

# Data-driven model reduction for the Bayesian solution of inverse problems

Tiangang Cui, Youssef M. Marzouk and Karen E. Willcox<sup>\*,†</sup>

*Department of Aeronautics and Astronautics, Massachusetts Institute of Technology, Cambridge, MA 02139, USA*

## SUMMARY

One of the major challenges in the Bayesian solution of inverse problems governed by partial differential equations (PDEs) is the computational cost of repeatedly evaluating numerical PDE models, as required by Markov chain Monte Carlo (MCMC) methods for posterior sampling. This paper proposes a data-driven projection-based model reduction technique to reduce this computational cost. The proposed technique has two distinctive features. First, the model reduction strategy is tailored to inverse problems: the snapshots used to construct the reduced-order model are computed adaptively from the posterior distribution. Posterior exploration and model reduction are thus pursued simultaneously. Second, to avoid repeated evaluations of the full-scale numerical model as in a standard MCMC method, we couple the full-scale model and the reduced-order model together in the MCMC algorithm. This maintains accurate inference while reducing its overall computational cost. In numerical experiments considering steady-state flow in a porous medium, the data-driven reduced-order model achieves better accuracy than a reduced-order model constructed using the classical approach. It also improves posterior sampling efficiency by several orders of magnitude compared with a standard MCMC method. Copyright © 2014 John Wiley & Sons, Ltd.

Received 3 March 2014; Revised 15 June 2014; Accepted 24 June 2014

KEY WORDS: model reduction; inverse problem; adaptive Markov chain Monte Carlo; approximate Bayesian inference

## 1. INTRODUCTION AND MOTIVATION

An important and challenging task in computational modeling is the solution of inverse problems, which convert noisy and indirect observational data into useful characterizations of the unknown parameters of a numerical model. In this process, statistical methods—Bayesian methods in particular—play a fundamental role in modeling various information sources and quantifying the uncertainty of the model parameters [1, 2]. In the Bayesian framework, the unknown parameters are modeled as random variables and hence can be characterized by their posterior distribution. Markov chain Monte Carlo (MCMC) methods [3] provide a powerful and flexible approach for sampling from posterior distributions. The Bayesian framework has been applied to inverse problems in various fields, for example, geothermal reservoir modeling [4], groundwater modeling [5], ocean dynamics [6], remote sensing [7], and seismic inversion [8].

To generate a sufficient number of samples from the posterior distribution, MCMC methods require sequential evaluations of the posterior probability density at many different points in the parameter space. Each evaluation of the posterior density involves a solution of the forward model used to define the likelihood function, which typically is a computationally intensive undertaking (e.g., the solution of a system of partial differential equations (PDEs)). In this many-query situation, one way to address the computational burden of evaluating the forward model is to replace it with

<sup>\*</sup>Correspondence to: Karen E. Willcox, Department of Aeronautics and Astronautics, Massachusetts Institute of Technology, Cambridge, MA 02139, USA.

<sup>†</sup>E-mail: kwillcox@mit.edu

a computationally efficient surrogate. Surrogate models have been applied to inverse problems in several settings; for example, in [9, 10], generalized polynomial chaos expansions are employed; in [11], Gaussian process models are employed; and in [12–15], projection-based reduced-order models are used. In this work, we also focus on projection-based reduced-order models (although the data-driven strategy underlying our approach should be applicable to other types of surrogate models). A projection-based reduced-order model reduces the computational complexity of the original or ‘full’ forward model by solving a projection of the full model onto a reduced subspace. For the model reduction methods we consider, the construction of the reduced subspace requires evaluating the full model at representative samples drawn from the parameter space. The solutions of the full model at these samples are referred to as snapshots [16]. Their span defines the reduced subspace, represented via an orthogonal basis.

The quality of the reduced-order model relies crucially on the choice of the samples for computing the snapshots. To construct reduced-order models targeting the Bayesian solution of the inverse problem, we employ existing projection-based model reduction techniques. Our innovation is in a data-driven approach that adaptively selects samples from the posterior distribution for the snapshot evaluations. This approach has two distinctive features:

1. We integrate the reduced-order model construction process into an adaptive MCMC algorithm that simultaneously samples the posterior and selects posterior samples for computing the snapshots. During the sampling process, the numerical accuracy of the reduced-order model is adaptively improved.
2. The approximate posterior distribution defined by the reduced-order model is used to increase the efficiency of MCMC sampling. We either couple the reduced-order model and the full model together to accelerate the sampling of the full posterior distribution<sup>‡</sup> or directly explore the approximate posterior distribution induced by the reduced-order model. In the latter case, sampling the approximate distribution yields a biased Monte Carlo estimator, but the bias can be controlled using error indicators or estimators.

Compared to the classical offline approaches that build the reduced-order model before using it in the many-query situation, the motivation for collecting snapshots during posterior exploration is to build a reduced-order model that focuses on a more concentrated region in the parameter space. Because the solution of the inverse problem is unknown until the data are available, reduced-order models built offline have to retain a level of numerical accuracy over a rather large region in the parameter space, which covers the support of the posterior distributions for all the possible observed data sets. For example, samples used for computing the snapshots are typically drawn from the prior distribution [13]. In comparison, our data-driven approach focuses only on the posterior distribution resulting from a particular data set. As the observed data necessarily increase the information divergence of the prior distribution from the posterior distribution [17], the support of the posterior distribution is more compact than that of the prior distribution.

Figure 1 illustrates a simple two-dimensional inference problem, where the prior distribution and the posterior distribution are represented by the blue and red contours, respectively. The left plot of Figure 1 shows 50 randomly drawn prior samples for computing the snapshots, each of which has a low probability of being in the support of the posterior. In comparison, the samples selected by our data-driven approach, as shown in the right plot of Figure 1, are scattered within the region of high posterior probability.

By retaining numerical accuracy only in a more concentrated region, the data-driven reduced-order model requires a basis of lower dimension to achieve the same level of accuracy compared with the reduced-order model built offline. For the same reason, the data-driven model reduction technique can potentially have better scalability with parameter dimension than the offline approach.

We note that goal-oriented model reduction approaches have been developed in the context of PDE-constrained optimization [18–21], in which the reduced-order model is simultaneously

<sup>‡</sup>The term ‘full posterior’ refers to the posterior distribution induced by the original or full forward model.

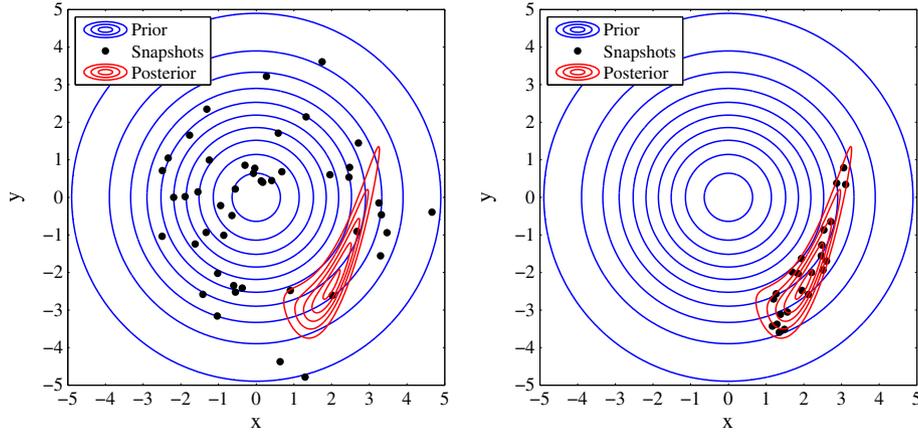


Figure 1. A two-dimensional example demonstrating data-driven model reduction. The prior distribution and posterior distribution are represented by the blue contours and red contours, respectively. The black dots represent samples used for computing the snapshots in the reduced-order model construction. Left: sampling using the classical approach (from the prior). Right: our data-driven approach.

constructed during the optimization process. In these methods, the snapshots for constructing the reduced basis are only evaluated at points in the parameter space that are close to the trajectory of the optimization algorithm.

The remainder of this paper is organized as follows. In Section 2, we outline the Bayesian framework for solving inverse problems and discuss the sampling efficiency and Monte Carlo error of MCMC. In Section 3, we introduce the data-driven model reduction approach and construct the data-driven reduced-order model within an adaptive delayed acceptance algorithm to speed up MCMC sampling. We also provide results on the ergodicity of the algorithm. Section 4 analyzes some properties of the data-driven reduced-order model. In Section 5, we discuss a modified framework that adaptively constructs the reduced-order model and simultaneously explores the induced approximate posterior distribution. We also provide an analysis of the mean square error of the resulting Monte Carlo estimator. In Section 6, we demonstrate and discuss various aspects of our methods through numerical examples. Section 7 offers concluding remarks.

## 2. SAMPLE-BASED INFERENCE FOR INVERSE PROBLEMS

The first part of this section provides a brief overview of the Bayesian framework for inverse problems. Further details can be found in [1, 2, 22]. The second part of this section discusses the efficiency of MCMC sampling for computationally intensive inverse problems.

### 2.1. Posterior formulation and sampling

Given a physical system, let  $\mathbf{x} \in \mathbb{X} \subset \mathbb{R}^{N_p}$  denote the  $N_p$ -dimensional unknown parameter, and  $\mathbf{d}_{\text{obs}} \in \mathbb{D} \subset \mathbb{R}^{N_d}$  denote the  $N_d$ -dimensional observed data. The forward model  $\mathbf{d} = F(\mathbf{x})$  maps a given parameter  $\mathbf{x}$  to the observable model outputs  $\mathbf{d}$ .

In a Bayesian setting, the first task is to construct the prior models and the likelihood function as probability distributions. The prior density is a stochastic model representing knowledge of the unknown  $\mathbf{x}$  before any measurements and is denoted by  $\pi_0(\mathbf{x})$ . The likelihood function specifies the probability density of the observation  $\mathbf{d}_{\text{obs}}$  for a given set of parameters  $\mathbf{x}$ , denoted by  $L(\mathbf{d}_{\text{obs}} | \mathbf{x})$ . We assume that the data and the model parameters follow the stochastic relationship

$$\mathbf{d}_{\text{obs}} = F(\mathbf{x}) + \mathbf{e}, \quad (1)$$

where the random vector  $\mathbf{e}$  captures the measurement noise and other uncertainties in the observation-model relationship, including model errors. Without additional knowledge of the

measurement process and model errors,  $\mathbf{e}$  is modeled as a zero mean Gaussian,  $\mathbf{e} \sim \mathcal{N}(0, \Sigma_e)$ , where  $\Sigma_e$  is the covariance. Let

$$\Phi(\mathbf{x}) = \frac{1}{2} \left\| \Sigma_e^{-\frac{1}{2}} (F(\mathbf{x}) - \mathbf{d}_{\text{obs}}) \right\|^2 \tag{2}$$

denote the data-misfit function. The resulting likelihood function is proportional to  $\exp(-\Phi(\mathbf{x}))$ . By Bayes' formula, the posterior probability density is

$$\pi(\mathbf{x}|\mathbf{d}_{\text{obs}}) = \frac{1}{Z} \exp(-\Phi(\mathbf{x}))\pi_0(\mathbf{x}), \tag{3}$$

where

$$Z = \int_{\mathbb{X}} \exp(-\Phi(\mathbf{x}))\pi_0(\mathbf{x})d\mathbf{x}. \tag{4}$$

is the normalizing constant.

