' formula in (2.2) is a normalization constant that does not affect the shape of the PPDF. As such, in the hereafter discussion concerning building surrogate systems, the notation $p(\boldsymbol{\theta}|\boldsymbol{d})$ or the terminology PPDF will only refer to the product $p(\boldsymbol{d}|\boldsymbol{\theta})p(\boldsymbol{\theta})$.

Due to the nonlinearity of the forward QoI $\boldsymbol{f}(\boldsymbol{\theta})$ with respect to the parameter vector $\boldsymbol{\theta}$, it is often difficult to draw samples from the PPDF directly. The MCMC method provides an effective approach to sample the posterior distribution, such as Metropolis-Hastings (M-H) algorithm [9] and its variants. The essence of the MCMC method is that that parameter samples are drawn from a proposal distribution instead of the PPDF and the Markov property guarantees the convergence of the proposal distribution to the posterior distribution. However, in practice, the convergence is often very slow when using a poor proposal distribution that deviates far from the posterior distribution. Thus, in this work, we use an improved MCMC algorithm based on the differential evolution adaptive Metropolis approach (DREAM) developed in [31, 32]. The DREAM algorithm uses multiple Markov chains simultaneously, where the set of all chains are viewed as a population and the sampling procedure is treated as the evolution of the population. As such, the classic proposal distribution used in the M-H algorithm is not necessary and the jump of each Markov chain at each step, i.e., the direction and the scale of jump, is determined by differential evolution coming from a genetic algorithm. It was shown in [31, 32] that DREAM is generally more efficient than traditional MCMC algorithms in the absence of additional information about the PPDF. Our algorithm will be built in the DREAM framework to develop surrogates for evaluating the likelihood function in (2.3).

3 METHODOLOGY

In this section, we introduce the aSG-hSC method for constructing the surrogate system for the PPDF. As discussed in [27], when approximating a multivariate PPDF with a low variance, i.e., with a thin mode, interpolating the PPDF itself results in a very slow convergence rate. Thus, in this effort, we instead construct the surrogate system for the logarithm of the PPDF. To provide context for our new iterative algorithm discribed in Section 3.3, we first discuss, in Section 3.1, the determination of the prior domain by means of global optimization, then, in Section 3.2, describe the high-order hierarchical polynomial basis and the adaptive sparse-grid interpolation.

3.1 Global optimization for determining the prior domain

To construct a surrogate system, we need to define a prior domain for each significant mode, which is a bounded domain covering the high-probability region of the mode. To this end, we start by searching the global maximum of $\log(p(\boldsymbol{\theta}|\boldsymbol{d}))$ using global optimization. As shown in Section 3.3, this will be conducted iteratively to search all significant modes of the PPDF. In this effort, we use the algorithm DIRECT, first proposed in [14], which is a derivative-free global optimization algorithm. The algorithm is an improvement of the standard Lipschitzian approach that eliminates the need to specify a Lipschitz constant.= Hereinafter, we denote by $\mathbb{D}(\cdot)$ the global optimization solver with the input being the objective function and the output being the global maximum. The searching domain for $\mathbb{D}(\cdot)$ is represented by

$$\Gamma = [a_1, b_1] \times \cdots \times [a_{N_\theta}, b_{N_\theta}], \tag{3.1}$$

which is set sufficiently large due to the lack of prior information.

By running the solver to obtain $\widehat{\boldsymbol{\theta}} = \mathbb{D}(\log(p(\boldsymbol{\theta}|\boldsymbol{d})))$, we then define a neighboring domain of $\widehat{\boldsymbol{\theta}}$ that covers the high-probability region of the PPDF around $\widehat{\boldsymbol{\theta}}$. Note that when $\boldsymbol{\theta}$ follows the multi-variate Gaussian distribution, the Hessian matrix of $-\log(p(\boldsymbol{\theta}|\boldsymbol{d}))$, denoted by $H(\boldsymbol{\theta})$, is just the inverse of the covariance matrix of $\boldsymbol{\theta}$, so that $H^{-1}(\boldsymbol{\theta})$ can provide sufficient information to determine the prior domain. In the case $p(\boldsymbol{\theta}|\boldsymbol{d})$ does not have a standard Gaussian mode around $\widehat{\boldsymbol{\theta}}$ (e.g., a skewed or twisted Gaussian mode), $H^{-1}(\boldsymbol{\theta})$ still provides good sensitivity information about $p(\boldsymbol{\theta}|\boldsymbol{d})$ by which an appropriate prior domain can be defined. Thus, we use $H^{-1}(\boldsymbol{\theta})$ as the basis of constructing the prior domain, as shown below.

We estimate the value of the Hessian matrix $H(\boldsymbol{\theta})$ at the global maximum $\widehat{\boldsymbol{\theta}}$ using the formulas in [26]. By defining $\rho(\boldsymbol{\theta}) = -\log(p(\boldsymbol{\theta}|\boldsymbol{d}))$, the Hessian can be estimated by

$$H_{ll}(\widehat{\boldsymbol{\theta}}) = \frac{\rho(\widehat{\boldsymbol{\theta}} + \Delta\boldsymbol{\theta}_l) - 2\rho(\widehat{\boldsymbol{\theta}}) + \rho(\widehat{\boldsymbol{\theta}} - \Delta\boldsymbol{\theta}_l)}{(\Delta\theta_l)^2}, \tag{3.2}$$

for the diagonal entries and

$$
\begin{aligned}
&H_{lm}(\widehat{\boldsymbol{\theta}}) \\
=&\frac{1}{4\Delta\theta_l\Delta\theta_m}[\rho(\widehat{\boldsymbol{\theta}}+\Delta\boldsymbol{\theta}_l+\Delta\boldsymbol{\theta}_m)-\rho(\widehat{\boldsymbol{\theta}}+\Delta\boldsymbol{\theta}_l-\Delta\boldsymbol{\theta}_m) \\
&\qquad -\rho(\widehat{\boldsymbol{\theta}}-\Delta\boldsymbol{\theta}_l+\Delta\boldsymbol{\theta}_m)+\rho(\widehat{\boldsymbol{\theta}}-\Delta\boldsymbol{\theta}_l-\Delta\boldsymbol{\theta}_m)]
\end{aligned} \tag{3.3}
$$

for the off-diagonal entries. Here $\Delta\boldsymbol{\theta}_l$ and $\Delta\boldsymbol{\theta}_m$ are vectors with zero elements except for the lth and mth entries which are equal to properly selected steps $\Delta\theta_l$ and $\Delta\theta_m$. By calculating the inverse $H^{-1}(\widehat{\boldsymbol{\theta}})$, the diagonal elements of $H^{-1}(\widehat{\boldsymbol{\theta}})$ are used to define the prior domain $\Gamma_{\widehat{\boldsymbol{\theta}}}$ covering $\widehat{\boldsymbol{\theta}}$, i.e.,

$$
\Gamma_{\widehat{\boldsymbol{\theta}}}=[\widehat{\boldsymbol{\theta}}-\beta\boldsymbol{\sigma},\widehat{\boldsymbol{\theta}}+\beta\boldsymbol{\sigma}], \tag{3.4}
$$

where $\boldsymbol{\sigma}=(\sigma_1,\ldots,\sigma_{N_\theta})$ is a vector of the square roots of the diagonal elements of $H^{-1}(\widehat{\boldsymbol{\theta}})$, i.e., an estimation of the standard deviations of $p(\boldsymbol{\theta}|\boldsymbol{d})$ in the neighborhood of $\widehat{\boldsymbol{\theta}}$, with $\beta>0$ a user-defined constant. It is well known that in the Gaussian case, 99.7% of the samples are within 3 standard deviations ($\beta=3$) from the mean value. Although it is difficult to automatically find an optimal value of β that forces the prior domain $\Gamma_{\widehat{\boldsymbol{\theta}}}$ to just fit the desired high-probability region for a non-Gaussian mode, we can set it a little larger, e.g., $\beta=6$, to guarantee that $\Gamma_{\widehat{\boldsymbol{\theta}}}$ covers the significant mode very well for most common cases.

While both the global optimization solver $\mathbb{D}(\cdot)$ and the calculation of the Hessian matrix require forward model executions, such computational expense is worthwhile because it provides a bounded prior domain $\Gamma_{\widehat{\boldsymbol{\theta}}}$ having a significantly reduced volume, compared to the volume of the searching domain Γ in (3.1). This results in a diminished number of model executions required to construct the surrogate system of the PPDF. If the PPDF $p(\boldsymbol{\theta}|\boldsymbol{d})$ has multiple significant modes, the procedure discussed above will be used in each iteration of our new algorithm discussed in Section 3.3.

3.2 Adaptive sparse-grid high-order stochastic collocation method

After obtaining the prior domain $\Gamma_{\widehat{\boldsymbol{\theta}}}$ in (3.4) for the significant mode of the PPDF around $\widehat{\boldsymbol{\theta}}$ in the parameter space, the next task is to build the surrogate model for $\log(p(\boldsymbol{\theta}|\boldsymbol{d}))$ on $\Gamma_{\widehat{\boldsymbol{\theta}}}$ using the aSG-hSC method. Here, we briefly describe the development of the aSG-hSC method and refer to [2, 5, 11, 16] for details. For convenience of discussion, we use a general function $\eta(\boldsymbol{\theta})$ to represent $\log(p(\boldsymbol{\theta}|\boldsymbol{d}))$.

3.2.1 One-dimensional hierarchical interpolation

We begin by introducing hierarchical interpolation in one-dimensional (1-D) which is then used to construct the desired sparse-grid approximation in the multi-dimensional setting. Consider a function $\eta(\theta):[0,1]\to\mathbb{R}$ where the standard domain $[0,1]$ can be re-scaled to any bounded

domain by translation and dilation. The 1-D hierarchical Lagrange interpolation formula is defined by

$$\mathcal{U}^L(\eta)(\theta) = \sum_{i=0}^{L} \Delta\mathcal{U}^i(\eta)(\theta), \tag{3.5}$$

where the incremental interpolation operator $\Delta\mathcal{U}^i(\eta)$ is given as

$$\Delta\mathcal{U}^i(\eta)(\theta) = \sum_{j=0}^{m_i} c_j^i \phi_j^i(\theta) \quad i = 0, \ldots, L. \tag{3.6}$$

The nonnegative integer L in (3.5) is called the *resolution level* of the hierarchical interpolant $\mathcal{U}^L(\eta)$ and the summation over the resolution level in (3.5) exhibits the hierarchical structure of the interpolant $\mathcal{U}^L(\eta)$. For $j = 1, \ldots, m_i$, $\phi_j^i(\theta)$ and c_j^i in (3.6) are the basis functions and the interpolation coefficients for $\Delta\mathcal{U}^i(\eta)$, respectively. For $i = 0, \ldots, L$, the integer m_i in (3.6) is the number of interpolation points involved in $\Delta\mathcal{U}^i(\eta)$, which is defined by

$$\begin{cases} m_0 = 1 \\ m_1 = 2 \\ m_i = 2^{i-1} \quad \text{if} \quad i \geq 2. \end{cases} \tag{3.7}$$

We utilize a uniform grid, denoted by $\Delta\mathcal{X}^i = \{\theta_j^i\}_{j=1}^{m_i}$, for the incremental interpolant $\Delta\mathcal{U}^i(\eta)$. The abscissas of $\Delta\mathcal{X}^i$ are defined by

$$\begin{aligned} &\theta_1^0 = 0.5 && \text{for} \quad i = 0, \\ &\theta_1^1 = 0,\ \theta_2^1 = 1 && \text{for} \quad i = 1, \\ &\theta_j^i = \frac{2j-1}{\sum_{k=0}^{i} m_k - 1} && \text{for} \quad j = 1 \ldots, m_i,\ i \geq 2. \end{aligned} \tag{3.8}$$

Then, the hierarchical grid for $\mathcal{U}^L(\eta)(\theta)$ is defined by

$$\mathcal{X}^L = \bigcup_{i=0}^{L} \Delta\mathcal{X}^i. \tag{3.9}$$

Since the representation of c_j^i depends on the properties of the selected basis function $\phi_j^i(\theta)$, we first discuss the selection of these functions. Different from the previous studies that utilize linear hierarchical basis functions to build surrogate models, in this study, we use high-order hierarchical polynomial basis functions, including quadratic and cubic hierarchical basis defined in [5], in order to improve the accuracy and efficiency when constructing the surrogate system. For comparisons, below we provide the expressions for linear, quadratic, and cubic hierarchical polynomial bases.

Linear hierarchical basis For $i = 0$, $m_i = 1$

$$\phi_1^0(\theta) = 1 \text{ on } [0, 1]. \tag{3.10}$$

For $i > 0$, $j = 1, \ldots, m_i$,

$$\phi_j^i(\theta) = \begin{cases} 1 - \dfrac{\left|\theta - \theta_j^i\right|}{d\theta}, & \text{if} \quad \left|\theta - \theta_j^i\right| < d\theta, \\ 0, & \text{otherwise,} \end{cases} \tag{3.11}$$

where $d\theta = \frac{1}{2^i}$.

Quadratic hierarchical basis For $i = 0$, we use the linear hierarchical basis defined in (3.10). For $i = 1$, $j = 1, 2$, define $d\theta = \frac{1}{2}$ and set

$$\phi_1^1(\theta) = \frac{\theta - \theta_1^1 - d\theta}{-d\theta} \cdot \frac{\theta - \theta_1^1 - 2d\theta}{-2d\theta} \text{ on } [0, 1], \tag{3.12}$$

$$\phi_2^1(\theta) = \frac{\theta - \theta_2^1 + d\theta}{d\theta} \cdot \frac{\theta - \theta_2^1 + 2d\theta}{2d\theta} \text{ on } [0, 1]. \tag{3.13}$$

For $i \geq 2$, $j = 1, \ldots, m_i$, let

$$\phi_j^i(\theta) = \begin{cases} \dfrac{\theta - \theta_j^i - d\theta}{-d\theta} \cdot \dfrac{\theta - \theta_j^i + d\theta}{d\theta} & \text{if } \theta \in \Pi_j^i, \\ 0, & \text{otherwise,} \end{cases} \tag{3.14}$$

where $d\theta = \frac{1}{2^i}$ and $\Pi_j^i = [\theta_j^i - d\theta, \theta_j^i + d\theta]$.

Cubic hierarchical basis For $i = 0$ and $i = 1$, we use the linear hierarchical basis and quadratic hierarchical basis, defined by (3.10) and (3.12)-(3.13) respectively. For $i \geq 2$, $j = 1, \ldots, m_i$ and j is *odd*,

$$\phi_j^i(\theta) = \begin{cases} \displaystyle\prod_{k=1}^{3} \frac{\theta - \theta_j^i - (2k-3)d\theta}{-(2k-3)d\theta}, & \text{if } \theta \in \Pi_j^i, \\ 0, & \text{otherwise.} \end{cases} \tag{3.15}$$

where $d\theta = \frac{1}{2^i}$ and $\Pi_j^i = [\theta_j^i - d\theta, \theta_j^i + d\theta]$.
For $i \geq 2$, $j = 1, \ldots, m_i$ and j is *even*,

$$\phi_j^i(\theta) = \begin{cases} \displaystyle\prod_{k=1}^{3} \frac{\theta - \theta_j^i - (2k-5)d\theta}{-(2k-5)d\theta}, & \text{if } \theta \in \Pi_j^i, \\ 0, & \text{otherwise.} \end{cases} \tag{3.16}$$

where $d\theta = \frac{1}{2^i}$ and $\Pi_j^i = [\theta_j^i - d\theta, \theta_j^i + d\theta]$.