In the general framework setup by [23], we can explore the posterior distribution given by (3) using MCMC methods such as the Metropolis–Hastings algorithm [24, 25]. The Metropolis–Hastings algorithm uses a proposal distribution and an acceptance/rejection step to construct the transition kernel of a Markov chain that converges to the desired target distribution.

### 2.2. Sampling efficiency and Monte Carlo error

Once we draw a sufficient number of samples from the posterior distribution, the expectations of functions over the posterior distribution can be estimated by Monte Carlo integration. Suppose we wish to estimate the expectation of a function  $h(x)$  over the posterior distribution

$$I(h) = \int_{\mathbb{X}} h(\mathbf{x})\pi(\mathbf{x}|\mathbf{d}_{\text{obs}})d\mathbf{x}, \tag{5}$$

by  $N$  posterior samples,  $\mathbf{x}^{(i)} \sim \pi(\mathbf{x}|\mathbf{d}_{\text{obs}}), i = 1, \dots, N$ . The resulting Monte Carlo estimator of  $I(h)$  is

$$\widehat{I(h)} = \frac{1}{N} \sum_{i=1}^N h(\mathbf{x}^{(i)}), \tag{6}$$

which is an unbiased estimator with the mean square error

$$\text{MSE}(\widehat{I(h)}) = \text{Var}(\widehat{I(h)}) = \frac{\text{Var}(h)}{\text{ESS}(h)}. \tag{7}$$

Because the samples drawn by an MCMC algorithm are correlated, the variance of  $\widehat{I(h)}$  is dependent on the effective sample size (ESS)

$$\text{ESS}(h) = \frac{N}{2 \times \text{IACT}(h)}, \tag{8}$$

where

$$\text{IACT}(h) = \frac{1}{2} + \sum_{j=1}^{\infty} \text{corr}(h(\mathbf{x}^{(1)}), h(\mathbf{x}^{(j+1)}))$$

is the integrated autocorrelation of  $h(x)$ ; see [3, Chapter 5.8] for a detailed derivation and further references.

In practice, we wish to improve the *computational efficiency* of the MCMC algorithm, which is defined by the ESS for a given budget of CPU time. There are two ways to achieve this:

1. Improve the *statistical efficiency*. To increase the ESS for a given number of MCMC iterations, one can reduce the sample correlation by designing proposals that traverse the parameter space more efficiently. For example, adaptive MCMC methods [26, 27] use adaptation to learn the covariance structure of the posterior distribution online and use this information to design proposal distributions. Algorithms such as stochastic Newton [8] and Riemannian manifold MCMC [28] use local derivative information to construct efficient proposals that adapt to the local geometry of the posterior distribution.
2. Increase the number of MCMC steps for a given amount of CPU time. Because simulating the forward model in the data-misfit function (2) is CPU intensive, computing the posterior density at every iteration is the major bottleneck of the standard MCMC algorithm. By using a fast approximation to the forward model, the amount of computing time can be reduced. This effort complements the use of MCMC algorithms that require local derivatives, as fast approximations of the forward model also enable fast evaluations of its gradient and even higher-order derivatives.<sup>§</sup>

One important remark is that by drawing samples directly from an approximate posterior distribution (i.e., the one induced by our approximation of the forward model), the resulting Monte Carlo estimator (6) is *biased*. MCMC algorithms such as surrogate transition [29] or delayed acceptance [30] can be used to couple the posterior distribution and its approximation together to ensure the samples are drawn from the full posterior distribution. On the other hand, if the accuracy of the posterior approximation can be tightly controlled, then some bias may be acceptable if it enables significant variance reduction, and thus an overall reduction in the mean squared error (MSE) of an estimate of a posterior expectation, for a given computational effort. We explore both options in this work.

### 3. DATA-DRIVEN REDUCED-ORDER MODEL AND FULL TARGET ALGORITHM

This section introduces our data-driven model reduction approach, and the adaptive sampling framework for simultaneously constructing the reduced-order model and exploring the posterior distribution.

#### 3.1. Posterior approximation

Suppose the system of interest can be described by a system of nonlinear steady PDEs, with a finite element or finite difference discretization that results in a discretized system in the form of

$$\mathbf{A}(\mathbf{x})\mathbf{u} + \mathbf{f}(\mathbf{x}, \mathbf{u}) + \mathbf{q}(\mathbf{x}) = 0. \quad (9)$$

In Equation (9),  $\mathbf{u} \in \mathbb{R}^n$  represents the discretized state of the system,  $n$  is the dimension of the system (the number of unknowns),  $\mathbf{A}(\mathbf{x}) \in \mathbb{R}^{n \times n}$  is a discretized linear operator,  $\mathbf{f}(\mathbf{x}, \mathbf{u}) \in \mathbb{R}^n$  represents the discretized nonlinear terms of the governing PDE, and  $\mathbf{q}(\mathbf{x})$  denotes the forcing terms. All of  $\mathbf{A}$ ,  $\mathbf{f}$ , and  $\mathbf{q}$  can be parameterized by the unknown parameter  $\mathbf{x}$ . An observation operator  $\mathbf{C}$  maps the state of the system to the observable model outputs, that is,

$$\mathbf{d} = \mathbf{C}(\mathbf{u}, \mathbf{x}). \quad (10)$$

Equations (9) and (10) define the *forward model*,  $\mathbf{d} = F(\mathbf{x})$ , that maps a realization of the unknown parameter  $\mathbf{x}$  to observable model outputs  $\mathbf{d}$ .

<sup>§</sup>Using approximate derivatives does not impact the ergodicity of the MCMC algorithm, as the bias caused by the approximate derivatives is corrected by the acceptance/rejection step in the Metropolis–Hastings algorithm.

Our data-driven model reduction approach selects a set of  $m$  posterior samples  $\{\mathbf{x}^{(0)}, \dots, \mathbf{x}^{(m)}\}$  to compute the snapshots  $\{\mathbf{u}^{(0)}, \dots, \mathbf{u}^{(m)}\}$  by solving Equation (9) for each sample  $\mathbf{x}^{(j)}$ . By constructing the reduced basis  $\mathbf{V}_m = \text{span}\{\mathbf{u}^{(0)}, \dots, \mathbf{u}^{(m)}\}$ , the state  $\mathbf{u}$  can be approximated by a linear combination of the reduced basis vectors, that is,  $\mathbf{u} \approx \mathbf{V}_m \mathbf{u}_m$ . Then, Equation (9) can be approximated by applying the Galerkin projection:

$$\underbrace{\mathbf{V}_m^T \mathbf{A}(\mathbf{x}) \mathbf{V}_m}_{\mathbf{A}_m(\mathbf{x})} \mathbf{u}_m + \mathbf{V}_m^T \mathbf{f}(\mathbf{x}, \mathbf{V}_m \mathbf{u}_m) + \underbrace{\mathbf{V}_m^T \mathbf{q}(\mathbf{x})}_{\mathbf{q}_m(\mathbf{x})} = 0, \quad (11)$$

and the associated model outputs are

$$\mathbf{d}_m = \mathbf{C}(\mathbf{V}_m \mathbf{u}_m, \mathbf{x}). \quad (12)$$

If  $\dim(\mathbf{V}_m) \ll n$ , the dimension of the unknown state in (11) and (12) is greatly reduced compared with that of the original full system (9) and (10). Thus, (11) and (12) define a reduced-order model  $\mathbf{d}_m = F_m(\mathbf{x})$  that maps a realization of the unknown parameter  $\mathbf{x}$  to an approximation of the observable model outputs  $\mathbf{d}_m$ .

Care must be taken to ensure efficient solution of the reduced-order model, as in the presence of general parametric dependence, Equations (11) and (12) have low state dimension but are not necessarily fast to solve. This is because for each new parameter  $\mathbf{x}$  solution of the reduced-order model requires evaluating the full scale system matrices or residual, projecting those matrices or residual onto the reduced basis, and then solving the resulting low dimensional system. As many elements of these computations depend on  $n$ , the dimension of the original system, this process is typically computationally expensive (unless there is special structure to be exploited, such as affine parametric dependence). To reduce the computational cost of this process, methods such as the missing point estimation [31], the empirical interpolation [32], and its discrete variant [33] approximate the nonlinear term in the reduced-order model by selective spatial sampling.

We note that for systems that exhibit a wide range of behaviors, recently developed localization approaches such as [34–37] adaptively construct multiple local reduced bases, each tailored to a particular system behavior that is associated with a subdomain of the state space or the parameter space. Our use of adaptation is different, focusing on the adaptive selection of posterior samples for evaluating the snapshots. In this paper, we consider only the case of a global reduced basis; however, future work could combine our adaptive posterior sampling approach with a localization strategy to build multiple local reduced bases, each adapted to the local structure of the posterior.

The data-misfit function (2) can be approximated by replacing the forward model  $F(\cdot)$  with the reduced-order model  $F_m(\cdot)$ , which gives

$$\Phi_m(\mathbf{x}) = \frac{1}{2} \left\| \Sigma_e^{-\frac{1}{2}} (F_m(\mathbf{x}) - \mathbf{d}_{\text{obs}}) \right\|^2. \quad (13)$$

Then, the resulting approximate posterior distribution has the form

$$\pi_m(\mathbf{x} | \mathbf{d}_{\text{obs}}) = \frac{1}{Z_m} \exp(-\Phi_m(\mathbf{x})) \pi_0(\mathbf{x}), \quad (14)$$

where  $Z_m = \int_{\mathbb{X}} \exp(-\Phi_m(\mathbf{x})) \pi_0(\mathbf{x}) d\mathbf{x}$  is the normalizing constant.

Our data-driven approach adaptively selects the sample for evaluating the next snapshot such that the scaled error of the reduced-order model outputs,

$$\mathbf{t}_m(\mathbf{x}) = \Sigma_e^{-\frac{1}{2}} (F(\mathbf{x}) - F_m(\mathbf{x})), \quad (15)$$

for the current reduced basis  $\mathbf{V}_m$ , is above a user-specified threshold. The whitening transformation  $\Sigma_e^{-\frac{1}{2}}$  computes the relative error of the reduced-order model compared with noise level of the observed data, which has standard deviation 1 after the transformation. In case that we wish to

bypass the forward model evaluations, so that the error  $\mathbf{t}_m(\mathbf{x})$  cannot be computed directly, an error indicator or an error estimator,  $\hat{\mathbf{t}}_m(\mathbf{x}) \approx \mathbf{t}_m(\mathbf{x})$ , can be used. In this work, we use the dual weighted residual technique [38] to compute an error indicator.

### 3.2. Full target algorithm

To achieve simultaneous model reduction and posterior exploration, we employ the adaptive delayed acceptance algorithm of [39]. Suppose we have a reduced-order model constructed from an initial reduced basis. At each iteration of the MCMC sampling, we first sample the approximate posterior distribution based on the reduced-order model for a certain number of steps using a standard Metropolis–Hastings algorithm. This first-stage subchain simulation should have sufficient number of steps, so that its initial state and last state are uncorrelated. Then, the last state of the subchain is used as a proposal candidate in the second-stage of the algorithm, and the acceptance probability for this candidate is computed based on the ratio of the full posterior density value to the approximate posterior density value. This delayed acceptance algorithm employs the first-stage subchain to decrease the sample correlation by sampling the computationally fast approximate posterior and then uses the acceptance/rejection in the second-stage to ensure that the algorithm correctly samples the full posterior distribution.

The statistical efficiency of the delayed acceptance algorithm relies on the accuracy of the approximate posterior. An approximate posterior induced by an inaccurate reduced-order model can potentially result in a higher second-stage rejection rate, which decreases the statistical efficiency of the delayed acceptance algorithm. This is because duplicated MCMC samples generated by rejections increase the sample correlation (we refer to [30] for a formal justification). To maintain statistical efficiency, we aim to construct a sufficiently accurate reduced-order model, so that the resulting second-stage acceptance probability is close to 1. To achieve this, we introduce adaptive reduced basis enrichment into the delayed acceptance algorithm.

After each full posterior density evaluation, the state of the associated forward model evaluation can be used as a potential new snapshot. We compute the scaled error (15) of the reduced-order model outputs at each new posterior sample, and the reduced basis is updated with the new snapshot when the error exceeds a user-given threshold  $\epsilon$ . By choosing an appropriate threshold  $\epsilon$ , we can control the maximum allowable amount of error of the reduced-order model.

The resulting reduced-order model is data driven, as it uses the information provided by the observed data (in the form of the posterior distribution) to select samples for computing the snapshots. It is also an online model reduction approach, as the reduced-order model is built concurrently with posterior sampling.

*3.2.1. The algorithm.* As the adaptive sampling procedure samples from the full posterior distribution, hereafter, we refer to it as the ‘full target algorithm’. Details of the full target algorithm are given in Algorithm 1.

Lines 1–13 of Algorithm 1 simulate a Markov chain with invariant distribution  $\pi_m(\mathbf{y}|\mathbf{d}_{\text{obs}})$  for  $L$  steps. In lines 14–19 of Algorithm 1, to ensure that the algorithm samples from the full posterior distribution, we post-process the last state of the subchain using a second-stage acceptance/rejection based on the full posterior distribution.

The second-stage acceptance probability  $\beta$  is controlled by the accuracy of the reduced-order model and the subchain length  $L$ . Ideally, we want to have large  $L$  to give uncorrelated samples. However, if the second-stage acceptance probability  $\beta$  is low, the effort spent on simulating the subchain will be wasted because the proposal is more likely to be rejected in the second step. To avoid this situation, we dynamically monitor the accuracy of the reduced-order model by evaluating its error indicator at each state of the subchain (lines 10–12). The subchain is terminated if the  $L$ -infinity norm of the scaled error indicator exceeds a threshold  $\epsilon$ .

Lines 20–22 of Algorithm 1 describe the adaptation of the reduced basis, which is controlled by the scaled error (15) of the reduced-order model and a given threshold  $\epsilon$ . Three criteria must be satisfied for the reduced basis to be updated: (i) the scaled error at  $X_{n+1}$  must exceed the threshold  $\epsilon$ ; (ii) the dimensionality of the reduced basis must not exceed the maximum allowable dimensionality  $M$ ;

**Algorithm 1** Full target algorithm

**Require:** Given the subchain length  $L$ , the maximum allowable reduced basis dimension  $M$ , and the error threshold  $\epsilon$ . At step  $n$ , given state  $X_n = \mathbf{x}$ , a proposal distribution  $q(\mathbf{x}, \cdot)$ , and a reduced-order model  $F_m(\cdot)$  defined by the reduced basis  $\mathbf{V}_m$ , one step of the algorithm is:

- 1: Set  $Y_1 = X_n, \mathbf{y} = \mathbf{x}$ , and  $i = 1$
- 2: **while**  $i \leq L$  **do**
- 3: Propose a candidate  $\mathbf{y}' \sim q(\mathbf{y}, \cdot)$ , then evaluate the acceptance probability

$$\alpha(\mathbf{y}, \mathbf{y}') = 1 \wedge \frac{\pi_m(\mathbf{y}'|\mathbf{d}_{\text{obs}}) q(\mathbf{y}', \mathbf{y})}{\pi_m(\mathbf{y}|\mathbf{d}_{\text{obs}}) q(\mathbf{y}, \mathbf{y}')}$$

- 4: **if**  $\text{Uniform}(0, 1] < \alpha(\mathbf{y}, \mathbf{y}')$  **then**
- 5:     Accept  $\mathbf{y}'$  by setting  $Y_{i+1} = \mathbf{y}'$
- 6: **else**
- 7:     Reject  $\mathbf{y}'$  by setting  $Y_{i+1} = \mathbf{y}$
- 8: **end if**
- 9:      $i = i + 1$
- 10: **if** finite adaptation criterion not satisfied **and**  $\|\hat{\mathbf{t}}_m(X_{n+1})\|_{\infty} \geq \epsilon$  **then**
- 11:     **break**
- 12: **end if**
- 13: **end while**
- 14: Set  $\mathbf{x}' = Y_i$ , then evaluate the acceptance probability using the full posterior

$$\beta(\mathbf{x}, \mathbf{x}') = 1 \wedge \frac{\pi(\mathbf{x}'|\mathbf{d}_{\text{obs}}) \pi_m(\mathbf{x}|\mathbf{d}_{\text{obs}})}{\pi(\mathbf{x}|\mathbf{d}_{\text{obs}}) \pi_m(\mathbf{x}'|\mathbf{d}_{\text{obs}})}$$

- 15: **if**  $\text{Uniform}(0, 1] < \beta(\mathbf{x}, \mathbf{x}')$  **then**
- 16:     Accept  $\mathbf{x}'$  by setting  $X_{n+1} = \mathbf{x}'$
- 17: **else**
- 18:     Reject  $\mathbf{x}'$  by setting  $X_{n+1} = \mathbf{x}$
- 19: **end if**
- 20: **if** finite adaptation criterion not satisfied **and**  $m < M$  **and**  $\|\mathbf{t}_m(X_{n+1})\|_{\infty} \geq \epsilon$  **then**
- 21:     Update the reduced basis  $\mathbf{V}_m$  to  $\mathbf{V}_{m+1}$  using the new full model evaluation at  $\mathbf{x}'$  by a Gram-Schmidt process
- 22: **end if**

and (iii) the finite adaptation criterion (Definition 1) should not yet be satisfied. The finite adaptation criterion is defined precisely as follows.

*Definition 1*

Finite adaptation criterion. The average number of MCMC steps used for each reduced basis enrichment exceeds  $N_{\text{max}} = \frac{1}{c\epsilon}$ , for some user-specified  $c > 0$ , where  $\epsilon$  is the error threshold used in Algorithm 1.

The finite adaptation criterion is a threshold for how *infrequent* updates to the reduced-order model should become before model adaptation is stopped entirely. When more and more MCMC steps are used between updates, the reduced-order model has satisfied the accuracy threshold  $\epsilon$  over larger and larger regions of the posterior. The threshold of at least  $N_{\text{max}}$  steps between updates can thus be understood as a measure of ‘sufficient coverage’ by the reduced-order model. Further discussion and justification of the finite adaptation criterion are deferred to Section 4.

Algorithm 1 uses the adaptation in lines 20–22 to drive the online construction of the reduced-order model. Once the adaptation stops, the algorithm runs as a standard delayed acceptance algorithm, with an  $L$ -step subchain in the first stage.

To prevent the situation where the reduced basis dimension is large so that the reduced-order model becomes computationally inefficient, we limit the reduced basis dimension to be less than a user-given threshold  $M$ . When the reduced basis dimension reaches  $M$  but the finite adaptation criterion is not satisfied, one strategy is to stop the reduced basis enrichment and continue simulating Algorithm 1. This does not affect the ergodicity of Algorithm 1, as will be discussed in Section 3.2.2. However, the error of the resulting reduced-order model can be large in the subregion of the support of the posterior that remains unexplored. Consequently, this large reduced-order model error may decrease the second-stage acceptance rate and the statistical efficiency of the algorithm.

A wide range of proposal distributions can be used in Algorithm 1. In this work, we use the grouped-components adaptive Metropolis [39], which is a variant of adaptive Metropolis [26]. More advanced algorithms such as stochastic Newton [8] or manifold MCMC [28] can also be used within Algorithm 1. The computational cost of evaluating derivative information for these algorithms can also be reduced by using the reduced-order models.

*3.2.2. Ergodicity of the full target algorithm.* Throughout this work, the following assumption on the forward model and the reduced-order model is used:

*Assumption 2*

The forward model  $F(\mathbf{x})$  and the reduced-order model  $F_m(\mathbf{x})$  are Lipschitz continuous and bounded on  $\mathbb{X}$ .

As the data-misfit functions  $\Phi(\mathbf{x})$  and  $\Phi_m(\mathbf{x})$  are quadratic, we have  $\Phi(\mathbf{x}) \geq 0$  and  $\Phi_m(\mathbf{x}) \geq 0$ . Assumption 2 implies that there exists a constant  $K > 0$  such that  $\Phi(\mathbf{x}) \leq K$  and  $\Phi_m(\mathbf{x}) \leq K$ ,  $\forall \mathbf{x} \in \mathbb{X}$ . We also note that Lipschitz continuity of  $F(\mathbf{x})$  implies the Lipschitz continuity of  $\Phi(\mathbf{x})$ . Similarly,  $\Phi_m(\mathbf{x})$  is Lipschitz continuous.