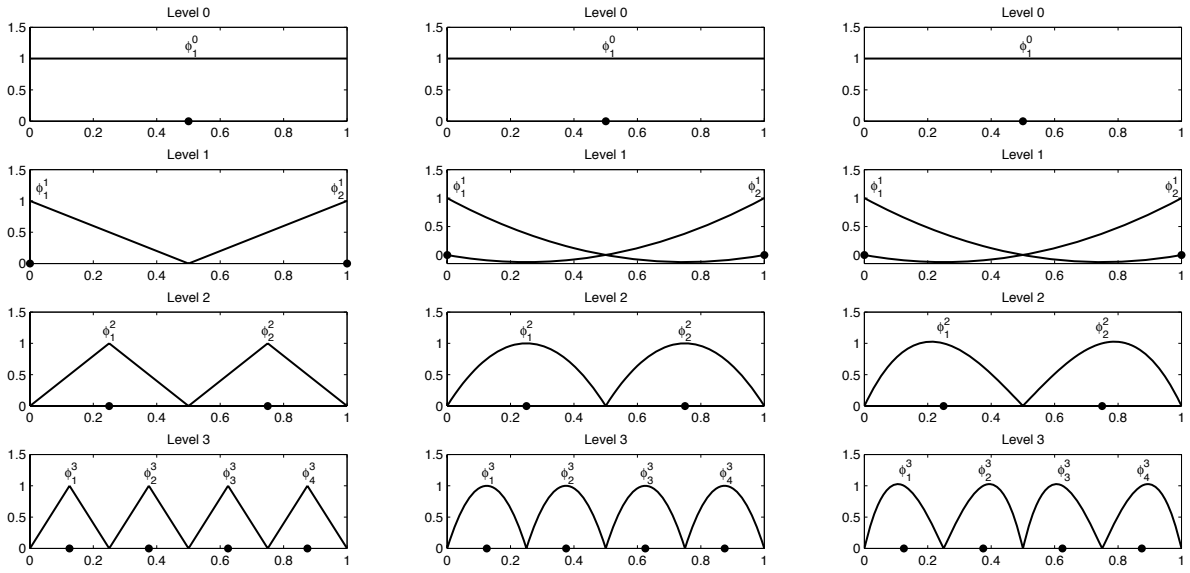

Figure 1: One dimensional 3-level hierarchical bases: linear basis (left), quadratic basis (middle), and cubic basis (right).

Figure 1 depicts the hierarchical basis functions from level 0 to level 3 for the linear, quadratic, and cubic bases given by (3.10)-(3.16). It is easy to see that on each level $i > 0$

$$\phi_j^i(\theta_k^i) = \begin{cases} 1, & \text{if} \quad j = k \\ 0, & \text{if} \quad j \neq k, j = 1, \ldots, m_i. \end{cases} \tag{3.17}$$

Thus, based on (3.5), (3.6), (3.17) and the interpolatory property of $\mathcal{U}^L(\eta)$, i.e., $\mathcal{U}^L(\eta)(\theta_j^i) = \eta(\theta_j^i)$ for $j = 1, \ldots, m_i$ and $i = 0, \ldots, L$, we can derive the representation of the coefficient c_j^i as follows. For $i = 0$,

$$c_1^0 = \Delta\mathcal{U}^0(\eta)(\theta_1^0) = \mathcal{U}^0(\eta)(\theta_1^0) = \eta(\theta_1^0) \tag{3.18}$$

and for $i > 0$, $j = 1, \ldots, m_i$,

$$\begin{aligned} c_j^i &= \Delta\mathcal{U}^i(\eta)(\theta_j^i) \\ &= \mathcal{U}^i(\eta)(\theta_j^i) - \mathcal{U}^{i-1}(\eta)(\theta_j^i) \\ &= \eta(\theta_j^i) - \mathcal{U}^{i-1}(\eta)(\theta_j^i). \end{aligned} \tag{3.19}$$

The coefficient c_j^i is defined as the hierarchical *surplus* of the basis function $\phi_j^i(\theta)$, which is the difference between the value of the interpolated function $\eta(\theta)$ and the value of the interpolant $\mathcal{U}^{i-1}(\eta)$ at θ_j^i. As discussed in [16] and [20], when the function $\eta(\theta)$ is smooth with respect to θ, the magnitude of the surplus c_j^i will approach to zero as the resolution level i increases, so that it can be used as an error indicator for the interpolant $\mathcal{U}^i(\eta)$ in order to guide the mesh refinement. Further discussion about building adaptive sparse grids will be provided in Section 3.2.3. First we explain how to interpolate multi-dimensional functions through the isotropic sparse grid construction that helps alleviate the *curse of dimensionality* in higher-dimensional problems.

3.2.2 Isotropic sparse-grid interpolation

Based on the one-dimensional hierarchical interpolation discussed in Section 3.2.1, we consider constructing an approximation for a multivariate function. We start from the isotropic sparse-grid interpolation by considering a function $\eta(\boldsymbol{\theta}) : [0,1]^{N_\theta} \to \mathbb{R}$ where $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_{N_\theta})$. Analogous to the definitions of $\mathcal{U}^L(\eta)$ in (3.5) and $\Delta\mathcal{U}^i(\eta)$ in (3.6), we define the multi-dimensional hierarchical interpolation formula as

$$\mathcal{I}^{L,N_\theta}(\eta)(\boldsymbol{\theta}) = \sum_{\lambda(\mathbf{i})\leq L} \Delta\mathcal{I}^{\mathbf{i},N_\theta}(\eta)(\boldsymbol{\theta}) \tag{3.20}$$

and the multi-dimensional incremental interpolation operator $\Delta\mathcal{I}^{\mathbf{i},N_\theta}(\eta)(\boldsymbol{\theta})$ is defined by

$$\Delta\mathcal{I}^{\mathbf{i},N_\theta}(\eta)(\boldsymbol{\theta}) = \Delta\mathcal{U}^{i_1} \otimes \cdots \otimes \Delta\mathcal{U}^{i_{N_\theta}}(\eta)(\boldsymbol{\theta}) = \sum_{\mathbf{j}\in B_\mathbf{i}} c_\mathbf{j}^\mathbf{i} \boldsymbol{\phi}_\mathbf{j}^\mathbf{i}(\boldsymbol{\theta}), \tag{3.21}$$

where $\mathbf{i} = (i_1, \ldots, i_{N_\theta})$ is a multi-index that indicates the resolution level of $\Delta\mathcal{I}^{\mathbf{i},N_\theta}(\eta)$, $\lambda(\mathbf{i}) : \mathbb{N}^{N_\theta} \to \mathbb{N}$ is a strictly increasing function, $\mathbf{j} = (j_1, \ldots, j_{N_\theta})$ belongs to the multi-index set

$$B_\mathbf{i} = \left\{ \mathbf{j} \in \mathbb{N}^{N_\theta} \middle| \, j_n = 1, \ldots, m_{i_n}, n = 1, \ldots, N_\theta \right\}, \tag{3.22}$$

and $c_\mathbf{j}^\mathbf{i}$ is the multi-dimensional hierarchical surplus. The function $\boldsymbol{\phi}_\mathbf{j}^\mathbf{i}(\boldsymbol{\theta})$ is the multi-dimensional hierarchical function defined by

$$\boldsymbol{\phi}_\mathbf{j}^\mathbf{i}(\boldsymbol{\theta}) = \prod_{n=1}^{N_\theta} \phi_{j_n}^{i_n}(\theta_n), \tag{3.23}$$

where, for $n = 1, \ldots N_\theta$, $\phi_{j_n}^{i_n}(\theta_n)$ is the one-dimensional hierarchical basis function. We also define the multi-dimensional grid point $\boldsymbol{\theta}_\mathbf{j}^\mathbf{i} = (\theta_{1,j_1}^{i_1}, \ldots, \theta_{N_\theta,j_{N_\theta}}^{i_{N_\theta}})$ that corresponds to the basis $\boldsymbol{\phi}_\mathbf{j}^\mathbf{i}(\boldsymbol{\theta})$. From (3.21), we can see that the multi-dimensional incremental interpolation operator on level $\mathbf{i}$ is the tensor product of N_θ one-dimensional incremental interpolation operators, so we use the notation $\Delta\mathcal{U}^{i_1} \otimes \cdots \otimes \Delta\mathcal{U}^{i_{N_\theta}}(\eta)$ to illustrate the tensor-product operation. In the following discussion, we only use the equivalent notation $\Delta\mathcal{I}^{\mathbf{i},N_\theta}(\eta)$ to denote the incremental interpolation operator. The grids for $\Delta\mathcal{I}^{\mathbf{i},N_\theta}(\eta)$ and $\mathcal{I}^{L,N_\theta}(\eta)$, denoted by $\Delta\mathcal{H}^{\mathbf{i},N_\theta}$ and $\mathcal{H}^{L,N_\theta}$ respectively, are represented as

$$\begin{aligned} \Delta\mathcal{H}^{\mathbf{i},N_\theta} &= \Delta\mathcal{X}^{i_1} \times \cdots \times \Delta\mathcal{X}^{i_{N_\theta}} \\ \mathcal{H}^{L,N_\theta} &= \bigcup_{\lambda(\mathbf{i})\leq L} \Delta\mathcal{H}^{\mathbf{i},N_\theta}. \end{aligned} \tag{3.24}$$

Note that $\Delta\mathcal{I}^{\mathbf{i},N_\theta}(\eta)$ involves a total of $\prod_{n=1}^{N_\theta} m_{i_n}$ grid points. In addition, $\mathcal{I}^{L,N_\theta}(\eta)$ is composed of several incremental interpolants. Thus, the definition of the function $\lambda(\mathbf{i})$ in (3.20) determines the number of grid points involved in $\mathcal{I}^{L,N_\theta}(\eta)$ and also the structure of the resulting grid. We give the following two definitions corresponding to full tensor-product grids and isotropic sparse grids:

$$\begin{aligned} &\text{(a) Full tensor-product grid: } \lambda(\mathbf{i}) = \max_{n=1,\ldots,N_\theta} i_n, \\ &\text{(b) Isotropic sparse grid: } \lambda(\mathbf{i}) = |\mathbf{i}| = i_1 + \cdots + i_{N_\theta}. \end{aligned} \tag{3.25}$$

Clearly, a L-level full tensor-product interpolant needs $(\sum_{i=1}^{L} m_i)^{N_\theta}$ grid points and function values where m_i is defined in (3.7). This is also the number of model executions needed when building the surrogate system. Using the full tensor-product formulation, the number of grid points grows exponentially with the number of random parameters N_θ, and therefore, we utilize the second definition of $\lambda(\mathbf{i})$, which corresponds to the isotropic sparse-grid interpolation, to combat the *curse of dimensionality* as the dimension N_θ increases.

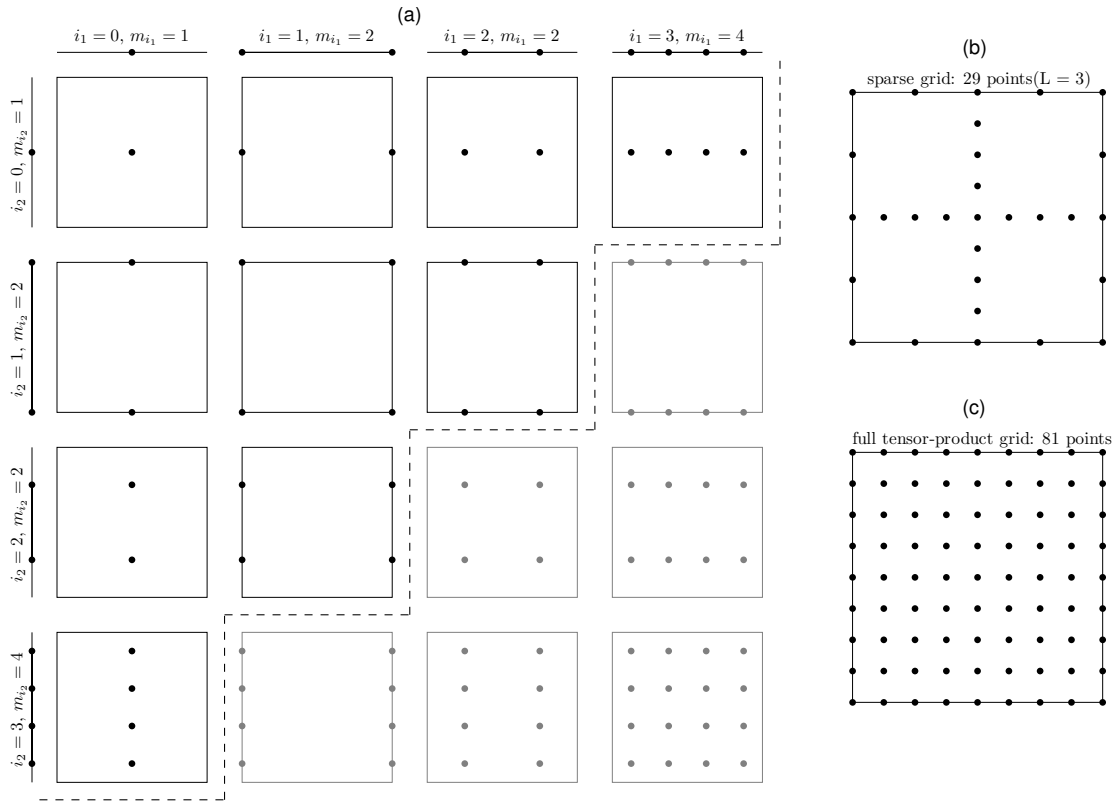

Figure 2: A comparison of a 3-level isotropic sparse grid (b) and the corresponding full tensor-product grid (c) based on Newton-Cotes points. The sparse grid (b) consists of 10 coarse sub-grids above the dashed line in (a)(grids shown in black), each of which is a coarse tensor-product grid with $i_1 + i_2 \leq 3$. The sparse grid has only 29 points. The full tensor-product grid (c), constructed by the all 16 sub-grids in (a) (grids shown in both black and gray) with $\max(i_1, i_2) \leq 3$, has 81 points.

The definitions of $\lambda(\mathbf{i})$ in (3.25) show that an L-level isotropic sparse grid is a sub-grid of an L-level full tensor-product grid. For example, the construction of a two-dimensional level $L = 3$ isotropic sparse grid is illustrated in Figure 2. The resolution level in one dimension can be $i_1 = 0, 1, 2, 3$ as shown in the top horizontal lines in Figure 2(a). The same is true for the other dimension as shown in the left vertical lines. There are a total of 16 sub-grids in Figure 2(a), each of which corresponds to an incremental interpolant $\Delta\mathcal{I}^{\mathbf{i},2}(\eta)$ in (3.20), where $\mathbf{i} = (i_1, i_2)$ and $0 \leq i_1, i_2 \leq 3$. Different combinations of i_1 and i_2 with $\max(i_1, i_2) \leq 3$ lead to all the 16 sub-grids in Figure 2(a), the union of which constitutes the level $L = 3$ full tensor-product grid with the 81 grid points shown in Figure 2(c). In comparison, different combinations of i_1 and i_2

with $|\mathbf{i}| = i_1 + i_2 \leq 3$ lead to only 10 sub-grids above the dashed line in Figure 2(a), the union of which constitutes the level $L = 3$ isotropic sparse grid with only the 29 grid points shown in Figure 2(b). This reduction is significant even though the maximum number of interpolation points in each dimension is the same for the both grids. Generally speaking, an isotropic sparse grid involves approximately $\mathcal{O}(M \cdot \log(M)^{N_\theta - 1})$ points where $M = \sum_{i=1}^{L} m_i$, whereas a full tensor-product grid involves $\mathcal{O}(M^{N_\theta})$ points [25]. However, even though many fewer points are used the accuracy of the sparse-grid interpolation does not appreciably deteriorate compared to that for the full tensor-product interpolation, as proved by [2, 5]. Thus, in the sequel, we fix the definition of $\lambda(\mathbf{i})$ in (3.20) to be $\lambda(\mathbf{i}) = |\mathbf{i}| = i_1 + \cdots + i_{N_\theta}$ and refer to (3.20) as an isotropic sparse-grid interpolant.