We first establish the ergodicity of a non-adaptive version of Algorithm 1, where the reduced-order model is assumed to be given and fixed.

*Lemma 3*

In the first stage of Algorithm 1, suppose the proposal distribution  $q(\mathbf{x}, \mathbf{y})$  is  $\pi$  irreducible. Then, the non-adaptive version of Algorithm 1 is ergodic.

*Proof*

The detailed balance condition and aperiodicity condition are satisfied by Algorithm 1; see [29, 30]. As  $\exp(-\Phi_m(\mathbf{y})) > 0$  for all  $\mathbf{y} \in \mathbb{X}$  by Assumption 2, we have  $q(\mathbf{x}, \mathbf{y}) > 0$  for all  $\mathbf{x}, \mathbf{y} \in \mathbb{X}$ ; thus, the irreducibility condition is satisfied; see [30]. Ergodicity follows from the detailed balance, aperiodicity, and irreducibility.  $\square$

The adaptation used in Algorithm 1 is different from that of standard adaptive MCMC algorithms such as [26] and the original adaptive delayed acceptance algorithm [39]. These algorithms carry out an infinite number of adaptations because modifications to the proposal distribution or to the approximation continue throughout MCMC sampling. The number of adaptations in Algorithm 1, on the other hand, cannot exceed the maximum allowable dimension  $M$  of the reduced basis, and hence, it is finite. Furthermore, the adaptation stops after finite time because of the finite adaptation criterion. As Algorithm 1 is ergodic for each of the reduced-order models constructed by Lemma 3, Proposition 2 in [27] ensures the ergodicity of Algorithm 1. Lemma 3 also reveals that Algorithm 1 always converges to the full posterior distribution, regardless of the accuracy of the reduced-order model.

#### 4. PROPERTIES OF THE APPROXIMATION

In Algorithm 1, the accuracy of the reduced-order model is adaptively improved during MCMC sampling, and thus, it is important to analyze the error of the approximate posterior induced by the reduced-order model, compared with the full posterior. As bounds on the bias of the resulting

Monte Carlo estimator can be derived from the error of the approximate posterior, this analysis is particularly useful in situations where we want to utilize the approximate posterior directly for further CPU time reduction. This error analysis also justifies the finite adaptation criterion (Definition 1) used in Algorithm 1 to terminate the enrichment of the reduced basis.

We provide here an analysis of the Hellinger distance between the full posterior distribution and the approximate posterior distribution

$$d_{\text{Hell}}(\pi, \pi_m) = \left( \frac{1}{2} \int_{\mathbb{X}} \left( \sqrt{\pi(\mathbf{x}|\mathbf{d}_{\text{obs}})} - \sqrt{\pi_m(\mathbf{x}|\mathbf{d}_{\text{obs}})} \right)^2 d\mathbf{x} \right)^{\frac{1}{2}}. \quad (16)$$

The Hellinger distance translates directly into bounds on expectations, and hence, we use it as a metric to quantify the error of the approximate posterior distribution.

The framework set by [22] is adapted here to analyze the Hellinger distance between the full posterior distribution and its approximation induced by the  $F_m$  constructed in Algorithm 1, with respect to a given threshold  $\epsilon$ . Given a set of samples  $\{\mathbf{x}^{(0)}, \dots, \bar{\mathbf{x}}^{(m)}\}$  where the snapshots are computed, and the associated reduced basis  $\mathbf{V}_m$ , we define the  $\epsilon$ -feasible set and the associated posterior measure.

*Definition 4*

For a given  $\epsilon > 0$  and a reduced-order model  $F_m(\mathbf{x})$ , define the  $\epsilon$ -feasible set as

$$\Omega^{(m)}(\epsilon) = \left\{ \mathbf{x} \in \mathbb{X} \mid \left\| \Sigma_e^{-\frac{1}{2}} (F(\mathbf{x}) - F_m(\mathbf{x})) \right\|_{\infty} \leq \epsilon \right\}. \quad (17)$$

The set  $\Omega^{(m)}(\epsilon) \subseteq \mathbb{X}$  has posterior measure

$$\mu \left( \Omega^{(m)}(\epsilon) \right) = \int_{\Omega^{(m)}(\epsilon)} \pi(\mathbf{x}|\mathbf{d}_{\text{obs}}) d\mathbf{x}. \quad (18)$$

The complement of the  $\epsilon$ -feasible set is given by  $\Omega_{\perp}^{(m)}(\epsilon) = \mathbb{X} \setminus \Omega^{(m)}(\epsilon)$ , which has posterior measure  $\mu \left( \Omega_{\perp}^{(m)}(\epsilon) \right) = 1 - \mu(\Omega^{(m)}(\epsilon))$ .

On any measurable subset of the  $\epsilon$ -feasible set, the error of the reduced-order model is bounded above by  $\epsilon$ . Then, the Hellinger distance (16) can be bounded by the user-specified  $\epsilon$  and the posterior measure of  $\Omega_{\perp}^{(m)}(\epsilon)$ , which is the region that has not been well approximated by the reduced-order model. We formalize this notion through the following propositions and theorems.

*Proposition 5*

Given a reduced-order model  $F_m(\mathbf{x})$  and some  $\epsilon > 0$ , there exists a constant  $K > 0$  such that  $|\Phi(\mathbf{x}) - \Phi_m(\mathbf{x})| \leq K\epsilon, \forall \mathbf{x} \in \Omega^{(m)}(\epsilon)$ .

*Proof*

This result directly follows from the definition of the  $\epsilon$ -feasible set and Assumption 2. □

*Proposition 6*

Given a reduced-order model  $F_m(\mathbf{x})$  and a  $\epsilon > 0$ , there exist constants  $K_1 > 0$  and  $K_2 > 0$  such that  $|Z - Z_m| \leq K_1\epsilon + K_2\mu \left( \Omega_{\perp}^{(m)}(\epsilon) \right)$ .

*Proof*

From the definition of the  $\epsilon$ -feasible set, we have a bound on the difference of the normalizing constants:

$$\begin{aligned} |Z - Z_m| &\leq \int_{\Omega^{(m)}(\epsilon)} |\exp(-\Phi(\mathbf{x})) - \exp(-\Phi_m(\mathbf{x}))| \pi_0(\mathbf{x}) d\mathbf{x} \\ &\quad + \int_{\Omega_{\perp}^{(m)}(\epsilon)} |1 - \exp(\Phi(\mathbf{x}) - \Phi_m(\mathbf{x}))| Z \pi(\mathbf{x}|\mathbf{d}_{\text{obs}}) d\mathbf{x}. \end{aligned}$$

As  $\Phi$  and  $\Phi_m$  are Lipschitz continuous and greater than zero, we have

$$|\exp(-\Phi(\mathbf{x})) - \exp(-\Phi_m(\mathbf{x}))| \leq K_3 |\Phi(\mathbf{x}) - \Phi_m(\mathbf{x})|,$$

for some constant  $K_3 > 0$ . Then, by Proposition 5, there exists a constant  $K_1 > 0$  such that

$$\int_{\Omega^{(m)}(\epsilon)} |\exp(-\Phi(\mathbf{x})) - \exp(-\Phi_m(\mathbf{x}))| \pi_0(\mathbf{x}) d\mathbf{x} \leq K_1 \epsilon.$$

There also exists a constant  $K_2 > 0$  such that

$$\begin{aligned} \int_{\Omega_{\perp}^{(m)}(\epsilon)} |1 - \exp(\Phi(\mathbf{x}) - \Phi_m(\mathbf{x}))| Z \pi(\mathbf{x}|\mathbf{d}_{\text{obs}}) d\mathbf{x} &\leq K_2 \int_{\Omega_{\perp}^{(m)}(\epsilon)} \pi(\mathbf{x}|\mathbf{d}_{\text{obs}}) d\mathbf{x} \\ &= K_2 \mu(\Omega_{\perp}^{(m)}(\epsilon)). \end{aligned}$$

Thus, we have  $|Z - Z_m| \leq K_1 \epsilon + K_2 \mu(\Omega_{\perp}^{(m)}(\epsilon))$ . □

*Theorem 7*

Suppose we have the full posterior distribution  $\pi(\mathbf{x}|\mathbf{d}_{\text{obs}})$  and its approximation  $\pi_m(\mathbf{x}|\mathbf{d}_{\text{obs}})$  induced by a reduced-order model  $F_m(\mathbf{x})$ . For a given  $\epsilon > 0$ , there exist constants  $K_1 > 0$  and  $K_2 > 0$  such that

$$d_{\text{Hell}}(\pi, \pi_m) \leq K_1 \epsilon + K_2 \mu(\Omega_{\perp}^{(m)}(\epsilon)). \tag{19}$$

*Proof*

Following Theorem 4.6 of [22], we have

$$\begin{aligned} d_{\text{Hell}}(\pi, \pi_m)^2 &= \frac{1}{2} \int_{\mathbb{X}} \left( \sqrt{\frac{1}{Z} \exp(-\Phi(\mathbf{x}))} - \sqrt{\frac{1}{Z_m} \exp(-\Phi_m(\mathbf{x}))} \right)^2 \pi_0(\mathbf{x}) d\mathbf{x} \\ &\leq I_1 + I_2, \end{aligned}$$

where

$$\begin{aligned} I_1 &= \frac{1}{Z} \int_{\mathbb{X}} \left( \exp\left(-\frac{1}{2}\Phi(\mathbf{x})\right) - \exp\left(-\frac{1}{2}\Phi_m(\mathbf{x})\right) \right)^2 \pi_0(\mathbf{x}) d\mathbf{x}, \\ I_2 &= \left| Z^{-\frac{1}{2}} - Z_m^{-\frac{1}{2}} \right|^2 Z_m. \end{aligned}$$

Following the same derivation as Proposition 6, we have

$$\begin{aligned} I_1 &\leq \frac{1}{Z} \int_{\Omega^{(m)}} \left( \exp\left(-\frac{1}{2}\Phi(\mathbf{x})\right) - \exp\left(-\frac{1}{2}\Phi_m(\mathbf{x})\right) \right)^2 \pi_0(\mathbf{x}) d\mathbf{x} \\ &\quad + \int_{\Omega_{\perp}^{(m)}} \left( 1 - \exp\left(\frac{1}{2}\Phi(\mathbf{x}) - \frac{1}{2}\Phi_m(\mathbf{x})\right) \right)^2 \pi(\mathbf{x}|\mathbf{d}_{\text{obs}}) d\mathbf{x}. \end{aligned}$$

Thus, there exist constants  $K_3, K_4 > 0$  such that  $I_1 \leq K_3 \epsilon^2 + K_4 \mu(\Omega_{\perp}^{(m)}(\epsilon))$ .

Applying the bound

$$\left| Z^{-\frac{1}{2}} - Z_m^{-\frac{1}{2}} \right|^2 \leq K \max(Z^{-3}, (Z_m)^{-3}) |Z - Z_m|^2,$$

and Proposition 6, we have

$$I_2 \leq \left( K_5 \epsilon + K_6 \mu \left( \Omega_{\perp}^{(m)}(\epsilon) \right) \right)^2$$

for some constants  $K_5, K_6 > 0$ .

Combining the preceding results, we have

$$d_{\text{Hell}}(\pi, \pi_m)^2 \leq K_3 \epsilon^2 + K_4 \mu \left( \Omega_{\perp}^{(m)}(\epsilon) \right) + \left( K_5 \epsilon + K_6 \mu \left( \Omega_{\perp}^{(m)}(\epsilon) \right) \right)^2.$$

As  $\epsilon > 0$  and  $\mu \left( \Omega_{\perp}^{(m)}(\epsilon) \right) \geq 0$ , the preceding inequality can be rearranged as

$$d_{\text{Hell}}(\pi, \pi_m)^2 \leq \left( K_1 \epsilon + K_2 \mu \left( \Omega_{\perp}^{(m)}(\epsilon) \right) \right)^2,$$

for some constants  $K_1 > 0$  and  $K_2 > 0$ . □

Under certain technical conditions [40, 41], the pointwise error of the reduced-order model decreases asymptotically as the dimensionality of the reduced basis increases. Thus, the  $\epsilon$ -feasible set asymptotically grows with the reduced basis enrichment, and hence,  $\mu \left( \Omega_{\perp}^{(m)}(\epsilon) \right)$  asymptotically decays. If the posterior distribution is sufficiently well sampled such that

$$\mu \left( \Omega_{\perp}^{(m)}(\epsilon) \right) \leq \epsilon, \tag{20}$$

then the Hellinger distance (16) is characterized entirely by  $\epsilon$ , as shown in Theorem 7. Thus, by adaptively updating the data-driven reduced-order model until condition (20) is satisfied, we can build an approximate posterior distribution whose error is proportional to the user-specified error threshold  $\epsilon$ .

In practice, it is not feasible to check condition (20) within MCMC sampling, so we use heuristics to motivate the finite adaptation criterion in Definition 1. Consider a stationary Markov chain that has invariant distribution  $\pi(\mathbf{x}|\mathbf{d}_{\text{obs}})$ . We assume that the probability of visiting  $\Omega_{\perp}^{(m)}(\epsilon)$  is proportional to its posterior measure. Suppose we have  $\mu \left( \Omega_{\perp}^{(m)}(\epsilon) \right) = \epsilon$ , and then the probability of the Markov chain visiting  $\Omega_{\perp}^{(m)}(\epsilon)$  is about  $c\epsilon$ , for some  $c > 0$ . In this case, the average number of MCMC steps needed for the next reduced basis refinement is about  $\frac{1}{c\epsilon}$ . As the posterior measure of  $\Omega_{\perp}^{(m)}(\epsilon)$  decays asymptotically with refinement of the reduced basis, the average number of steps needed for the basis refinement asymptotically increases. Thus, we can treat condition (20) as holding if the average number of steps used for reduced basis refinement exceeds  $N_{\text{max}} = \frac{1}{c\epsilon}$ , and then terminate the adaptation. We recommend to choose a small  $c$  value, for example  $c = 0.1$ , to delay the stopping time of the adaptation. In this way, the adaptive construction process can search the parameter space more thoroughly to increase the likelihood that the condition  $\mu \left( \Omega_{\perp}^{(m)}(\epsilon) \right) < \epsilon$  is satisfied.

## 5. EXTENSION TO APPROXIMATE BAYESIAN INFERENCE

The full target algorithm introduced in Section 3.2 has to evaluate the forward model after each subchain simulation to preserve ergodicity. The resulting Monte Carlo estimator (6) is unbiased. However, the ESS for a given budget of computing time is still characterized by the number of full model evaluations. To increase the ESS, we also consider an alternative approach that directly samples from the approximate posterior distribution when the reduced-order model has sufficient accuracy.

Suppose that the scaled error indicator  $\hat{\tau}(\mathbf{x})$  provides a reliable estimate of the scaled true error of the reduced-order model. Then, the reliability and the refinement of the reduced-order model can be controlled by the error indicator. During MCMC sampling, we only evaluate the full model to

correct a Metropolis acceptance and to update the reduced-order model when the error indicator exceeds the threshold  $\epsilon$ . When a reduced-order model evaluation has error indicator less than  $\epsilon$ , we treat the reduced-order model as a sufficient approximation to the full model. In this case, decisions in the MCMC algorithm are based on the approximate posterior distribution.

Compared to the full target algorithm that draws samples from the full posterior distribution, this approach only samples from an approximation that mimics the full posterior up to the error threshold  $\epsilon$ . We refer to the proposed approach as the ‘ $\epsilon$ -approximate algorithm’. Even though sampling the approximate posterior distribution will introduce bias to the Monte Carlo estimator (6), this bias may be acceptable if the resulting Monte Carlo estimator has a smaller MSE for a given amount of computational effort, compared with the standard Metropolis–Hasting algorithm. In the remainder of this section, we provide details on the algorithm and analyze the bias of the resulting Monte Carlo estimator.

### 5.1. $\epsilon$ -approximate algorithm

Algorithm 2 details the  $\epsilon$ -approximate algorithm. During the adaptation, for a proposed candidate  $\mathbf{x}'$ , we discard the reduced-order model evaluation if its error indicator exceeds the new upper threshold  $\epsilon_0$ . We set  $\epsilon_0 = 1$ , which means that Algorithm 2 does not use the information from the reduced-order model if its estimated error is greater than one standard deviation of the measurement noise. In this case (lines 3–6), we run the full model directly to evaluate the posterior distribution and the acceptance probability. If the error indicator of the reduced-order model is between the lower threshold  $\epsilon$  and the upper threshold  $\epsilon_0$  (lines 7–15), the reduced-order model is considered to be a reasonable approximation, and the delayed acceptance scheme is used to make the correction. If the error indicator is less than the lower threshold  $\epsilon$ , or if the adaptation is stopped (lines 17–19), the reduced-order model is considered to be a sufficiently accurate approximation to the full model and is used to accept/reject the proposal directly.

The adaptation criterion used in line 2 of Algorithm 2 has two conditions: the dimension of reduced basis should not exceed a specified threshold  $M$ , and the finite adaptation criterion (Definition 1) should not yet be triggered. The reduced basis is updated using the full model evaluations at proposals accepted by MCMC.

When the reduced basis dimension reaches  $M$  but the finite adaptation criterion is not satisfied, it is not appropriate to use the  $\epsilon$ -approximate algorithm for the prescribed error threshold. This is because the large reduced-order model error can potentially result in unbounded bias in the Monte Carlo estimator. In this situation, we should instead use the full target algorithm, for which convergence is guaranteed.

### 5.2. Monte Carlo error of the $\epsilon$ -approximate algorithm

To analyze the performance of the  $\epsilon$ -approximate algorithm, we compare the MSE of the resulting Monte Carlo estimator with that of a standard single-stage MCMC algorithm that samples the full posterior distribution.

We wish to compute the expectation of a function  $h(\mathbf{x})$  over the posterior distribution  $\pi(\mathbf{x}|\mathbf{d}_{\text{obs}})$ , that is,

$$I(h) = \int_{\mathbf{x}} h(\mathbf{x})\pi(\mathbf{x}|\mathbf{d}_{\text{obs}}) d\mathbf{x}, \quad (21)$$

where the first and second moments of  $h(\mathbf{x})$  are assumed to be finite. Suppose a single-stage MCMC algorithm can sample the full posterior distribution for  $N_1$  steps in a fixed amount of CPU time, and  $\text{ESS}(h)$  effective samples are produced. The resulting Monte Carlo estimator

$$\widehat{I}(h) = \frac{1}{N_1} \sum_{i=1}^{N_1} h(\mathbf{x}^{(i)}), \quad \mathbf{x}^{(i)} \sim \pi(\cdot|\mathbf{d}_{\text{obs}}), \quad (22)$$

**Algorithm 2**  $\epsilon$ -approximate algorithm

**Require:** Given the subchain length  $L$ , the maximum allowable reduced basis dimension  $M$ , the upper threshold  $\epsilon_0$ , and the error threshold  $\epsilon$ . At step  $n$ , given state  $X_n = \mathbf{x}$ , a proposal  $q(\mathbf{x}, \cdot)$ , and a reduced-order model  $F_m(\cdot)$  defined by the reduced basis  $\mathbf{V}_m$ , one step of the algorithm is:

- 1: Propose  $\mathbf{x}' \sim q(\mathbf{x}, \cdot)$ , then evaluate the reduced-order model  $F_m(\mathbf{x}')$  and  $\hat{\mathbf{t}}_m(\mathbf{x}')$
- 2: **if finite adaptation criterion** is not satisfied **and**  $m < M$  **and**  $\|\hat{\mathbf{t}}_m(\mathbf{x}')\|_\infty \geq \epsilon$  **then**
- 3: **if**  $\|\hat{\mathbf{t}}_m(\mathbf{x}')\|_\infty \geq \epsilon_0$  **then**
- 4: Discard the reduced-order model evaluation, and compute the acceptance probability using the full posterior distribution

$$\alpha(\mathbf{x}, \mathbf{x}') = 1 \wedge \frac{\pi(\mathbf{x}'|\mathbf{d}_{\text{obs}}) q(\mathbf{x}', \mathbf{x})}{\pi(\mathbf{x}|\mathbf{d}_{\text{obs}}) q(\mathbf{x}, \mathbf{x}' )}$$

- 5: Accept/reject  $\mathbf{x}'$  according to  $\text{Uniform}(0, 1] < \alpha(\mathbf{x}, \mathbf{x}')$
- 6: Update the reduced basis  $\mathbf{V}_m$  to  $\mathbf{V}_{m+1}$  using the new full model evaluation at accepted  $\mathbf{x}'$  by a Gram-Schmidt process
- 7: **else**  $\{\epsilon \leq \|\hat{\mathbf{t}}_m(\mathbf{x}')\|_\infty < \epsilon_0\}$
- 8: Run the delayed acceptance for 1 step, evaluate the acceptance probability

$$\beta_1(\mathbf{x}, \mathbf{x}') = 1 \wedge \frac{\pi_m(\mathbf{x}'|\mathbf{d}_{\text{obs}}) q(\mathbf{x}', \mathbf{x})}{\pi_m(\mathbf{x}|\mathbf{d}_{\text{obs}}) q(\mathbf{x}, \mathbf{x}' )}$$