Next, we turn to the computation of the coefficients $c_{\mathbf{j}}^{\mathbf{i}}$. Analogous to the one-dimensional case and the discussion in [16], for $L = 0$, i.e., $i_1 = i_2 = \cdots = i_{N_\theta} = 0$, we have

$$c_{\mathbf{1}}^{\mathbf{0}} = \Delta\mathcal{I}^{0,N_\theta}(\eta)(\boldsymbol{\theta}_{\mathbf{1}}^{\mathbf{0}}) = \mathcal{I}^{0,N_\theta}(\eta)(\boldsymbol{\theta}_{\mathbf{1}}^{\mathbf{0}}) = \eta(\boldsymbol{\theta}_{\mathbf{1}}^{\mathbf{0}}). \tag{3.26}$$

For $L > 0$, $|\mathbf{i}| = L$ and $\forall \boldsymbol{\theta}_{\mathbf{j}}^{\mathbf{i}} \in \Delta\mathcal{H}^{\mathbf{i},N_\theta}$, we arrive at

$$\begin{aligned} c_{\mathbf{j}}^{\mathbf{i}} &= \Delta\mathcal{I}^{\mathbf{i},N_\theta}(\eta)(\boldsymbol{\theta}_{\mathbf{j}}^{\mathbf{i}}) \\ &= \mathcal{I}^{L,N_\theta}(\eta)(\boldsymbol{\theta}_{\mathbf{j}}^{\mathbf{i}}) - \mathcal{I}^{L-1,N_\theta}(\eta)(\boldsymbol{\theta}_{\mathbf{j}}^{\mathbf{i}}) \\ &= \eta(\boldsymbol{\theta}_{\mathbf{j}}^{\mathbf{i}}) - \mathcal{I}^{L-1,N_\theta}(\eta)(\boldsymbol{\theta}_{\mathbf{j}}^{\mathbf{i}}). \end{aligned} \tag{3.27}$$

Next, we will explain how to construct aSG-hSC approximations using the higher-order hierarchical polynomials described in (3.10)-(3.16).

3.2.3 Adaptive sparse-grid interpolation

Analogous to the discussion in Section 3.2.1, in the multi-dimensional case, if $\eta(\boldsymbol{\theta})$ is smooth with respect to $\boldsymbol{\theta}$, the magnitude of the hierarchical surplus will also decay to zero as the resolution level L of $\mathcal{I}^{L,N_\theta}(\eta)$ increases. Furthermore, the smoother the function $\eta(\boldsymbol{\theta})$ is, the faster the surplus decays. This provides us a good approach to construct adaptive sparse-grid interpolants using the surplus as an error indicator. Another adaptive sparse grid stochastic collocation approach that uses the magnitude of wavelet coefficients to guide refinement is described in [12].

We start by focusing on the construction of one-dimensional adaptive grids and then extending the adaptivity to the multi-dimensional sparse grid. As shown in Figure 3, the one-dimensional hierarchical grid points have a tree-like structure. In general, a grid point θ_j^i on level i has two children, namely θ_{2j-1}^{i+1} and θ_{2j}^{i+1} on level $i+1$. Special treatment is required when moving from level 1 to level 2, where we only add one child point on level 2 for each of the nodes θ_1^1 and θ_2^1. On each successive interpolation level, the basic idea of adaptivity is to use the hierarchical surplus as an error indicator to detect the smoothness of the target function and refine the grid by adding two new points on the next level for each point whose magnitude of the surplus is larger than the prescribed error tolerance. For example, in Figure 3, we illustrate the 6-level adaptive grid for interpolating the function $\eta(\theta) = \exp[-(\theta - 0.4)^2/0.0625^2]$ on $[0, 1]$ with the error tolerance being

0.01. From level 0 to level 2, because the magnitude of every surplus is larger than 0.01, two points are added for each grid point on level 0 and 2; one point is added for each grid point on level 1. But on level 3, there is only 1 point θ_2^3, the magnitude of whose surplus is larger than 0.01, so only two new points are added on level 4. If we proceed through levels 5 and 6, we end up with the 6-level adaptive grid with only 21 points (points in black in Figure 3), whereas the 6-level non-adaptive grid has a total of 65 points (points in black and gray in Figure 3).

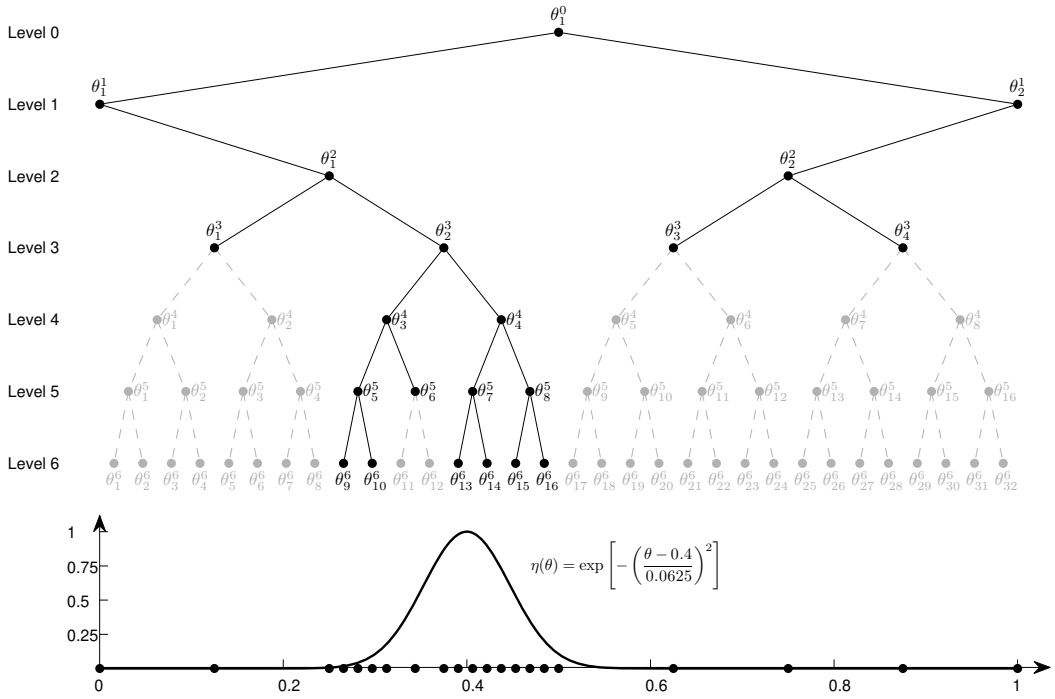

Figure 3: A 6-level adaptive sparse grid for interpolating a one-dimensional function $\eta(\theta) = \exp[-(\theta - 0.4)^2/0.0625^2]$ on $[0, 1]$ with the error tolerance of 0.01. The resulting adaptive sparse grid has only 21 points (black points) whereas the full grid has 65 points (black and gray points).

It is trivial to extend the adaptivity from the one-dimensional adaptive grid to the multi-dimensional adaptive sparse grid. The isotropic level L sparse grid $\mathcal{H}^{L,N_\theta}$ in (3.24) can be rewritten as

$$\mathcal{H}^{L,N_\theta} = \left\{ \boldsymbol{\theta}_{\mathbf{j}}^{\mathbf{i}} = \left(\theta_{j_1}^{i_1}, \ldots, \theta_{j_{N_\theta}}^{i_{N_\theta}} \right) \middle| \, |\mathbf{i}| \leq L \right\}, \tag{3.28}$$

where the grid points have the tree-like structure in each dimension. For example, a point $\boldsymbol{\theta}_{\mathbf{j}}^{\mathbf{i}} \in \mathcal{H}^{L,N_\theta}$ has 2 children points in each direction, so that it has a total of $2N_\theta$ children. For $n = 1, \ldots, N_\theta$, the two children of $\boldsymbol{\theta}_{\mathbf{j}}^{\mathbf{i}}$, denoted by $C_1^n(\boldsymbol{\theta}_{\mathbf{j}}^{\mathbf{i}})$ and $C_2^n(\boldsymbol{\theta}_{\mathbf{j}}^{\mathbf{i}})$, are represented by

$$\begin{aligned} C_1^n(\boldsymbol{\theta}_{\mathbf{j}}^{\mathbf{i}}) &= \left(\theta_{j_1}^{i_1}, \ldots, \theta_{j_{n-1}}^{i_{n-1}}, \theta_{2j_n-1}^{i_n+1}, \theta_{j_{n+1}}^{i_{n+1}}, \ldots, \theta_{j_{N_\theta}}^{i_{N_\theta}} \right), \\ C_2^n(\boldsymbol{\theta}_{\mathbf{j}}^{\mathbf{i}}) &= \left(\theta_{j_1}^{i_1}, \ldots, \theta_{j_{n-1}}^{i_{n-1}}, \theta_{2j_n}^{i_n+1}, \theta_{j_{n+1}}^{i_{n+1}}, \ldots, \theta_{j_{N_\theta}}^{i_{N_\theta}} \right), \end{aligned} \tag{3.29}$$

where $C_1^n(\boldsymbol{\theta}_{\mathbf{j}}^{\mathbf{i}}), C_2^n(\boldsymbol{\theta}_{\mathbf{j}}^{\mathbf{i}}) \in \Delta\mathcal{H}^{\widehat{\mathbf{i}}_n,N_\theta}$ with $\widehat{\mathbf{i}}_n = (i_1,\ldots,i_{n-1},i_n+1,i_{n+1},\ldots,i_{N_\theta})$ and $|\widehat{\mathbf{i}}_n| = |\mathbf{i}|+1$. Note that the children of each sparse-grid point on level $|\mathbf{i}|$ belong to the sparse-grid point set of level $|\mathbf{i}|+1$. By adding children points, we actually perform the sparse-grid interpolation from level $|\mathbf{i}|$ to level $|\mathbf{i}|+1$. Thus, in this way, we refine the sparse grid locally while not breaking the structure of sparse grids. For a prescribed error tolerance α, the adaptive sparse-grid interpolant is defined as

$$\mathcal{I}_\alpha^{L,N_\theta}(\eta)(\boldsymbol{\theta}) = \sum_{|\mathbf{i}|\le L}\sum_{\mathbf{j}\in B_{\mathbf{i}}^\alpha} c_{\mathbf{j}}^{\mathbf{i}}\phi_{\mathbf{j}}^{\mathbf{i}}(\boldsymbol{\theta}), \tag{3.30}$$

where the multi-index set $B_{\mathbf{i}}^\alpha$ can be defined by modifying the multi-index set $B_{\mathbf{i}}$ in (3.22), i.e.,

$$B_{\mathbf{i}}^\alpha = \left\{\mathbf{j}\in B_{\mathbf{i}} \,\middle|\, |c_{\mathbf{j}}^{\mathbf{i}}| > \alpha\right\}. \tag{3.31}$$

Thus, the level L adaptive sparse-grid interpolant $\mathcal{I}_\alpha^{L,N_\theta}(\eta)$ in (3.30) only retains the terms of the isotropic sparse-grid interpolant $\mathcal{I}^{L,N_\theta}(\eta)$ in (3.20) for which the magnitudes of the corresponding surpluses are larger than α. The corresponding adaptive spare grid can be represented by

$$\mathcal{H}_\alpha^{L,N_\theta} = \left\{\boldsymbol{\theta}_{\mathbf{j}}^{\mathbf{i}} \,\middle|\, |\mathbf{i}|\le L \text{ and } \mathbf{j}\in B_{\mathbf{i}}^\alpha\right\}, \tag{3.32}$$

which is a sub-grid of the level L isotropic sparse grid $\mathcal{H}^{L,N_\theta}$ in (3.24). If the tolerance $\alpha = 0$, the adaptive sparse-grid interpolant $\mathcal{I}_\alpha^{L,N_\theta}(\eta)$ is equivalent to the isotropic sparse grid interplant $\mathcal{I}_{L,N_\theta}(\eta)$ in (3.20); if $\alpha > 0$, it will adaptively select which points are added to the sparse grid. Subsequently, the sparse-grid points will become concentrated in the non-smooth region, e.g., where oscillations or sharp transitions occur, to guarantee the prescribed accuracy of the interpolation. On the other hand, in the region where $\eta(\boldsymbol{\theta})$ is very smooth, e.g., insensitive to certain parameters, this approach will save a significant number of grid points but still achieve the prescribed accuracy. Therefore, the adaptive sparse-grid interpolation can further reduce the number of model executions, which significantly improves the efficiency for building the surrogate system.

3.3 Algorithm for constructing the surrogate PPDF

In this section, we introduce our algorithm for building the surrogate system for $\log(p(\boldsymbol{\theta}|\boldsymbol{d}))$ using the global optimization and aSG-hSC method discussed in Section 3.1 and 3.2, respectively. We remark that the PPDF may have multiple significant modes due to the non-linearity of reactive transport models. Thus, we propose an iterative algorithm in order to capture all the significant modes one-by-one, and build the surrogate system consisting of several components, each of which approximates one significant mode.

We explain our algorithm according to the flow chart shown in Figure 4. First, the searching domain Γ for the global optimization solver $\mathbb{D}(\cdot)$ is defined as in (3.1). The target function $g(\boldsymbol{\theta})$ is set to

$$g(\boldsymbol{\theta}) = \log(p(\boldsymbol{\theta}|\boldsymbol{d})) + C \tag{3.33}$$

where $C > 0$ is a sufficiently large constant that enforces $g(\boldsymbol{\theta}) \ge 0$; we will discuss the functionality of C later in this section. The surrogate system, denoted by $S(\boldsymbol{\theta})$, has no component initially, i.e., $S(\boldsymbol{\theta}) = 0$. For the first iteration ($k = 1$) in Figure 4, the global optimization solver $\mathbb{D}(\cdot)$ is

used to search for the global maximum $\widehat{\boldsymbol{\theta}}_1$ of the function $g_1(\boldsymbol{\theta}) = g(\boldsymbol{\theta})$ where $g_1(\hat{\boldsymbol{\theta}}_1)$ is the highest peak of $g(\boldsymbol{\theta})$. Then, we calculate the inverse Hessian matrix $H^{-1}(\widehat{\boldsymbol{\theta}}_1)$ of $-g_1(\boldsymbol{\theta})$, determine the prior domain $\Gamma_{\widehat{\boldsymbol{\theta}}_1}$ in (3.4) that covers the high-probability region of the PPDF around $\widehat{\boldsymbol{\theta}}_1$, and then build the adaptive sparse-grid interpolant

$$S_1(\boldsymbol{\theta}) = \mathcal{I}_{\alpha}^{L,N_\theta}(g)(\boldsymbol{\theta}) \tag{3.34}$$

on the prior domain $\Gamma_{\widehat{\boldsymbol{\theta}}_1}$ by setting $\eta(\boldsymbol{\theta}) = g(\boldsymbol{\theta})$ in (3.30). $S_1(\boldsymbol{\theta})$ is the first component of the surrogate system $S(\boldsymbol{\theta})$. After that, $S(\boldsymbol{\theta})$ is updated to $S(\boldsymbol{\theta}) = S_1(\boldsymbol{\theta})\mathcal{X}_{\Gamma_{\widehat{\boldsymbol{\theta}}_1}}(\boldsymbol{\theta})$, where $\mathcal{X}_{\Gamma_{\widehat{\boldsymbol{\theta}}_1}}(\boldsymbol{\theta})$ is the characteristic function of the prior domain $\Gamma_{\widehat{\boldsymbol{\theta}}_1}$.