- 9: **if**  $\text{Uniform}(0, 1] < \beta_1(\mathbf{x}, \mathbf{x}')$  **then**
- 10: Run the full model at  $\mathbf{x}'$  to evaluate the full posterior and the acceptance probability

$$\beta_2(\mathbf{x}, \mathbf{x}') = 1 \wedge \frac{\pi(\mathbf{x}'|\mathbf{d}_{\text{obs}}) \pi_m(\mathbf{x}|\mathbf{d}_{\text{obs}})}{\pi(\mathbf{x}|\mathbf{d}_{\text{obs}}) \pi_m(\mathbf{x}'|\mathbf{d}_{\text{obs}})}$$

- 11: Accept/reject  $\mathbf{x}'$  according to  $\text{Uniform}(0, 1] < \beta_2(\mathbf{x}, \mathbf{x}')$
- 12: **else**
- 13: Reject  $\mathbf{x}'$  by setting  $X_{n+1} = \mathbf{x}$
- 14: **end if**
- 15: Update the reduced basis  $\mathbf{V}_m$  to  $\mathbf{V}_{m+1}$  as in Line 6
- 16: **end if**
- 17: **else**  $\{\|\hat{\mathbf{t}}_m(\mathbf{x}')\|_\infty < \epsilon$  **or** Adaptation is stopped}
- 18: The reduced-order model is used directly to evaluate the acceptance probability

$$\theta(\mathbf{x}, \mathbf{x}') = 1 \wedge \frac{\pi_m(\mathbf{x}'|\mathbf{d}_{\text{obs}}) q(\mathbf{x}', \mathbf{x})}{\pi_m(\mathbf{x}|\mathbf{d}_{\text{obs}}) q(\mathbf{x}, \mathbf{x}' )}$$

- 19: Accept/reject  $\mathbf{x}'$  according to  $\text{Uniform}(0, 1] < \theta(\mathbf{x}, \mathbf{x}')$
- 20: **end if**

has MSE

$$\text{MSE}(\widehat{I(h)}) = \frac{\text{Var}(h)}{\text{ESS}(h)}, \quad (23)$$

which is characterized by the ESS and the variance of  $h$  over  $\pi(\mathbf{x}|\mathbf{d}_{\text{obs}})$ .

By sampling the approximate posterior distribution, the expectation  $I(h)$  can be approximated by

$$I_m(h) = \int_{\mathbb{X}} h(\mathbf{x}) \pi_m(\mathbf{x}|\mathbf{d}_{\text{obs}}) d\mathbf{x}. \quad (24)$$

Suppose we can sample the approximate posterior for  $N_2$  steps in the same amount of CPU time as sampling the full posterior, and that these  $N_2$  samples have ESS

$$\text{ESS}_m(h) = S(m) \times \text{ESS}(h), \quad (25)$$

where  $S(m) > 1$  is the speedup factor that depends on the computational expense of the reduced-order model  $F_m(\mathbf{x})$ . The Monte Carlo estimator

$$\widehat{I}_m(h) = \frac{1}{N_2} \sum_{i=1}^{N_2} h(\mathbf{x}^{(i)}), \quad \mathbf{x}^{(i)} \sim \pi_m(\cdot|\mathbf{d}_{\text{obs}}), \quad (26)$$

has the MSE

$$\text{MSE}(\widehat{I}_m(h)) = \frac{\text{Var}_m(h)}{\text{ESS}_m(h)} + \text{Bias}(\widehat{I}_m(h))^2, \quad (27)$$

where the bias is defined by

$$\text{Bias}(\widehat{I}_m(h)) = I_m(h) - I(h). \quad (28)$$

We are interested in the situation for which sampling the approximation leads to a smaller MSE than sampling the full posterior distribution, that is,

$$\frac{\text{Var}_m(h)}{\text{ESS}_m(h)} + \text{Bias}(\widehat{I}_m(h))^2 \leq \frac{\text{Var}(h)}{\text{ESS}(h)}. \quad (29)$$

Rearranging the preceding inequality gives

$$\text{ESS}(h) \leq \tau := \frac{\text{Var}(h)}{\text{Bias}(\widehat{I}_m(h))^2} \left( 1 - \frac{\text{Var}_m(h)}{\text{Var}(h)} \frac{1}{S(m)} \right). \quad (30)$$

Equation (30) reveals that when our target ESS for drawing from the full posterior distribution does not exceed the bound  $\tau$ , sampling the approximate posterior will produce a smaller MSE. This suggests that the  $\epsilon$ -approximate algorithm will be more accurate for a fixed computational cost than the single-stage MCMC when the target ESS of the single-stage MCMC satisfies (30). In such cases, the MSE of the Monte Carlo estimator is dominated by the variance rather than the bias.

The bias is characterized by the Hellinger distance between the full posterior distribution and its approximation (Lemma 6.37 of [22]), that is,

$$\text{Bias}(\widehat{I}_m(h))^2 \leq 4 \left( \int h(\mathbf{x})^2 \pi(\mathbf{x}|\mathbf{d}_{\text{obs}}) d\mathbf{x} + \int h(\mathbf{x})^2 \pi_m(\mathbf{x}|\mathbf{d}_{\text{obs}}) d\mathbf{x} \right) d_{\text{Hell}}(\pi, \pi_m)^2.$$

We assume that there exists an  $m^* = m(\epsilon)$  and a set of samples  $\{\mathbf{x}^{(0)}, \dots, \mathbf{x}^{(m^*)}\}$  such that the resulting reduced-order model  $F_{m^*}(\mathbf{x})$  satisfies the condition (20), that is,  $\mu(\Omega_{\perp}^{(m^*)}(\epsilon)) \leq \epsilon$ . Applying Theorem 7, the ratio of variance to squared bias can be simplified to

$$\frac{\text{Var}(h)}{\text{Bias}(\widehat{I}_{m(\epsilon)}(h))^2} \geq \frac{K}{\epsilon^2},$$

for some constant  $K > 0$ , and hence, we have

$$\tau \geq \frac{K}{\epsilon^2} \left( 1 - \frac{\text{Var}_{m(\epsilon)}(h)}{\text{Var}(h)S(m(\epsilon))} \right). \quad (31)$$

For problems that have reliable error indicators or estimators, the  $\epsilon$ -approximate algorithm provides a viable way to select the set of  $m^* = m(\epsilon)$  samples for computing the snapshots. However, it is computationally infeasible to verify that condition (20) holds in practice. We thus employ the finite adaptation criterion (Definition 1) to perform a heuristic check on the condition (20), as discussed in Section 4.

The bound  $\tau$  is characterized by the user-given error threshold  $\epsilon$  and the speedup factor  $S(m(\epsilon))$ , where  $S(m(\epsilon))$  is a problem-specific factor that is governed by the convergence rate and computational complexity of the reduced-order model. For a reduced-order model such that  $S(m(\epsilon)) > \text{Var}_{m(\epsilon)}(h)/\text{Var}(h)$ , there exists a  $\tau$  so that the MSE of sampling the approximate posterior for  $\tau \times S(m(\epsilon))$  steps will be less than the MSE of sampling the full posterior for the same amount of CPU time. In the regime where the reduced-order models have sufficiently large speedup factors, the bound  $\tau$  is dominated by  $\epsilon^2$ , and hence, decreasing  $\epsilon$  results in a higher bound  $\tau$ . However, there is a trade-off between the numerical accuracy and speedup factors. We should avoid choosing a very small  $\epsilon$  value, because this can potentially lead to a high-dimensional reduced basis and a correspondingly expensive reduced-order model such that  $S(m(\epsilon)) < \text{Var}_{m(\epsilon)}(h)/\text{Var}(h)$ , where the ratio  $\text{Var}_{m(\epsilon)}(h)/\text{Var}(h)$  should be close to one for such an accurate reduced-order model. In this case, sampling the approximate posterior can be less efficient than sampling the full posterior.

## 6. NUMERICAL RESULTS AND DISCUSSION

To benchmark the proposed algorithms, we use a model of isothermal steady flow in porous media, which is a classical test case for inverse problems.

### 6.1. Problem setup

Let  $D = [0, 1]^2$  be the problem domain,  $\partial D$  the boundary of the domain, and  $r \in D$  the spatial coordinate. Let  $k(r)$  be the unknown permeability field,  $u(r)$  the pressure head, and  $q(r)$  the source/sink. The pressure head for a given realization of the permeability field is governed by

$$\nabla \cdot (k(r)\nabla u(r)) + q(r) = 0, \quad r \in D, \quad (32)$$

where the source/sink term  $q(r)$  is defined by the superposition of four weighted Gaussian plumes with standard deviation 0.05, centered at  $r = [0.3, 0.3]$ ,  $[0.7, 0.3]$ ,  $[0.7, 0.7]$ ,  $[0.3, 0.7]$ , and with weights  $\{2, -3, -2, 3\}$ . A zero-flux Neumann boundary condition

$$k(r)\nabla u(r) \cdot \vec{n}(r) = 0, \quad r \in \partial D, \quad (33)$$

is prescribed, where  $\vec{n}(r)$  is the outward normal vector on the boundary. To make the forward problem well posed, we impose the extra boundary condition

$$\int_{\partial D} u(r) dl(r) = 0. \quad (34)$$

Equation (32) with boundary conditions (33) and (34) is solved by the finite element method with  $120 \times 120$  linear elements. This leads to the system of Equation (9).

In Section 6.2, we use a nine-dimensional example to carry out numerical experiments to benchmark various aspects of our algorithms. In this example, the spatially distributed permeability field is projected onto a set of radial basis functions, and hence, inference is carried out on the weights associated with each of the radial basis functions. In Section 6.3, we apply our algorithms to a higher dimensional problem, where the parameters are defined on the computational grid and endowed with a Gaussian process prior. Both examples use fixed Gaussian proposal distributions, where the covariances of the proposals are estimated from a short run of the  $\epsilon$ -approximate algorithm. In Section 6.4, we offer additional remarks on the performance of the  $\epsilon$ -approximate algorithm.

6.2. The nine-dimensional inverse problem

The permeability field is defined by  $N_p = 9$  radial basis functions:

$$k(r) = \sum_{i=1}^{N_p} b(r; r_i) x_i, \quad b(r; r_i) = \exp \left[ -0.5 \left( \frac{\|r - r_i\|}{0.15} \right)^2 \right], \quad (35)$$

where  $r_1, \dots, r_9$  are the centers of the radial basis functions. These radial basis functions are shown in Figure 2. The prior distributions on each of the weights  $x_i, i = 1, \dots, 9$  are independent and log-normal, and hence, we have

$$\pi_0(\mathbf{x}) \propto \prod_{i=1}^{N_p} \exp \left( -\frac{\log(x_i)^2}{2\sigma_0^2} \right), \quad (36)$$

where  $\sigma_0 = 2$  and  $N_p = 9$ . The true permeability field used to generate the test data, and the corresponding pressure head are shown in Figure 3. The measurement sensors are evenly distributed over  $D$  with grid spacing 0.1, and the signal-to-noise ratio of the observed data is 50.

Numerical experiments for various choices of  $\epsilon$  are carried out to test the computational efficiency of both algorithms and the dimensionality of reduced basis in the reduced-order model. For  $\epsilon = \{10^{-1}, 10^{-2}, 10^{-3}\}$ , we run the full target algorithm for  $10^4$  iterations, with subchain length  $L = 50$ . To make a fair comparison in terms of the number of posterior evaluations, we run the  $\epsilon$ -approximate algorithm for  $5 \times 10^5$  iterations, also with  $\epsilon = \{10^{-1}, 10^{-2}, 10^{-3}\}$ . For both algorithms, the reduced-order model construction process is started at the beginning of the MCMC simulation. We set  $c = 10^{-1}$  in the finite adaptation criterion 1. As a reference, we run a single-stage MCMC algorithm

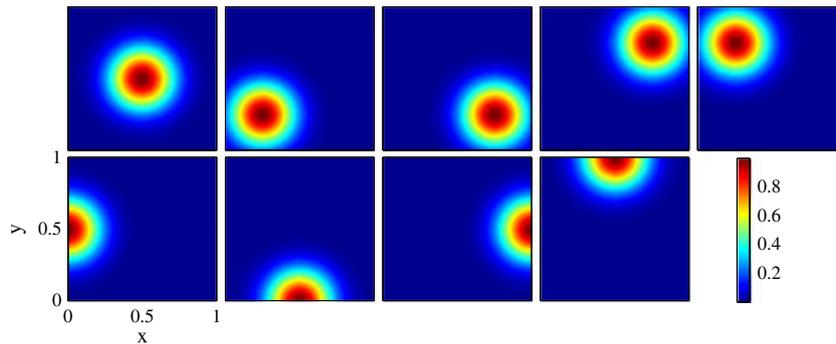


Figure 2. Radial basis functions used to define the permeability field in the nine-dimensional example.

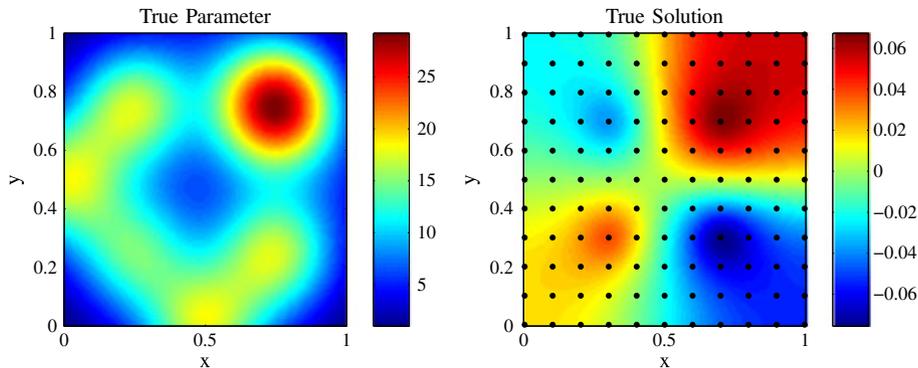


Figure 3. Setup of the test case for the nine-dimensional example. Left: the true permeability used for generating the synthetic data sets. Right: the model outputs of the true permeability. The black dots indicate the measurement sensors.

for  $5 \times 10^5$  iterations using the same proposal distribution. The reference algorithm only uses the full posterior distribution. The first 2000 samples of the simulations generated by the full target algorithm are discarded as burn-in samples. Similarly, the first  $10^5$  samples (to match the number of reduced-order model evaluations in the full target algorithm) are discarded for the simulations using the  $\epsilon$ -approximate algorithm and the reference algorithm.

In the remainder of this section, we provide various benchmarks of our data-driven model reduction approach and the two algorithms. These include the following:

1. A comparison of the full target algorithm and the  $\epsilon$ -approximate algorithm with the reference algorithm.
2. A comparison of the data-driven reduced-order model with the reduced-order model built with respect to the prior distribution.
3. A demonstration of the impact of observed data on our data-driven reduced-order model.

*6.2.1. Computational efficiency.* Table I summarizes the number of full model evaluations, the dimensionality of reduced basis, the CPU time, the ESS, and the speedup factor, comparing the full target algorithm and the  $\epsilon$ -approximate algorithm with the reference algorithm. For the reduced-order models generated by the adaptive construction process, we provide estimates of the posterior measure of the complement of the  $\epsilon$ -feasible set,  $\mu(\Omega_{\perp}^{(m)}(\epsilon))$ . We also provide a summary of the average second-stage acceptance probability,  $\beta$ , for the full target algorithm.

For the full target algorithm, the average second-stage acceptance probabilities for all three  $\epsilon$  values are greater than 0.96 in this test case. This shows that the reduced-order models produced by all three  $\epsilon$  values are reasonably accurate compared with the full model, and hence, simulating the approximate posterior distribution in the first stage usually yields the same Metropolis acceptance decision as simulating the full posterior distribution. As we enhance the accuracy of the reduced-order model by decreasing the value of  $\epsilon$ , the dimensionality of the resulting reduced basis increases, and thus, the reduced-order model takes longer to evaluate. Because the full target algorithm evaluates the full model for every 50 reduced-order model evaluations, its computational cost is dominated by the number of full model evaluations. Thus, the speedup factors for all three choices of  $\epsilon$  are similar (approximately 40). As all three reduced-order models are reasonably accurate here, the efficiency gain of using a small  $\epsilon$  value is not significant. In this situation, one could consider simulating the subchain in the first stage for more iterations (by increasing the subchain length  $L$ ) when the value of  $\epsilon$  is small.

The  $\epsilon$ -approximate algorithm produces speedup factors that are 4.7 to 7.4 times higher than the speedup factor of the full target algorithm in this test case. A larger  $\epsilon$  value produces a larger speedup factor, because the dimension of the associated reduced basis is smaller.

Table I. Comparison of the computational efficiency of the full target algorithm with  $\epsilon = \{10^{-1}, 10^{-2}, 10^{-3}\}$  and the  $\epsilon$ -approximate algorithm with  $\epsilon = \{10^{-1}, 10^{-2}, 10^{-3}\}$  with the reference algorithm.

	Reference	Full target			$\epsilon$ -approximate		
Error threshold $\epsilon$	—	$10^{-1}$	$10^{-2}$	$10^{-3}$	$10^{-1}$	$10^{-2}$	$10^{-3}$
Average $\beta$	—	0.97	0.98	0.98	—	—	—
Full model evaluations	$5 \times 10^5$	$10^4$	$10^4$	$10^4$	13	33	57
Reduced basis vectors	—	14	33	57	13	33	57
CPU time (s)	34,470	754	772	814	115	138	187
ESS	4709	4122	4157	4471	4672	4688	4834
ESS/CPU time	$1.4 \times 10^{-1}$	5.5	5.4	5.5	40.6	33.9	25.9
Speedup factor	1	40	39	40	297	248	189
$\mu(\Omega_{\perp}^{(m)}(\epsilon))$	—	$0.1 \times 10^{-4}$	$0.7 \times 10^{-4}$	0	$1.3 \times 10^{-4}$	$0.8 \times 10^{-4}$	0

The second step acceptance probability  $\beta$  is defined in Algorithm 1. The posterior measure of the complement of the  $\epsilon$ -feasible set,  $\mu(\Omega_{\perp}^{(m)}(\epsilon))$ , is given in Definition 4.

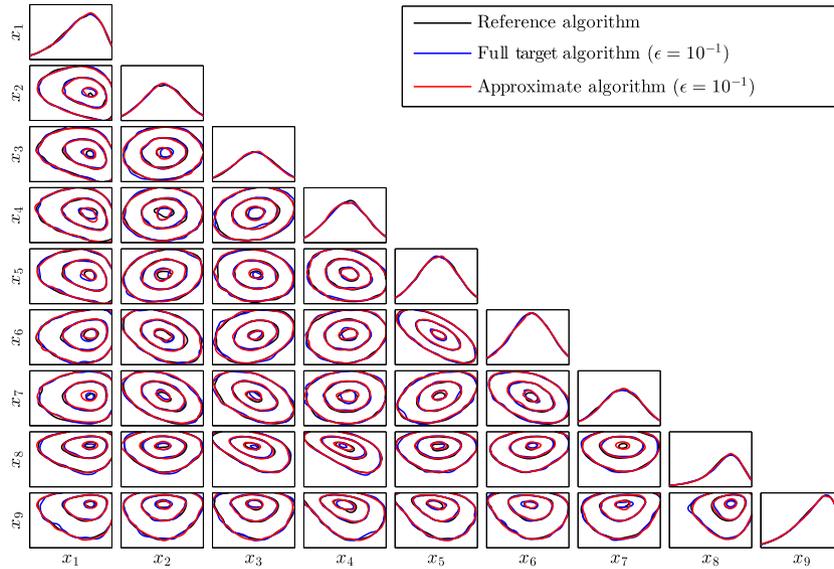


Figure 4. The marginal distribution of each component of the parameter  $\mathbf{x}$ , and the contours of the marginal distribution of each pair of components. Black line: the reference algorithm. Blue line: the full target algorithm with  $\epsilon = 10^{-1}$ . Red line: the  $\epsilon$ -approximate algorithm with  $\epsilon = 10^{-1}$ .

To assess the sampling accuracy of the  $\epsilon$ -approximate algorithm, Figure 4 provides a visual inspection of the marginal distributions of each component of the parameter  $\mathbf{x}$ , and the contours of the marginal distributions of each pair of components. The black lines represent the results generated by the reference algorithm, the blue lines represent results of the full target algorithm with  $\epsilon = 10^{-1}$ , and red lines represent results of the  $\epsilon$ -approximate algorithm with  $\epsilon = 10^{-1}$ . The results from the more accurate simulations that use smaller  $\epsilon$  values are not shown, as they are visually close to the case  $\epsilon = 10^{-1}$ . The plots in Figure 4 suggest that all the algorithms generate similar marginal distributions in this test case. We note that both the reference algorithm and the full target algorithm sample from the full posterior distribution, and thus, the small differences in the contours produced by various algorithms are likely caused by a Monte Carlo error.