From Figure 4, for $k \geq 2$, because the surrogate system already has $k-1$ components, the output of the optimization solver $\mathbb{D}(\cdot)$ searches the global maximum of the remainder

$$g_k(\boldsymbol{\theta}) = g(\boldsymbol{\theta}) - S(\boldsymbol{\theta}) = g(\boldsymbol{\theta}) - \sum_{m=1}^{k-1} S_m(\boldsymbol{\theta})\mathcal{X}_{\Gamma_{\widehat{\boldsymbol{\theta}}_m}}(\boldsymbol{\theta}), \tag{3.35}$$

where, for $m = 1, \ldots, k-1$, $S_m(\boldsymbol{\theta})$ is the m-th component of the surrogate system defined on the domain $\Gamma_{\widehat{\boldsymbol{\theta}}_m}$. Namely, the maximum $\widehat{\boldsymbol{\theta}}_k$, defined by

$$\widehat{\boldsymbol{\theta}}_k = \mathbb{D}(g_k(\boldsymbol{\theta})), \tag{3.36}$$

represents the k-th highest peak of the PPDF $p(\boldsymbol{\theta}|\boldsymbol{d})$. Here we explain the functionality of the constant C in (3.33). According to the formula of the likelihood function in (2.3), $\log(p(\boldsymbol{\theta}|\boldsymbol{d}))$ may be negative at the maxima of some significant modes. In this case, if $C = 0$ in (3.33) and $\log(p(\widehat{\boldsymbol{\theta}}_k|\boldsymbol{d})) < 0$, since $g_k(\boldsymbol{\theta})$ is very close to zero in the domain $\bigcup_{m=1}^{k-1}\Gamma_{\widehat{\theta}_m}$, the output of $\mathbb{D}(g_k(\boldsymbol{\theta}))$ in (3.36) would not be $\widehat{\boldsymbol{\theta}}_k$ but some other value in $\bigcup_{m=1}^{k-1}\Gamma_{\widehat{\theta}_m}$. If this occurs, our algorithm fails to capture the remaining significant modes. Therefore, we use the sufficiently large shift constant $C > 0$ in (3.33) to guarantee $g(\boldsymbol{\theta}) \geq 0$ at the maximum of each significant mode.

We use the significance ratio, defined by

$$\delta = \exp\left[\frac{g(\widehat{\boldsymbol{\theta}}_k)}{g(\widehat{\boldsymbol{\theta}}_1)}\right] \tag{3.37}$$

to stop the algorithm when $g(\widehat{\boldsymbol{\theta}}_k)$ is too small to be significant. Here δ measures the ratio of the PPDF values at $\widehat{\boldsymbol{\theta}}_k$ and $\widehat{\boldsymbol{\theta}}_1$. If the ratio is smaller than a user-defined tolerance $\tilde{\delta}$, for example $\delta < \tilde{\delta} = 10^{-5}$, then on the PPDF, the height of the peak at $\widehat{\boldsymbol{\theta}}_k$ is much smaller than that of the highest peak at $\widehat{\boldsymbol{\theta}}_1$, so we do not need to construct a surrogate component for such a small mode (i.e., the mode is insignificant). Otherwise, if $\delta \geq \tilde{\delta}$, then the mode around $\widehat{\boldsymbol{\theta}}_k$ is considered as a significant mode. We treat $\widehat{\boldsymbol{\theta}}_k$ as $\widehat{\boldsymbol{\theta}}_1$ by constructing the k-th component $S_k(\boldsymbol{\theta})$ and add it to the surrogate system $S(\boldsymbol{\theta})$. Eventually, we end up with a surrogate system for $\log(p(\boldsymbol{\theta}|\boldsymbol{d}))$ with M components, i.e.,

$$S(\boldsymbol{\theta}) = \sum_{m=1}^{M} S_m(\boldsymbol{\theta})\mathcal{X}_{\Gamma_{\widehat{\boldsymbol{\theta}}_m}}(\boldsymbol{\theta}) - C \ \text{ on } \ \Gamma. \tag{3.38}$$

The total number of model executions for constructing $S(\boldsymbol{\theta})$ in (3.38) includes three contributions: the global optimization, the estimation of the Hessian matrix, and the adaptive sparse-grid interpolation. We discussed the first two parts in Section 3.1. Based on the computational gains provided by the reduced volume of the union of all the prior domains $\bigcup_{m=1}^{M} \Gamma_{\widehat{\boldsymbol{\theta}}_m}$, the use of the surrogate system $S(\boldsymbol{\theta})$ based on the high-order hierarchical basis can further reduce the total computational cost.

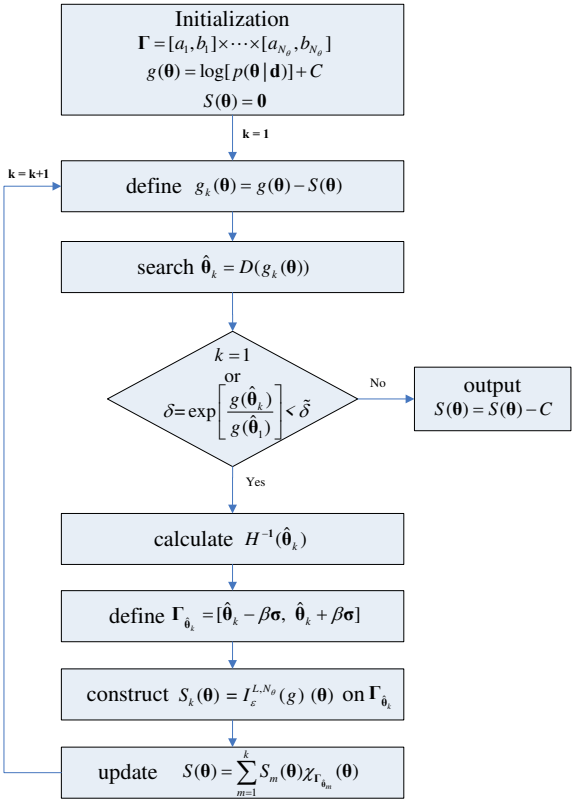

Figure 4: The algorithm for constructing the surrogate system of the PPDF.

4 APPLICATION IN GROUNDWATER REACTIVE TRANSPORT MODELING

In this section, first, the aSG-hSC method is evaluated in Appendix A using a bivariate function that can be viewed as a PDF of two parameters with four modes. This illustrates the performance of the iterative algorithm presented in Section 3.3 for building the surrogate system of a PDF with multiple significant modes. Next, the effectiveness and efficiency of the aSG-hSC method are illustrated in building the surrogate system for the PPDF using two synthetic groundwater reactive transport examples. The first example is about multi-species reactive transport discussed in [30]. The second example is related to reactive transport of uranium (VI) in column experiment, which is revised from [17]. While the first example involves only linear reactions, the second example is more complicated consisting of nonlinear reactions. we evaluate our aSG-hSC method by comparing the results of aSG-hSC-based MCMC with those of the conventional MCMC in approximating the PPDFs of model parameters and the PDFs of model predictions. In this work the efficiency of Bayesian inference is evaluated from two perspectives: (1) the number of model executions required to obtain an estimate of the PPDF within a prescribed accuracy, and (2) the accuracy of the approximate PPDF for a given number of model executions. These two criteria are complementary representing situations when large or limited model executions are affordable.

4.1 Case 1: Multi-species reactive transport equations

In the first synthetic study, we consider the transport of multiple reactive species coupled by a serial-parallel reaction network in a uniform flow field studied in [30]. As shown in Figure 5, the species A has one child species B and B has three child species C_1, C_2, C_3. The governing PDEs of simultaneous transport and degradation of the five species involved in the serial-parallel reaction network are as follows:

$$\begin{aligned}
&\frac{\partial c_A}{\partial t} - D\frac{\partial^2 c_A}{\partial x^2} + v\frac{\partial c_A}{\partial x} = -k_A c_A \\
&\frac{\partial c_B}{\partial t} - D\frac{\partial^2 c_B}{\partial x^2} + v\frac{\partial c_B}{\partial x} = y_B k_A c_A - k_B c_B \\
&\frac{\partial c_{C_1}}{\partial t} - D\frac{\partial^2 c_{C_1}}{\partial x^2} + v\frac{\partial c_{C_1}}{\partial x} = y_{C_1} k_B c_B - k_{C_1} c_{C_1} \\
&\frac{\partial c_{C_2}}{\partial t} - D\frac{\partial^2 c_{C_2}}{\partial x^2} + v\frac{\partial c_{C_2}}{\partial x} = y_{C_2} k_B c_B - k_{C_2} c_{C_2} \\
&\frac{\partial c_{C_3}}{\partial t} - D\frac{\partial^2 c_{C_3}}{\partial x^2} + v\frac{\partial c_{C_3}}{\partial x} = y_{C_3} k_B c_B - k_{C_3} c_{C_3},
\end{aligned} \tag{4.1}$$

where c_A, c_B, c_{C_1}, c_{C_2}, c_{C_3} are the concentrations of the five species A, B, C_1, C_2, C_3 respectively, t is the time, x is the spacial location in the domain [0,40], v is the constant flow velocity, D is the constant hydrodynamic dispersion coefficient, k_A, k_B, k_{C_1}, k_{C_2}, k_{C_3} are the reaction rates of the species, y_B is the stoichiometric yield factor that describes the production of its parent species A to B, and likewise for y_{C_1}, y_{C_2}, y_{C_3}.

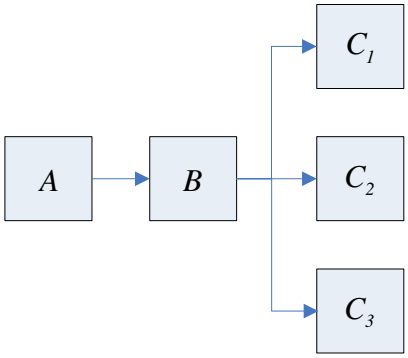

Figure 5: The serial-parallel reaction network in Case 1.

The true parameter values given in Table 1 are adopted from [30]. Based on these true parameter values, we generate the synthetic data by solving the system (4.1) at time $t = 40$ and $x = \{4, 8, 12, 16, 20, 24, 28, 32, 36, 40\}$. It generates a total of 50 simulated values of the five concentrations, c_A, c_B, c_{C_1}, c_{C_2}, and c_{C_3}. Then, these simulated values are corrupted with 3% Gaussian random noise, and the corrupted data are treated as measurements. In the MCMC simulation, six of the parameters listed in Table 1 are considered as unknown parameters which are the dispersion D and the logarithm of the five reaction rates, $\log(k_A)$, $\log(k_B)$, $\log(k_{C_1})$, $\log(k_{C_2})$ and $\log(k_{C_3})$.

Table 1: True parameter values used in Case 1.

Parameter	Symbol	Value
Dispersion	D	10
Velocity	v	0.4
Reaction rate of A	k_A	0.2
Reaction rate of B	k_B	0.1
Reaction rate of C_1	k_{C_1}	0.02
Reaction rate of C_2	k_{C_2}	0.02
Reaction rate of C_3	k_{C_3}	0.02
Stoichiometric yield of $A \to B$	y_B	0.5
Stoichiometric yield of $B \to C_1$	y_{C_1}	0.3
Stoichiometric yield of $B \to C_2$	y_{C_2}	0.2
Stoichiometric yield of $B \to C_3$	y_{C_3}	0.1

Here, we solve the PDEs in (4.1) using the numerical code PHT3D [28], and the number of forward model executions is equal to the number of times that PHT3D is called. To evaluate the performance of the aSG-hSC method, we take the conventional MCMC simulation results as reference, i.e., the results computed without using the surrogate system. The surrogate system is constructed by following the algorithm discussed in Section 3.3. The procedure of the algorithm is also illustrated in Figure 4. First, we define the searching domain Γ of the unknown parameters for the optimization solver $\mathbb{D}(\cdot)$ as shown in Table 2. Then, in the searching domain, we conduct

Table 2: The true values, the initial searching domain Γ and the prior domain $\Gamma_{\widehat{\boldsymbol{\theta}}_1}$ of the six unknown parameters in Case 1.

	D	$\log(k_A)$	$\log(k_B)$	$\log(k_{C_1})$	$\log(k_{C_2})$	$\log(k_{C_3})$
true value	10	-1.6094	-2.3026	-3.9120	-3.9120	-3.9120
Γ	[1,20]	[-10, -0.1]	[-10, -0.1]	[-10, -0.1]	[-10, -0.1]	[-10, -0.1]
$\Gamma_{\widehat{\boldsymbol{\theta}}_1}$	[8.0, 12.2]	[-1.8, -1.4]	[-2.5,-2.1]	[-4.5,-3.2]	[-4.5,-3.2]	[-4.5,-3.2]

the global optimization solver $\mathbb{D}(\cdot)$. The shift constant C in (3.33) is set to 10^4 here, which is considered large enough to make $\log(p(\boldsymbol{\theta}|\boldsymbol{d})) + C \geq 0$. The solver $\mathbb{D}(\cdot)$ takes 1034 model executions to find the first maximum of $g_1(\boldsymbol{\theta}) = \log(p(\boldsymbol{\theta}|\boldsymbol{d})) + C$ which is

$$\widehat{\boldsymbol{\theta}}_1 = (10.151, -1.607, -2.289, -3.819, -3.822, -3.911), \tag{4.2}$$

where the value of the logarithm of the joint PPDF at the peak $\widehat{\boldsymbol{\theta}}_1$ is $\log(p(\widehat{\boldsymbol{\theta}}_1|\boldsymbol{d})) = -7.9128$. Next, we compute the Hessian matrix $H(\widehat{\boldsymbol{\theta}}_1)$ using the formulas in (3.2) and (3.3), which requires 73 model executions. By taking the inverse, we obtain

$$H^{-1}(\widehat{\boldsymbol{\theta}}_1) = \begin{pmatrix} 0.1203 & 0.0080 & 0.0085 & 0.0277 & 0.0270 & 0.0296 \\ 0.0080 & 0.0007 & 0.0005 & 0.0019 & 0.0018 & 0.0020 \\ 0.0085 & 0.0005 & 0.0008 & 0.0021 & 0.0020 & 0.0023 \\ 0.0277 & 0.0019 & 0.0021 & 0.0101 & 0.0065 & 0.0071 \\ 0.0270 & 0.0018 & 0.0020 & 0.0065 & 0.0096 & 0.0070 \\ 0.0296 & 0.0020 & 0.0023 & 0.0071 & 0.0070 & 0.0112 \end{pmatrix}. \tag{4.3}$$

Based on this first found maximum, we construct the prior domain for $\widehat{\boldsymbol{\theta}}_1$ which is calculated using (3.4) with $\beta = 6$. The results are listed in Table 2 as $\Gamma_{\widehat{\boldsymbol{\theta}}_1}$. The difference between $\Gamma_{\widehat{\boldsymbol{\theta}}_1}$ and Γ in Table 2 indicates that the volume of the prior domain $\Gamma_{\widehat{\boldsymbol{\theta}}_1}$ is much smaller than the searching domain Γ, suggesting that building surrogate system on the prior domain can greatly reduce computational cost compared to that on the searching domain Γ. After that, on the prior domain $\Gamma_{\widehat{\boldsymbol{\theta}}_1}$, we construct the adaptive sparse-grid interpolant $\mathcal{I}_{\alpha}^{L,N_\theta}(\eta)(\boldsymbol{\theta})$ in (3.30) by setting $\eta(\boldsymbol{\theta}) = \log(p(\boldsymbol{\theta}|\boldsymbol{d})) + C$, $N_\theta = 6$, $L = 10$ and the tolerance $\alpha = 0.0001$, which is the first component $S_1(\boldsymbol{\theta})$ of the surrogate system $S(\boldsymbol{\theta})$. For comparison, we build three interpolants with linear, quadratic, and cubic basis functions, as shown in Figure 1. The number of model executions needed for the three interpolants are 6760, 1909, and 1299, respectively, which are also the number of points of the three corresponding adaptive sparse grids.

Next, based on the first component $S_1(\boldsymbol{\theta})$, we continue to search the second maximum of $\log(p(\boldsymbol{\theta}|\boldsymbol{d})) + C$ by having the global optimization solver $\mathbb{D}(\cdot)$ act on the remainder $g_2(\boldsymbol{\theta}) = \log(p(\boldsymbol{\theta}|\boldsymbol{d})) - S_1(\boldsymbol{\theta}) + C$. The solver $\mathbb{D}(\cdot)$ takes 1359 model executions to find the global maximum of $g_2(\boldsymbol{\theta})$ which is

$$\widehat{\boldsymbol{\theta}}_2 = (12.151, -1.535, -2.149, -4.222, -4.257, -3.583), \tag{4.4}$$

where the value of the logarithm of the PPDF at the peak $\widehat{\boldsymbol{\theta}}_2$ is $\log(p(\widehat{\boldsymbol{\theta}}_2|\boldsymbol{d})) = -45.2932$. By

setting the significance tolerance to $\tilde{\delta} = 10^{-5}$ in Figure 4, the significance ratio in (3.37) is

$$\delta = \frac{p(\widehat{\boldsymbol{\theta}}_2|\boldsymbol{d})}{p(\widehat{\boldsymbol{\theta}}_1|\boldsymbol{d})} = 5.8331 \times 10^{-17} < \tilde{\delta}. \tag{4.5}$$

This suggests that the second maximum is extremely small compared to the first one and there is no need to construct the surrogate component for it. So, finally, the surrogate system has only one component corresponding to the first maximum, as represented by $S(\boldsymbol{\theta}) = S_1(\boldsymbol{\theta}) - C$.

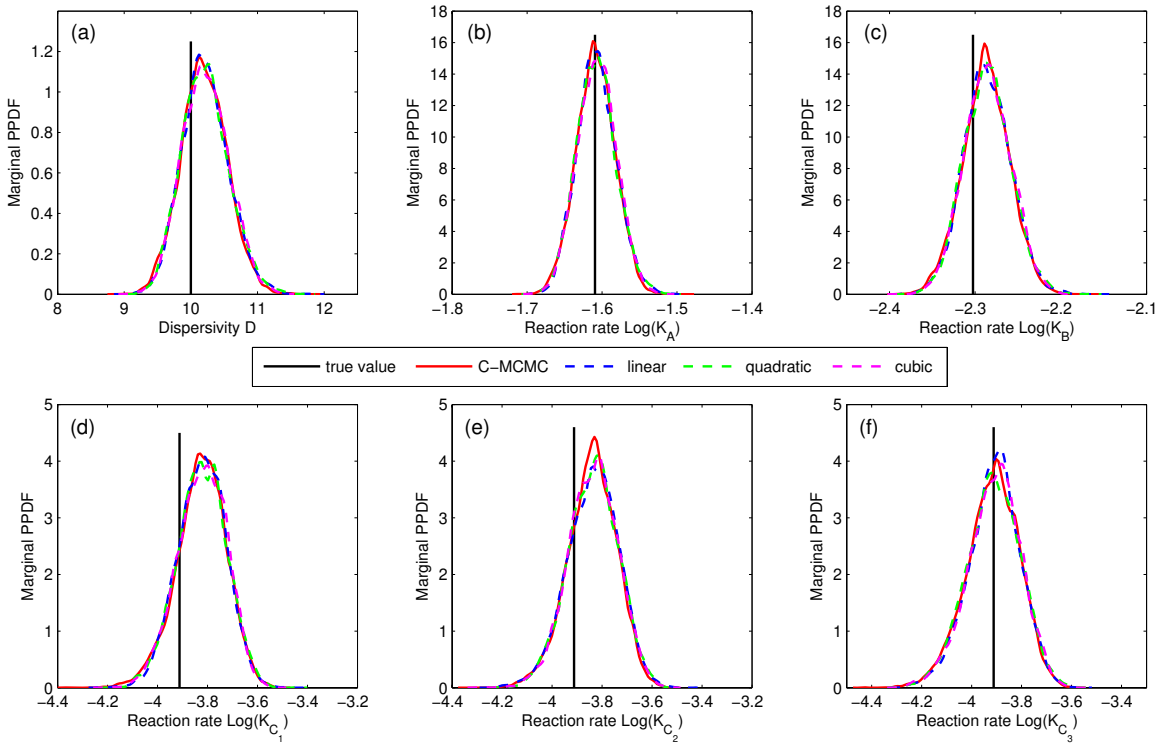

Figure 6: The marginal PPDFs for the six unknown parameters in Case 1 are estimated using the conventional MCMC (C-MCMC) with 60,000 model executions (red-solid lines), the linear, quadratic, and cubic surrogate systems with 9226, 4375, 3765 model executions (dashed lines), respectively. The true parameter values are plotted in black-solid lines. Take the conventional MCMC results as reference, the estimations by the surrogate systems are accurate enough but with computational cost greatly reduced.

With the constructed surrogate system $S(\boldsymbol{\theta})$, we explore the PPDF of model parameters by conducting the MCMC simulation which is executed in the searching domain Γ as listed in Table 2. The prior distribution of each parameter is assumed to be uniform distribution with bounds the same as the searching domain. Four MCMC simulations are conducted, three of which are based on the linear, quadratic, and cubic surrogate systems, respectively, and the other is conducted via the conventional MCMC without using the surrogate systems. Each MCMC simulation draws 60,000 parameter samples using three Markov chains, i.e., each chain evolves 20,000 generations. Before conducting further analysis, it is important to check convergence of the chains. Several methods

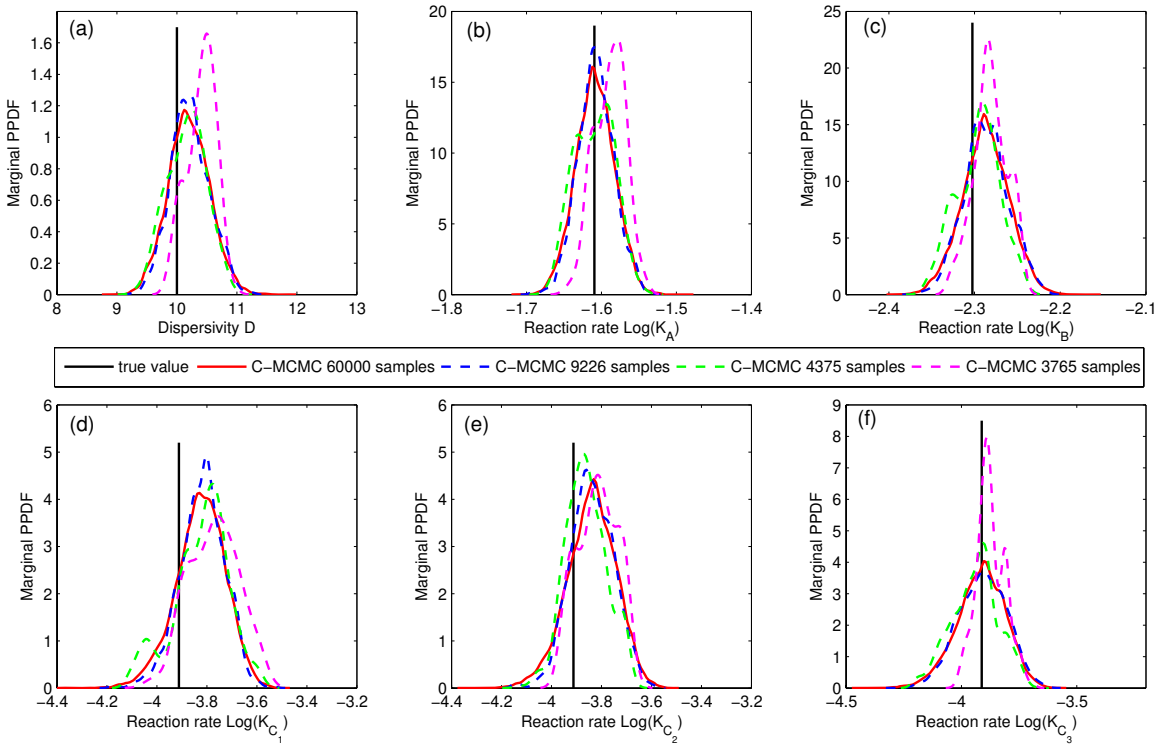

Figure 7: The marginal PPDFs for the six unknown parameters in Case 1 are estimated using the conventional MCMC (C-MCMC) with 60,000 model executions (red-solid lines) and 9226, 4375, 3765 model executions (dashed lines) which correspond to those used by the linear, quadratic, and cubic surrogate systems,respectively in Figure 6. The true parameter values are plotted in black-solid lines.

have been developed for the convergence diagnostics [6]. Here, we use the Gelman-Rubin R statistic [10] to determine the convergence (when the R is smaller than 1.2, the chains are thought to be converged). The calculated Gelman-Rubin R statistic indicates that the Markov chains converge after 720, 840, 970, and 760 generations for the linear, quadratic, and cubic surrogates and the conventional MCMC, respectively. For simplicity, we discard the first 1,000 generations in all the four simulations and use the remaining $60,000-1,000\times 3 = 57,000$ samples to explore the PPDF. The marginal PPDFs for the six parameters are plotted in Figure 6. The black vertical line represents the true value of each unknown parameter, as listed in Table 2. The red-solid lines represent the marginal PPDFs estimated by the conventional MCMC, and the dashed lines represent those estimated by the MCMC simulations based on the surrogate systems. Taking the conventional MCMC results as reference, Figure 6 indicates that the MCMC results based on the surrogate systems constructed by our aSG-hSC method can make good approximations for the marginal PPDFs. With the comparable accuracy, however, the surrogate-based MCMC needs significantly fewer model executions. The number of model executions for the conventional MCMC is 60,000 while the MCMC with linear, quadratic, and cubic surrogate systems only costs 9226, 4375 and 3765 model executions, respectively, where the 1034 executions for finding the first maximum, the 73 executions for calculating the Hessian matrix, the 6760, 1909, and 1299 executions for building the

linear, quadratic, and cubic adaptive sparse-grid interpolants, and the 1359 executions for finding the second maximum. For the surrogate-based MCMC simulations, drawing parameter samples does not take any model executions and the computational time is negligible since the calculation of the PPDF is just polynomial evaluation. The improvement of computational efficiency by using our surrogate systems is more outstanding when more parameter samples are drawn in the MCMC simulation. In addition, we compare the accuracy of the surrogate-based MCMC and the conventional MCMC with the same computational effort. The marginal PPDF for each parameter based on the conventional MCMC with 9226, 4375, 3765 samples, are plotted in Figure 7 as dashed lines. Comparing the corresponding graphs in Figure 6 and 7 indicates that with the same number of model executions, the approximations in Figure 6 using our surrogate systems are more accurate than those in Figure 7 using the conventional MCMC. This demonstrates the efficiency of our surrogate-based MCMC method from the other perspective.

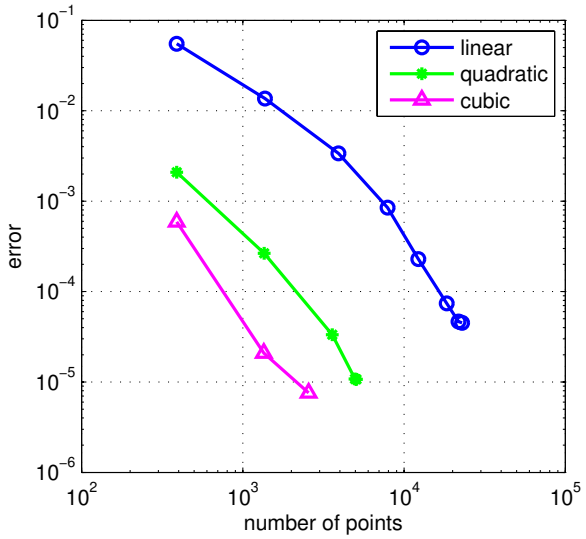

Figure 8: The errors of the surrogate systems based on linear, quadratic, and cubic hierarchical basis functions in Case 1.

Comparing different surrogate systems with linear, quadratic, and cubic interpolants, the distinction between Figure 6 and 7 suggests that the cubic surrogate system based on 3765 model executions shows the most improvement in efficiency. To further compare the efficiency between the three interpolants, we plot their error decay as the level L increases in Figure 8. To attain the same error, the cubic interpolant needs significantly fewer interpolation points than both the quadratic and linear interpolants. This indicates the surrogate system based on high-order hierarchical basis (i.e., the cubic basis) is more efficient than that with linear hierarchical basis. The estimated 95% credible intervals for the six unknown parameters based on the four MCMC simulations (three surrogate-based simulations and the conventional one) are listed in Table 3. Table 3 indicates that all the intervals can include the true parameter values listed in Table 1, and these intervals calculated based on different simulations are close to each other.

Finally, based on the parameter samples obtained in above MCMC simulations, we predict the

Table 3: The 95% credible intervals of the six unknown parameters obtained using the conventional MCMC and the surrogate systems with linear, quadratic, and cubic basis functions in Case 1.

	conventional MCMC	linear $S(\boldsymbol{\theta})$	quadratic $S(\boldsymbol{\theta})$	cubic $S(\boldsymbol{\theta})$
D	[9.5218,10.8981]	[9.5834, 10.9382]	[9.5723, 10.9560]	[9.5717, 10.9153]
$\log(k_A)$	[-1.6596, -1.5588]	[-1.6561, -1.5545]	[-1.6575, -1.5531]	[-1.6568, -1.5565]
$\log(k_B)$	[-2.3426, -2.2342]	[-2.3394, -2.2315]	[-2.3381, -2.2298]	[-2.3390, -2.2325]
$\log(k_{C_1})$	[-4.0410, -3.6442]	[-4.0137, -3.6462]	[-4.0223, -3.6378]	[-4.0254, -3.6501]
$\log(k_{C_2})$	[-4.0489, -3.6629]	[-4.0501, -3.6515]	[-4.0394, -3.6499]	[-4.0420, -3.6549]
$\log(k_{C_3})$	[-4.1405, -3.7258]	[-4.1259, -3.7205]	[-4.1338, -3.7143]	[-4.1340, -3.7151]

breakthrough curve of the concentration c_3 at velocity of $v = 0.2$. The breakthrough curve is predicted at time $t = 40$, and $x_i = 2i \in [0, 40]$, for $i = 1, \ldots, 20$. For each parameter sample drawn in the conventional MCMC, we run PHT3D to get predictions; and for that drawn in the surrogate-based MCMC, the predictions are estimated based on the surrogate system. For each predicted point in the breakthrough curve, the surrogate systems are built via $\mathcal{I}_{\alpha}^{L,N_\theta}(\eta)$ in (3.30) by setting $L = 10$ and the tolerance $\alpha = 0.0001$. In Figure 9(a), we plot the upper and lower bounds of the 95% credible intervals for the predictive breakthrough curve based on the conventional MCMC and the surrogate-based MCMC. As expected, the credible intervals via the surrogate systems match very closely to those obtained by the conventional MCMC but with computational effort greatly reduced. For example, Figure 9(b) shows the PDFs of point $c_3(x = 32, t = 40)$ in the predicted breakthrough curve based on the four MCMC simulation results. Figure 9(b) indicates that the four estimated PDFs are very similar to each other, but the surrogate system for the prediction $c_3(x = 32, t = 40)$ using linear, quadratic and cubic basis cost only 1853, 1032, 793 model executions respectively and the conventional MCMC requires 57,000 model executions.

4.2 Case 2: Reactive transport of uranium (VI) in column experiment

The second synthetic study is designed based on the real-world study of [17]. In order to study uranium reactive transport, [17] conducted eight column experiments in a well-characterized U(VI)-quartz-fluoride column system where seven of them have U(VI) concentration data. The concentration measurements have independent measurement errors with coefficient of variation of 3%. The breakthrough curves of U(VI) exiting the column over the course of several pore volumes of water showed the retardation effect due to uranium adsorption on the quartz surface. The uranium adsorption was simulated in [17] using seven alternative surface complex models (C1-C7) with different numbers of functional groups and different reaction stoichiometry. The models were calibrated against three column experiments (Experiments 1, 2, and 8) conducted under different experimental conditions and the calibrated models are used to predict the remaining four experiments (Experiments 3, 4, 5, and 7).

In the synthetic study, the true model is designed based on model C4 in [17]. As shown in Table 4, the true model has two functional groups, the weak site (S_1OH) and the strong site (S_2OH).

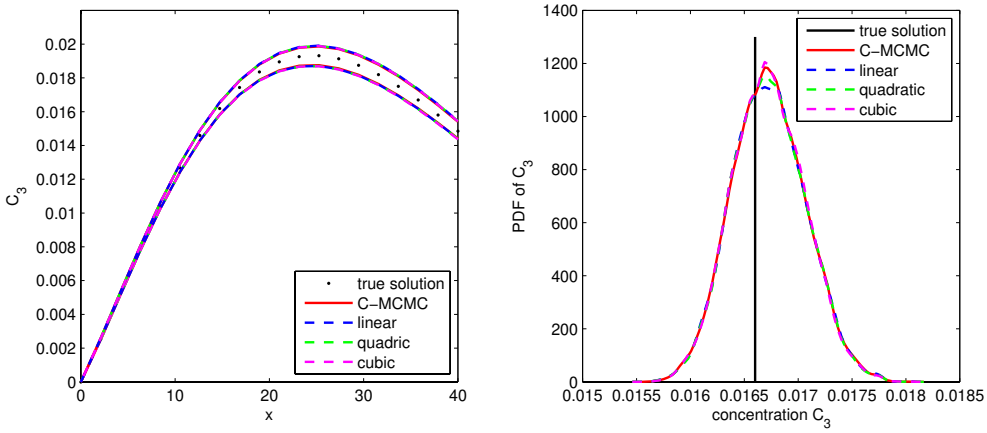

Figure 9: Left graph shows the true solution of $c_3(x,t)$ at $t = 40$ (black dots), the 95% credible intervals estimated using conventional MCMC (red-solid lines) and using the surrogate systems with linear, quadratic, and cubic hierarchical basis (dashed lines) in Case 1. Right graph shows the PDFs of a specific predicted quantity of $c_3(x = 32, t = 40)$ estimated using the conventional MCMC (red-solid line),and the linear, quadratic, and cubic surrogate systems (dashed lines) in Case 1. The true solution is plotted in black-solid line and the red-solid lines are taken as reference. The surrogate-based MCMC simulations require 1853, 1032, 793 model executions for the linear, quadratic and cubic surrogate system respectively, while the conventional MCMC requires 57,000 model executions

There are four parameters, three reaction rates and one site density for the strong site. Synthetic concentration data are first generated by the true models using the parameter values listed in Table 4, under the chemical conditions of Experiments 1, 2, and 8. The computer code RATEQ developed by [7] is used for the forward model execution. This process generates a total of 120 true values of uranium concentrations. The true values are then corrupted with 3% Gaussian random noise, and the 120 measurement data are used to define the likelihood function and build the surrogate system. To evaluate the performance of our method, we still use the conventional MCMC simulation result as a reference, i.e. the results computed without using the surrogate system.

Table 4: Surface complexation reactions and parameters of the true model in Case 2. Total site density used in this model is 1.3 M/L and the summation of site fraction in the last column is 1.0.

U(VI) surface reaction	reaction rate	site fraction
$S_1OH + UO_2^{2+} + H_2O = S_1OUO_2OH + 2H^+$	$K_1 = 7.3284 \times 10^{-3}$	1.0-Site
$S_2OH + UO_2^{2+} + H_2O = S_2OUO_2OH + 2H^+$	$K_2 = 3.4812 \times 10^{-2}$	Site = 1.8079×10^{-1}
$S_2OH + UO_2^{2+} = S_2OUO_2^+ + H^+$	$K_3 = 2.6353 \times 10^{0}$	

In this example, three parameters in Table 4 are considered as unknown parameters, which are the logarithms of the first two reaction rates denoted by $\log(K_1)$ and $\log(K_2)$, and the fraction of the strong site denoted by $\log(\text{Site})$. The searching domain Γ of the unknown parameters for the optimization solver $\mathbb{D}(\cdot)$ is given in Table 5. First we run the global optimization solver $\mathbb{D}(\cdot)$

Table 5: The true values, the initial searching domain Γ and the prior domain $\Gamma_{\widehat{\boldsymbol{\theta}}_1}$ of the three unknown parameters in Case 2.

	$\log(K_1)$	$\log(K_2)$	$\log(\text{Site})$
true value	-4.9160	-3.3578	-1.7104
Γ	[-10.0, -2.0]	[-6.0, -1.0]	[-4, -0.1]
$\Gamma_{\widehat{\boldsymbol{\theta}}_1}$	[-4.93, -4.90]	[-3.5, -3.3]	[-1.75,-1.68]

in the domain Γ. The shift constant C in (3.33) is set to 10^5 which is large enough to make $\log(p(\boldsymbol{\theta}|\boldsymbol{d})) + C \geq 0$. The solver $\mathbb{D}(\cdot)$ takes 2763 model executions to find the first maximum of $\log(p(\boldsymbol{\theta}|\boldsymbol{d}))$ which is

$$\widehat{\boldsymbol{\theta}}_1 = (-4.9169, -3.3603, -1.7111), \tag{4.6}$$

where the value of the logarithm of the PPDF at the peak $\widehat{\boldsymbol{\theta}}_1$ is $\log(p(\widehat{\boldsymbol{\theta}}_1|\boldsymbol{d})) = -11.4738$. Then, we compute the Hessian matrix $H(\widehat{\boldsymbol{\theta}}_1)$ using the formula in (3.2) and (3.3) which requires 19 model executions. By taking the inverse, we obtain

$$H^{-1}(\widehat{\boldsymbol{\theta}}_1) = \begin{pmatrix} 2.9495 \times 10^{-6} & 5.3356 \times 10^{-6} & -3.9045 \times 10^{-6} \\ 5.3356 \times 10^{-6} & 1.4448 \times 10^{-4} & -5.3501 \times 10^{-5} \\ -3.9045 \times 10^{-6} & -5.3501 \times 10^{-5} & 2.2257 \times 10^{-5} \end{pmatrix}. \tag{4.7}$$

Based on this first found maximum, we construct the prior domain for $\widehat{\boldsymbol{\theta}}_1$ which is calculated using (3.4) with $\beta = 6$. Results are shown in Table 5. Table 5 indicates that like in Case 1, the volume of prior domain $\Gamma_{\widehat{\boldsymbol{\theta}}_1}$ is much smaller than the searching domain Γ.

After that, based on the domain $\Gamma_{\widehat{\boldsymbol{\theta}}_1}$, we construct the adaptive sparse-grid interpolant $\mathcal{I}_{\alpha}^{L,N_\theta}(\eta)(\boldsymbol{\theta})$ in (3.30) by setting $\eta(\boldsymbol{\theta}) = \log(p(\boldsymbol{\theta}|\boldsymbol{d})) + C$, $N_\theta = 3$, $L = 8$ and the tolerance $\alpha = 0.001$, which is the first component $S_1(\boldsymbol{\theta})$ of the surrogate system $S(\boldsymbol{\theta})$. Again, we build three interpolants with linear, quadratic, and cubic basis functions. The number of model executions needed for the three interpolants are 1072, 705, and 341, respectively. The adaptive sparse grids with 1072, 705, 341 points and the same level $L = 8$ isotropic sparse grid with 6017 points are shown in Figure 10. Comparing these four graphs in Figure 10, it again suggests that the adaptive sparse grids surrogate used in this work is more efficient than the isotropic one, and the surrogate system based on high-order hierarchical basis is more efficient than the one with linear hierarchical basis.

Next, based on the first component $S_1(\boldsymbol{\theta})$, we continue to search the second maximum of $\log(p(\boldsymbol{\theta}|\boldsymbol{d})) + C$ by having the global optimization solver $\mathbb{D}(\cdot)$ acting on the remainder $g_2(\boldsymbol{\theta}) = \log(p(\boldsymbol{\theta}|\boldsymbol{d})) - S_1(\boldsymbol{\theta}) + C$. The solver $\mathbb{D}(\cdot)$ takes 3625 model executions to find the second maximum of $g_2(\boldsymbol{\theta})$ which is

$$\widehat{\boldsymbol{\theta}}_2 = (-5.1392, -3.8653, -1.6394), \tag{4.8}$$

where the value of the logarithm of the PPDF at the peak $\widehat{\boldsymbol{\theta}}_2$ is $\log(p(\widehat{\boldsymbol{\theta}}_2|\boldsymbol{d})) = -203.8976$. By setting the significance tolerance $\tilde{\delta} = 10^{-5}$ in Figure 4, the significance ratio δ in (3.37) is

$$\delta = \frac{p(\widehat{\boldsymbol{\theta}}_2|\boldsymbol{d})}{p(\widehat{\boldsymbol{\theta}}_1|\boldsymbol{d})} = 2.7003 \times 10^{-84} < \tilde{\delta}. \tag{4.9}$$

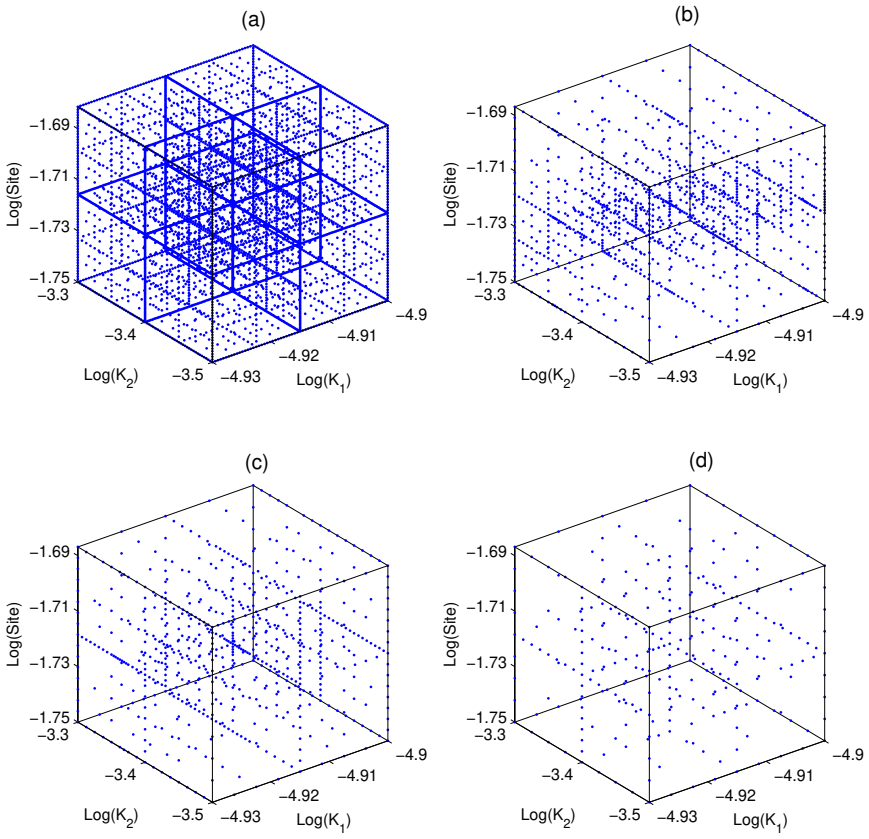

Figure 10: (a): The 8-level isotropic sparse grid with 6017 points; (b): the adaptive sparse grid with 1072 points for the linear surrogate system; (c): the adaptive sparse grid with 705 points for the quadratic surrogate system; and (d): the adaptive sparse grid with 341 points for the cubic surrogate system in Case 2.

So, finally, like Case 1, the surrogate system has only one component corresponding to the first maximum, as represented by $S(\boldsymbol{\theta}) = S_1(\boldsymbol{\theta}) - C$.

In this example, we only conduct two MCMC simulations, one is the conventional MCMC and the other is cubic surrogate based MCMC (since it is more efficient than the linear and quadratic ones.) For each MCMC simulation, like in Case 1, a total of 60,000 parameter samples are drawn using three Markov chains. The Gelman-Rubin R statistic indicates that the Markov chains converge after 510 and 590 generations for the cubic surrogate system and the conventional MCMC, respectively. For simplicity, we discard the first 600 generations in both simulations and use the remaining $60,000 - 600 \times 3 = 58,200$ samples to explore the PPDF. The marginal PPDFs of the three parameters and the PPDF contours of their combinations are plotted in Figure 12. As in Case 1, Figure 12 indicates that the MCMC results based on the surrogate systems constructed by our aSG-hSC method can make good approximations for the marginal PPDFs. With the comparable accuracy, however,the surrogate-based MCMC needs significantly fewer model executions. The number of model executions for the conventional MCMC is 60,000 while the surrogate-based MCMC only costs 6748 model executions.

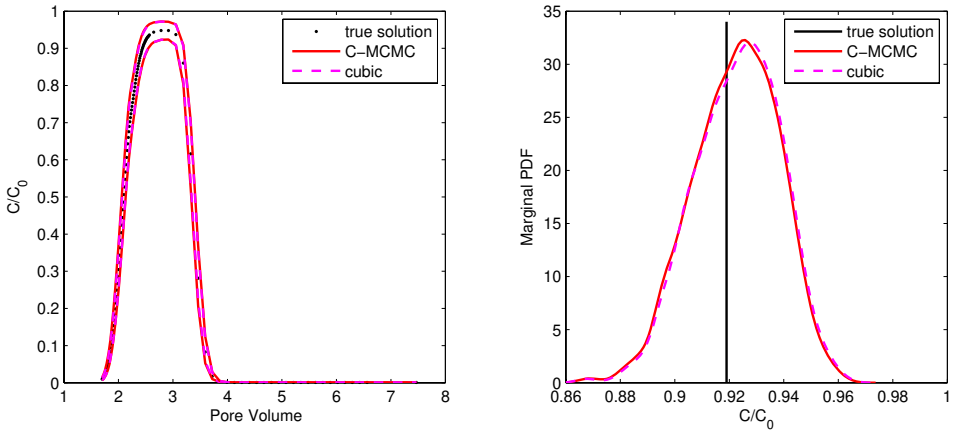

Figure 11: Left graph shows the true breakthrough curve of Experiment 4 in Case 2 (black dots), the 95% credible intervals estimated using the conventional MCMC (red-solid lines) and using the cubic surrogate system (pink-dashed lines). Right graph shows the PDFs of a specific predicted quantity at Pore Volume of 2.49 estimated using the conventional MCMC (red-solid line) with 58,200 model executions and the cubic surrogate system (pink-dashed line) with 411 model executions. The true solution is plotted in black-solid line and the red-solid lines are taken as reference.

Finally, based on the parameter samples obtained in above two MCMC simulations, we predict the breakthrough curve of Experiment 4 in [17]. It total contains 118 predicted points. Figure 11a plots the upper and lower bounds of the 95% credible intervals for the predictive breakthrough curve based on the conventional MCMC and the cubic surrogate based MCMC. As expected, the credible intervals via the cubic surrogate system match very closely to those obtained by the conventional MCMC but with computational effort greatly reduced. For example, Figure 11(b) shows the PDFs of point at Pore volume of 2.49 in the predicted breakthrough curve based on the two MCMC simulation results. Like in Case 1, Figure 11(b) indicates that the two estimated PDFs are very similar to each other, but the cubic surrogate system constructing the PDF costs only 411 model executions and the conventional MCMC requires 58,200 model executions.

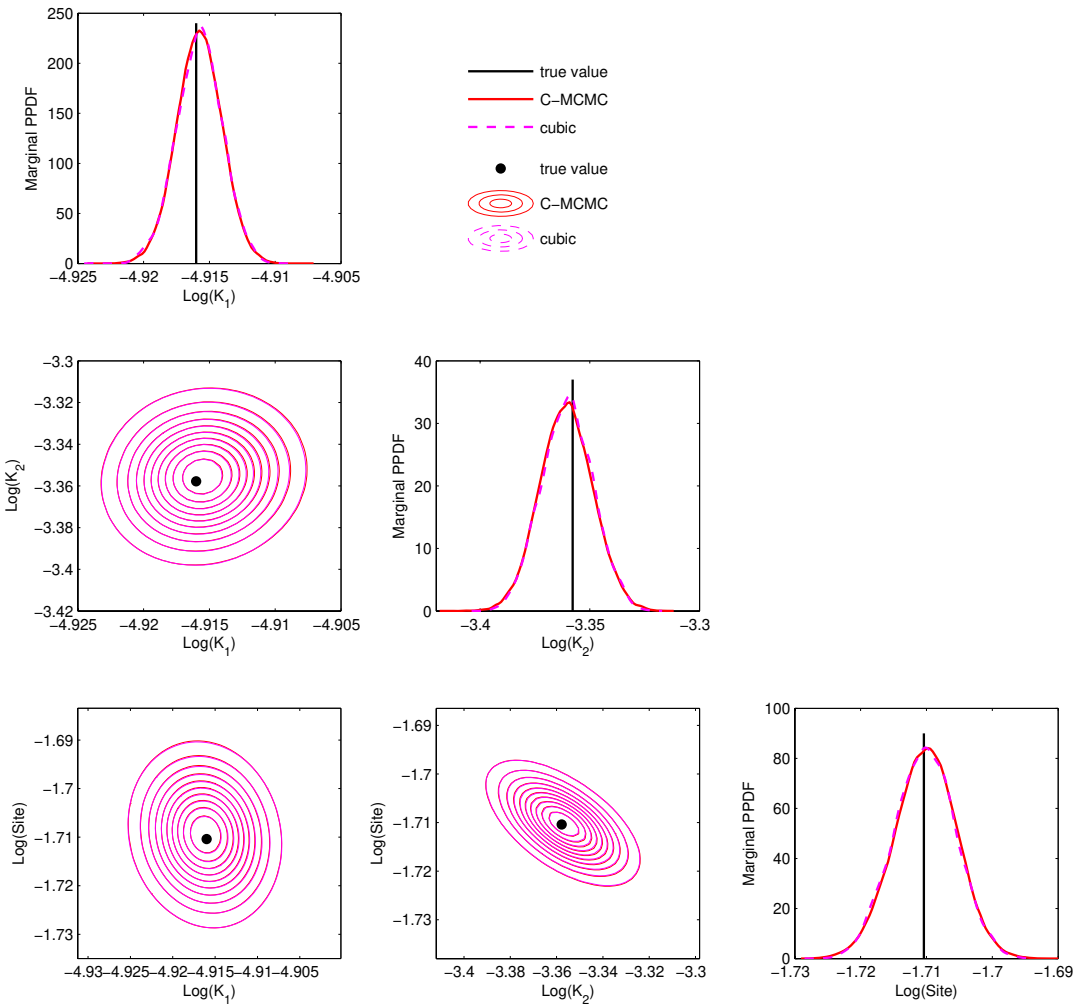

Figure 12: The 1D and 2D marginal PPDFs of the three unknown parameters in Case 2 are estimated using the conventional MCMC with 60,000 model executions (red-solid lines and contours) and the cubic surrogate system with 6748 model executions (pink-dashed lines and contours). The true parameter values are plotted in black-solid lines or black dots. Take the conventional MCMC results as reference, the estimations by the cubic surrogate system is accurate enough but with computational cost greatly reduced.

5 CONCLUSION

This work proposed a new adaptive sparse-grid high-order stochastic collocation method to improve the efficiency of the Bayesian inference. The high-order hierarchical polynomial basis is used to build the surrogate system for the PPDF. Combining the increased accuracy of the high-order hierarchical basis and the local adaptive sparse-grid technique, the computational cost for the desired surrogate system is greatly reduced compared to the existing methods. Moreover, we utilized a global optimization method and propose an iterative algorithm for building the surrogate system for the PPDF with multiple significant modes. The case studies compare our approach with the existing methods for building surrogate system in the context of Bayesian inference, and provide computational verification that our technique is efficient and superior to the compared methods. Such a computationally efficient method is critical for Bayeisan inference of time-consuming groundwater reactive transport modeling. Due to the non-intrusive nature of the new method, it can be used together with many models and sampling methods for Bayesian inference.

As a surrogate method, our approach also has some limitations. First, the performance of the algorithm relies on the ability of the selected global optimization solver. If the execution of the global optimization solver is computationally expensive, then it will deteriorate the efficiency of the method. If the solver fails to find the global maximum at a given iteration step in our algorithm, then we may miss a significant mode which may be lead to an incorrect estimation of the PPDF. In this case, one has to sacrifice computational efficiency and use more sparse grid points. However, it is worth mentioning that this kind of challenge is not specific to our method but to all numerical algorithms of uncertainty quantification and optimization. In addition, since we use the Hessian matrix to determine the prior domain for each significant mode, it remains empirical at this moment to find the optimal value for the user-defined constant β in (3.4). When the shape of the detected significant mode is extremely complicated, the commonly used value, e.g. $\beta = 6$, may not be appropriate to cover the high-probability region. In other words, the reduction of the bounds for building sparse grids may not be significant as shown in the numerical examples of this study. The major challenge resides in non-smoothness of the PPDF surface due to nonlinearity of groundwater reactive transport models. This requires further study in the future, especially for coupled biohydrogeochemical processes at larger scales, to determine applicability of the aSG-hSC method in groundwater modeling.

Appendix A BUILDING THE SURROGATE SYSTEM FOR A 2-D FUNCTION WITH MULTIPLE SIGNIFICANT MODES

In order to further test the performance of our method in approximating a PDF with multiple significant modes, we use our method to build the surrogate system for a 2-D artificial function with four modes. Although the target function is not a PPDF via the Bayesian approach, it is sufficient for the verification of the proposed algorithm. The 2-D target function is defined by

$$\begin{aligned}
\eta(\boldsymbol{\theta}) &= M_1(\boldsymbol{\theta}) + M_2(\boldsymbol{\theta}) + M_3(\boldsymbol{\theta}) + M_4(\boldsymbol{\theta}),\\
M_1(\boldsymbol{\theta}) &= w_1 \exp\left[(\boldsymbol{h}(\boldsymbol{\theta}) - \boldsymbol{\mu}_1)^\top \boldsymbol{\Sigma}_1^{-1} (\boldsymbol{h}(\boldsymbol{\theta}) - \boldsymbol{\mu}_1)\right],\\
M_2(\boldsymbol{\theta}) &= w_2 \exp\left[(\boldsymbol{h}(\boldsymbol{\theta}) - \boldsymbol{\mu}_2)^\top \boldsymbol{\Sigma}_2^{-1} (\boldsymbol{h}(\boldsymbol{\theta}) - \boldsymbol{\mu}_2)\right],\\
M_3(\boldsymbol{\theta}) &= w_3 \exp\left[(\boldsymbol{\theta} - \boldsymbol{\mu}_3)^\top \boldsymbol{\Sigma}_3^{-1} (\boldsymbol{\theta} - \boldsymbol{\mu}_3)\right],\\
M_4(\boldsymbol{\theta}) &= w_4 \exp\left[(\boldsymbol{\theta} - \boldsymbol{\mu}_4)^\top \boldsymbol{\Sigma}_4^{-1} (\boldsymbol{\theta} - \boldsymbol{\mu}_4)\right],
\end{aligned} \tag{A.1}$$

where $\boldsymbol{\theta} = (\theta_1, \theta_2)$, the translation function $\boldsymbol{h}(\boldsymbol{\theta}) = (h_1(\boldsymbol{\theta}), h_2(\boldsymbol{\theta}))$ is defined as

$$h_1(\boldsymbol{\theta}) = \theta_1 \quad \text{and} \quad h_2(\boldsymbol{\theta}) = \theta_2 + 0.2 \times \theta_1^2, \tag{A.2}$$

and the weights w_1, w_2, w_3, w_4 are set to

$$w_1 = 1.0, \quad w_2 = 2.0, \quad w_3 = 0.0002, \quad w_4 = 0.0001 \tag{A.3}$$

The means and covariance matrices in (A.1) are given by

$$\begin{aligned}
\boldsymbol{\mu}_1 &= [-6.0, -4.0], \quad \boldsymbol{\Sigma}_1 = \begin{pmatrix} 0.2 & 0.0 \\ 0.0 & 0.1 \end{pmatrix},\\
\boldsymbol{\mu}_2 &= [3.0, 1.0], \quad \boldsymbol{\Sigma}_2 = \begin{pmatrix} 0.2 & 0.0 \\ 0.0 & 0.1 \end{pmatrix},\\
\boldsymbol{\mu}_3 &= [-6.0, 5.0], \quad \boldsymbol{\Sigma}_3 = \begin{pmatrix} 0.1 & 0.0 \\ 0.0 & 0.1 \end{pmatrix},\\
\boldsymbol{\mu}_4 &= [6.0, 6.0], \quad \boldsymbol{\Sigma}_4 = \begin{pmatrix} 0.1 & 0.0 \\ 0.0 & 0.1 \end{pmatrix}.
\end{aligned} \tag{A.4}$$

(A.1)-(A.4) indicate that the target distribution $\eta(\boldsymbol{\theta})$ has 4 modes, of which the first two modes, $M_1(\boldsymbol{\theta})$ and $M_2(\boldsymbol{\theta})$, are twisted Gaussian modes due to the definition of $\boldsymbol{h}(\boldsymbol{\theta})$ and the other two modes, $M_3(\boldsymbol{\theta})$ and $M_4(\boldsymbol{\theta})$, are standard Gaussian modes. In Figure A1(a,b) we plot the surfaces of $\eta(\boldsymbol{\theta})$ and $\log(\eta(\boldsymbol{\theta}))$. $M_1(\boldsymbol{\theta})$ and $M_2(\boldsymbol{\theta})$ are more significant than $M_3(\boldsymbol{\theta})$ and $M_4(\boldsymbol{\theta})$ due to the much larger weights w_1, w_2. Thus, in Figure A1(a), only $M_1(\boldsymbol{\theta})$ and $M_2(\boldsymbol{\theta})$ are exhibited and the other two modes are too small to be displayed. The logarithm of η displays the four modes better, as Figure A1(b) of $\log(\eta(\boldsymbol{\theta}))$ shows all the four peaks. The coordinates of the four maxima are $(3.0, -0.8)$, $(-0.6, -0.32)$, $(-6.0, 5.0)$ and $(6.0, 6.0)$

Next, we treat $\eta(\boldsymbol{\theta})$ as a PPDF and apply the algorithm discussed in Section 3.3 to build a surrogate system. The searching domain Γ in (3.1) for the optimization solver $\mathbb{D}(\cdot)$ is set to $\Gamma =$

$[-10, 10] \times [-10, 10]$. The shift constant C in (3.33) is set to 1000 which is large enough to make $\log(\eta(\boldsymbol{\theta})) + C \geq 0$. The optimization solver $\mathbb{D}(\cdot)$ takes 284 function evaluations to find the largest maximum which is

$$\widehat{\boldsymbol{\theta}}_1 = (2.999, -0.801) \tag{A.5}$$

where the value of $\eta(\boldsymbol{\theta}_1)$ is 2.2508. Then we compute the Hessian matrix $H(\widehat{\boldsymbol{\theta}}_1)$ using the formula in (3.2), (3.3) which requires 9 function evaluations. By taking the inverse, we obtain $H^{-1}(\widehat{\boldsymbol{\theta}}_1)$

$$H^{-1}(\widehat{\boldsymbol{\theta}}_1) = \begin{pmatrix} 0.200 & -0.240 \\ -0.240 & 0.388 \end{pmatrix}, \tag{A.6}$$

then the prior domain for $\widehat{\boldsymbol{\theta}}_1$ is calculated using (3.4) with $\beta = 5$, which is

$$\Gamma_{\widehat{\boldsymbol{\theta}}_1} = [0.763, 5.236] \times [-3.915, 2.315], \tag{A.7}$$

whose volume is much smaller than the searching domain Γ.

After that, on the domain $\Gamma_{\widehat{\boldsymbol{\theta}}_1}$, we construct the adaptive sparse-grid interpolant $\mathcal{I}_{\alpha}^{L,N_\theta}(\eta)(\boldsymbol{\theta})$ in (3.30) by setting $\eta(\boldsymbol{\theta}) = \log(\eta(\boldsymbol{\theta})) + C$, $N_\theta = 2$, $L = 8$ and the tolerance $\alpha = 0.001$, which is the first component $S_1(\boldsymbol{\theta})$ of the surrogate system $S(\boldsymbol{\theta})$. In this example, we focus on testing the performance of our algorithm in detecting multiple significant modes, so we only use the cubic hierarchical basis to build the surrogate system. The number of function evaluations needed for the surrogate is 145, which is also the number of points of the adaptive sparse grid.

Based on the first component $S_1(\boldsymbol{\theta})$, we continue to search the second maximum of $\log(\eta(\boldsymbol{\theta}))+C$ by having the global optimization solver $\mathbb{D}(\cdot)$ acting on the remainder $g_2(\boldsymbol{\theta}) = \log(\eta(\boldsymbol{\theta})) - S_1(\boldsymbol{\theta}) + C$. The solver $\mathbb{D}(\cdot)$ takes 369 function evaluations to find the global maximum of $g_2(\boldsymbol{\theta})$ which is

$$\widehat{\boldsymbol{\theta}}_2 = (-6.002, -3.200), \tag{A.8}$$

where the value $\eta(\hat{\boldsymbol{\theta}}_2)$ is 1.1254. By setting the significance tolerance $\tilde{\delta} = 10^{-3}$ in Figure 4, the significance ratio δ in (3.37) is

$$\delta = \frac{\eta(\widehat{\boldsymbol{\theta}}_2)}{\eta(\widehat{\boldsymbol{\theta}}_1)} = 0.5001 > \tilde{\delta}. \tag{A.9}$$

Thus, the second mode around $\widehat{\boldsymbol{\theta}}_2$ is also a significant mode. Then we compute the Hessian matrix $H(\widehat{\boldsymbol{\theta}}_2)$. By taking the inverse, we obtain $H^{-1}(\widehat{\boldsymbol{\theta}}_2)$

$$H^{-1}(\widehat{\boldsymbol{\theta}}_2) = \begin{pmatrix} 0.200 & 0.480 \\ 0.480 & 1.252 \end{pmatrix}, \tag{A.10}$$

then the prior domain for $\widehat{\boldsymbol{\theta}}_2$ is calculated using (3.4) with $\beta = 5$, which is

$$\Gamma_{\widehat{\boldsymbol{\theta}}_2} = [-8.236, -3.763] \times [-8.794, 2.394], \tag{A.11}$$

whose volume is also much smaller than the searching domain Γ. Then we construct the second surrogate component $S_2(\boldsymbol{\theta})$ on $\Gamma_{\widehat{\boldsymbol{\theta}}_2}$ which requires 359 function evaluations. In Figure A1(c), we plot the plain view of the logarithm of $\eta(\boldsymbol{\theta})$ where the two black dots represent the peaks at $\widehat{\boldsymbol{\theta}}_1$ and

$\widehat{\boldsymbol{\theta}}_2$, and the two black boxes are their prior domains $\Gamma_{\widehat{\boldsymbol{\theta}}_1}$ and $\Gamma_{\widehat{\boldsymbol{\theta}}_2}$. Accordingly, in Figure A1(d), we plot the sparse grids built on $\Gamma_{\widehat{\boldsymbol{\theta}}_1}$ and $\Gamma_{\widehat{\boldsymbol{\theta}}_2}$.

Next, based on $S_1(\boldsymbol{\theta})$ and $S_2(\boldsymbol{\theta})$, we run the third iteration of the algorithm in Figure 4. The optimization solver $\mathbb{D}(\cdot)$ takes 217 function evaluations to find the global maximum of $g_3(\boldsymbol{\theta}) = \log(\eta(\boldsymbol{\theta})) - S_1(\boldsymbol{\theta}) - S_2(\boldsymbol{\theta}) + C$ which is

$$\widehat{\boldsymbol{\theta}}_3 = (-6.001, 4.999). \tag{A.12}$$

The value of $\eta(\widehat{\boldsymbol{\theta}}_3)$ is 3.183×10^{-4}, and the significance ratio δ in (3.37) is

$$\delta = \frac{\eta(\widehat{\boldsymbol{\theta}}_3)}{\eta(\widehat{\boldsymbol{\theta}}_1)} = 1.414 \times 10^{-4} < \tilde{\delta}. \tag{A.13}$$

Thus, the function $\eta(\boldsymbol{\theta})$ has two significant modes according to the selected value of the significance tolerance. Then the final expression of the surrogate system is $S(\boldsymbol{\theta}) = S_1(\boldsymbol{\theta}) + S_2(\boldsymbol{\theta}) - C$ and the two modes $M_3(\boldsymbol{\theta})$ and $M_4(\boldsymbol{\theta})$ are discarded.

Note that the significance tolerance $\tilde{\delta}$ can be set to any value according to the purpose of the simulation. For example, when the probability of a rare event is studied for risk assessment, then the two small modes, $M_3(\boldsymbol{\theta})$ and $M_4(\boldsymbol{\theta})$, become important in quantifying the uncertainty of the rare events. In this case, the tolerance $\tilde{\delta}$ can be set smaller, e.g. 10^{-5}, in order to capture $M_3(\boldsymbol{\theta})$ and $M_4(\boldsymbol{\theta})$ and the two additional components to the surrogate system.

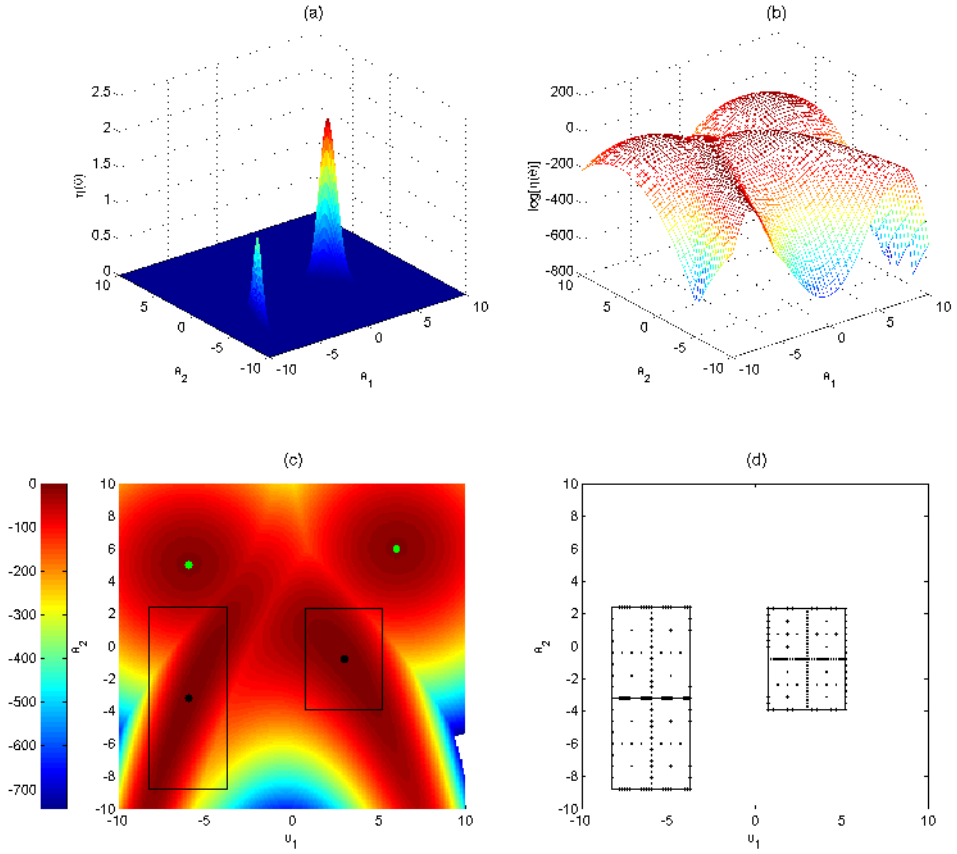

Figure A1: (a): Surface of the target function $\eta(\boldsymbol{\theta})$ where modes $M_3(\boldsymbol{\theta})$ and $M_4(\boldsymbol{\theta})$ are too small to be displayed. (b): Surface of $\log(\eta(\boldsymbol{\theta}))$ that has four modes displayed. (c): Plain view of $\log(\eta(\boldsymbol{\theta}))$. Since modes $M_1(\boldsymbol{\theta})$ and $M_2(\boldsymbol{\theta})$ (black dots) are more significant than $M_3(\boldsymbol{\theta})$ and $M_4(\boldsymbol{\theta})$ (green dots), our algorithm only build the surrogate system consisting of the two components for $M_1(\boldsymbol{\theta})$ and $M_2(\boldsymbol{\theta})$. The prior domains of the two components are marked by the black boxes around the two black dots. (d): The sparse grids for the two components of the surrogate system. The one for $M_1(\boldsymbol{\theta})$ has 341 points and the other for $M_2(\boldsymbol{\theta})$ has 145 points..

REFERENCES

[1] D. ALLAIRE AND K. WILLCOX, *Surrogate Modeling for Uncertainty Assessment with Application to Aviation Environmental System Models*, AIAA Journal, 48 (2010), pp. 1791–1803. 2

[2] V. BARTHELMANN, E. NOVAK, AND K. RITTER, *High dimensional polynomial interpolation on sparse grids*, Advances in Computational Mathematics, 12 (2000), pp. 273–288. 3, 7, 13

[3] J. BECK AND S. AU, *Bayesian updating of structural models and reliability using markov chain monte carlo simulation*, Journal of Engineering Mechanics, 128 (2002), pp. 380–391. 2

[4] G. BOX AND G. TIAO, *Bayesian inference in statistical analysis*, Wiley-Interscience, 1992. 2, 5

[5] H.-J. BUNGARTZ AND M. GRIEBEL, *Sparse grids*, Acta Numerica, 13 (2004), p. 147. 3, 7, 8, 13

[6] M. K. COWLES AND B. P. CARLIN, *Markov chain monte carlo convergence diagnostics: A comparative review*, Journal of the American Statistical Association, 91 (1996), pp. 883–904. 22

[7] G. CURTIS, *Documentation and applications of the reactive geochemical transport model RATEQ*, U.S. Geological Survey, Menlo Park, CA 94025, (2005). 25

[8] S. EZZEDINE, Y. RUBIN, AND J. CHEN, *Bayesian method for hydrogeological site characterization using borehole and geophysical survey data: Theory and application to the Lawrence Livermore National Laboratory Superfund site*, Water resources research, 35 (1999), pp. 2671–2683. 2

[9] D. GAMERMAN AND H. LOPES, *Markov Chain Monte Carlo: Stochastic Simulation for Bayesian Inference, 2nd edition*, Chapman and Hall/CRC, 2006. 2, 3, 5

[10] A. GELMAN, J. CARLIN, H. STERN, AND D. RUBIN, *Bayesian Data Analysis*, Chapman and Hall, London, 1995. 22

[11] M. GRIEBEL, *Adaptive sparse grid multilevel methods for elliptic PDEs based on finite differences*, Computing, 61 (1998), pp. 151–179. 3, 7

[12] M. GUNZBURGER, C. WEBSTER, AND G. ZHANG, *An adaptive wavelet stochastic collocation method for irregular solutions of partial differential equations with random input data*, submitted to SIAM/ASA Journal on Uncertainty Quantification, (2012). 13

[13] H. HAARIO, M. LAINE, A. MIRA, AND E. SAKSMAN, *DRAM: efficient adaptive MCMC*, Statistics and Computing, 16 (2006), pp. 339–354. 2

[14] D. R. JONES, C. D. PERTTUNEN, AND B. E. STUCKMAN, *Lipschitzian optimization without the lipschitz constant*, Journal of Optimization Theory and Applications, 79 (1993), pp. 157–181. 10.1007/BF00941892. 4, 6

[15] E. H. KEATING, J. DOHERTY, J. A. VRUGT, AND Q. KANG, *Optimization and uncertainty assessment of strongly nonlinear groundwater models with high parameter dimensionality*, Water Resources Research, 46 (2010), p. W10517. 2

[16] A. KLIMKE AND B. WOHLMUTH, *Algorithm 847: Spinterp: piecewise multilinear hierarchical sparse grid interpolation in MATLAB*, ACM Transactions on Mathematical Software (TOMS), 31 (2005), pp. 561–579. 7, 10, 13

[17] M. KOHLER, G. CURTIS, D. B. KENT, AND J. A. DAVIS, *Experimental investigation and modeling of uranium (VI) transport under variable chemical conditions*, Water Resources Research, 32 (1996), pp. 3539–3551. 18, 24, 28

[18] G. LIN AND A. 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. 3

[19] G. LIN, A. TARTAKOVSKY, AND D. TARTAKOVSKY, *Uncertainty quantification via random domain decomposition and probabilistic collocation on sparse grids*, Journal of Computational Physics, 229 (2010), pp. 6995 – 7012. 3

[20] X. MA AND N. ZABARAS, *An efficient Bayesian inference approach to inverse problems based on an adaptive sparse grid collocation method*, Inverse Problems, 25 (2009), p. 035013. 2, 3, 10

[21] L. MARSHALL, D. NOTT, AND A. SHARMA, *Hydrological model selection: A Bayesian alternative*, Water resources research, 41 (2005), pp. 1–11. 2

[22] Y. MARZOUK AND D. XIU, *A stochastic collocation approach to Bayesian inference in inverse problems*, Communications in Computational Physics, 6 (2009), pp. 826–847. 2

[23] Y. M. MARZOUK, H. N. NAJM, AND L. A. RAHN, *Stochastic spectral methods for efficient bayesian solution of inverse problems*, J. Comput. Phys., 224 (2007), pp. 560–586. 2

[24] F. NOBILE, R. TEMPONE, AND C. WEBSTER, *An anisotropic sparse grid stochastic collocation method for elliptic partial differential equations with random input data*, SIAM Journal on Numerical Analysis, 46 (2008), pp. 2411–2442. 3

[25] F. NOBILE, R. TEMPONE, AND C. G. WEBSTER, *A sparse grid stochastic collocation method for partial differential equations with random input data*, SIAM J. Numer. Anal., 46 (2008), pp. 2309–2345. 3, 13

[26] J. NOCEDAL AND S. WRIGHT, *Numerical optimization 2nd edition*, Springer, 2006. 6

[27] D. PFLÜGER, *Spatially Adaptive Sparse Grids for High-Dimensional Problems*, PhD thesis, 2005. 6

[28] H. PROMMER AND V. POST, *A reactive multicomponent transport model for saturated porous media*, http://www.pht3d.org, (2010). 19

[29] S. RAZAVI, B. A. TOLSON, AND D. H. BURN, *Review of surrogate modeling in water resources*, Water Resources Research, 48 (2012), p. W07401. 3

[30] Y. SUN, J. PETERSEN, T. CLEMENT, AND R. SKEEN, *Development of analytical solutions for multispecies transport with serial and parallel reactions*, Water Resources Research, 35 (1999), pp. 185–190. 18, 19

[31] J. VRUGT, C. TER BRAAK, C. DIKS, D. HIGDON, B. ROBINSON, AND J. HYMAN, *Accelerating Markov chain Monte Carlo simulation by differential evolution with self-adaptive randomized subspace sampling*, International Journal of Nonlinear Sciences and Numerical Simulation, 10 (2009), pp. 273–290. 2, 5

[32] J. A. VRUGT, C. J. F. TER BRAAK, M. P. CLARK, J. M. HYMAN, AND B. A. ROBINSON, *Treatment of input uncertainty in hydrologic modeling: Doing hydrology backward with Markov chain Monte Carlo simulation*, Water Resources Research, 44 (2008), pp. 1–15. 2, 5

[33] A. D. WOODBURY AND G. FERGUSON, *Ground surface paleotemperature reconstruction using information measures and empirical Bayes*, Geophysical Research Letters, 33 (2006), pp. 1–5. 2

[34] L. ZENG, L. SHI, D. ZHANG, AND L. WU, *A sparse grid based bayesian method for contaminant source identification*, Advances in Water Resources, 37 (2012), pp. 1 – 9. 2, 3

[35] F. ZHANG, G. YEH, AND J. PARKER, *Ground Water Reactive Transport Models*, Bentham Science Publishers, 2012. 2


OAK RIDGE NATIONAL LABORATORY
MANAGED BY UT-BATTELLE FOR THE DEPARTMENT OF ENERGY