An alternative way to assess the computational efficiency and sampling accuracy of the  $\epsilon$ -approximate algorithm is to compare the number of effective samples generated by the  $\epsilon$ -approximate algorithm and the reference algorithm for a fixed amount of CPU time. As shown in Table I, the  $\epsilon$ -approximate algorithm with  $\epsilon = 10^{-1}$  generates 4672 effective samples in 115.3 s; the reference algorithm can only generate about 16 effective samples in the same amount of CPU time. In the situation where the desired number of effective samples is at least an order of magnitude larger than the speedup factor, using the  $\epsilon$ -approximate algorithm is clearly advantageous to using the reference algorithm.

For both the full target algorithm and the  $\epsilon$ -approximate algorithm, and for each choice of  $\epsilon$ , we use  $2 \times 10^6$  samples generated by the reference algorithm to compute the Monte Carlo estimator of the posterior measure of the complement of the  $\epsilon$ -feasible set for the final reduced-order model produced by the adaptive construction process. As shown in Table I, for all the reduced-order models, we have estimated  $\mu \left( \Omega_{\perp}^{(m)}(\epsilon) \right) < \epsilon$ . This suggests that the Hellinger distances between the full posterior distribution and its approximation can be characterized by the  $\epsilon$  values in all three cases, and thus, the finite adaptation criterion (Definition 1) with  $c = 10^{-1}$  provides a useful indicator for terminating adaptation.

For  $\epsilon = 10^{-1}$ , we note that the dimensions of the reduced bases produced by the full target algorithm and the  $\epsilon$ -approximate algorithm are different. This is because the snapshots are evaluated at selected samples that are randomly drawn from the posterior. The spread of the sample set slightly affects the accuracy of the reduced-order model. Nonetheless, both reduced-order models achieve

the desirable level of accuracy because the estimated posterior measures  $\mu\left(\Omega_{\perp}^{(m)}(\epsilon)\right)$  are less than  $\epsilon = 10^{-1}$  in this case.

Numerical experiments with  $c = 10^{-2}$  and  $c = 10^{-3}$  in the finite adaptation criterion (Definition 1) are also conducted. For both algorithms, choosing these smaller  $c$  values leads only to one or two additional basis vectors being added in all the test cases, compared with the case  $c = 10^{-1}$ . The resulting marginal distributions generated by using  $c = 10^{-2}$  and  $c = 10^{-3}$  are similar to the case  $c = 10^{-1}$ . For brevity, the sampling results for these experiments are not reported. We consistently observe that the number of MCMC steps between adjacent basis enrichments increases as the adaptive construction progresses in these experiments. This is expected because the posterior measure  $\mu\left(\Omega_{\perp}^{(m)}(\epsilon)\right)$  asymptotically decreases with reduced basis enrichment. In this situation, choosing a smaller  $c$  value leads only to minor increases in both of the numerical accuracy and the computational cost of the reduced-order model. Thus, the sampling accuracy and the overall computational load of both sampling algorithms are not sensitive to the smaller  $c$  values in this case.

**6.2.2. Comparison with a reduced-order model built from the prior.** Now, we compare the accuracy of the data-driven reduced-order model built with  $\epsilon = 10^{-3}$  to that of a reduced-order model constructed with respect to the prior distribution (36). To construct the reduced-order model with respect to the prior, we use proper orthogonal decomposition (POD). To compute the snapshots,  $10^4$  random prior samples are drawn. The POD eigenspectrum is shown in the left plot of Figure 5. The eigendecomposition is truncated when it captures all but  $10^{-8}$  energy (relative 2-norm of the POD eigenvalues), leading to 110 reduced basis vectors being retained in the POD basis.

By using posterior samples generated from a separate reference algorithm, we compute the expectation of the  $L_{\infty}$  norm of the scaled true error (15) over the full posterior distribution. The  $L_{\infty}$  norm of the scaled true error gives the worst-case sensor error; its expectation over the posterior quantifies the average numerical accuracy of the resulting reduced-order model. The right plot of Figure 5 shows this expectation with respect to the dimension of the reduced basis. For this test case, the data-driven reduced-order model undergoes a significant accuracy improvement once it includes at least 10 reduced basis vectors. The figure shows that the data-driven reduced-order model has a better convergence rate compared with the reduced-order model built from the prior.

**6.2.3. The influence of posterior concentration.** The amount of information carried in the data affects the dimension of the data-driven reduced-order model and hence has an impact on its computational efficiency. By adjusting the signal-to-noise ratio in the observed data, we examine the influence of the posterior concentration on the dimension of the reduced basis. We gradually increase the signal-to-noise ratio from 10 to 100 and record the number of reduced basis vectors in the

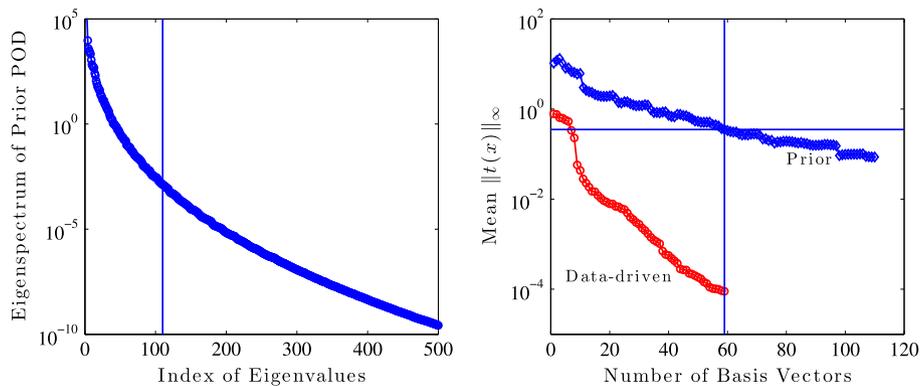


Figure 5. Left: The eigenspectrum of the POD basis computed from the prior distribution. The vertical line indicates the index for truncating the eigendecomposition. Right: Comparison of the numerical accuracy of the data-driven reduced-order model ( $\epsilon = 10^{-3}$ ) with the reduced-order model built with respect to the prior distribution (36). The expectation of the  $L_{\infty}$  norm of the scaled true errors over the full posterior distribution is used as a benchmark.

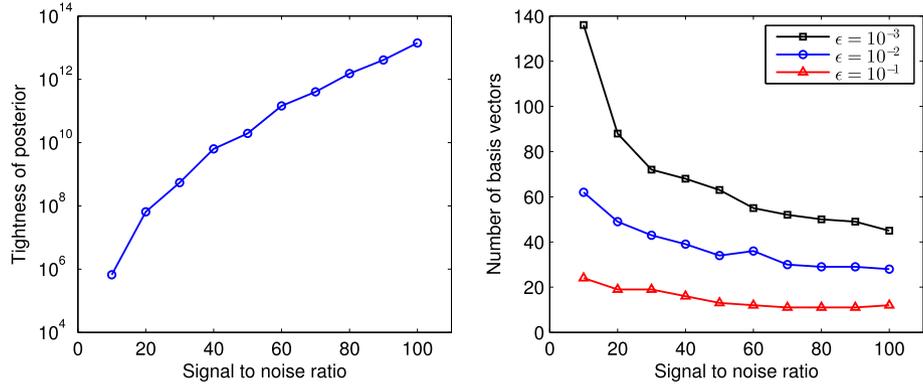


Figure 6. Left: the posterior concentration (37) versus the signal-to-noise ratio of the data. Right: the number of reduced basis vectors versus the signal-to-noise ratio.

reduced-order models. To quantify the posterior concentration, we use the ‘tightness’ of the posterior distribution defined by

$$\prod_{i=1}^{N_p} \frac{\sigma_0(x_i)}{\sigma(x_i)}, \quad (37)$$

where  $\sigma(x_i)$  is the standard deviation of the posterior marginal of  $x_i$ , and  $\sigma_0(x_i)$  is the standard deviation of the corresponding prior marginal. In Figure 6, we observe that the dimension of the reduced basis decreases as the signal-to-noise ratio increases. For this test problem, the larger amount of information in the data results in a lower dimensional reduced basis because our approach exploits the increasing concentration of the posterior.

### 6.3. The high dimensional inverse problem

In the high-dimensional example, a log-normal distribution is employed to model the permeabilities as a random field. Let  $r_i, i = 1, \dots, N_g$  denote the coordinates of the  $N_g$  grid points. Let  $k(r_i) = \exp(x(r_i))$  be the permeability field defined at each grid point, and then the latent field  $\mathbf{x} = [x(r_1), \dots, x(r_{N_g})]^T$  follows a Gaussian process prior

$$\pi_0(\mathbf{x}) \propto \exp\left(-\frac{1}{2}\mathbf{x}^T \Sigma \mathbf{x}\right), \quad \Sigma_{ij} = \exp\left(-\frac{|r_i - r_j|^2}{2s^2}\right), \quad (38)$$

where  $s = 0.25$  is used to provide sufficient spatial variability. After applying the eigendecomposition of the prior covariance, the parameters are defined on 43 eigenvectors that preserve 99.99% energy of the prior distribution. To avoid an inverse crime, we use a ‘true’ permeability field that is not directly drawn from the prior distribution. Figure 7 shows the true permeability field and the simulated pressure head. The setup of the measurement sensors is the same as the nine-dimensional example in Section 6.2.

Using the same setting as the nine-dimensional case, we simulate the full target algorithm with  $\epsilon = 10^{-1}$  for  $10^4$  iterations, with subchain length  $L = 50$ . For these full target MCMC simulations, the first 2000 samples are discarded as burn-in samples. We simulate the  $\epsilon$ -approximate algorithm with  $\epsilon = \{10^{-1}, 10^{-2}, 10^{-3}\}$ , for  $5 \times 10^5$  iterations. The single-stage MCMC method is simulated for  $5 \times 10^5$  iterations as the reference. For all the  $\epsilon$ -approximate MCMC simulations and the reference MCMC simulation, the first  $10^5$  samples are discarded as burn-in samples.

Table II summarizes the number of full model evaluations, the number of reduced basis vectors, the CPU time, ESS, and speedup factor. The speedup factor of the full target algorithm is about 67. In comparison, the speedup factors of the  $\epsilon$ -approximate algorithm range from 61 to 249. The speedup factor increases as the error threshold  $\epsilon$  increases. Figure 8 shows the mean and standard

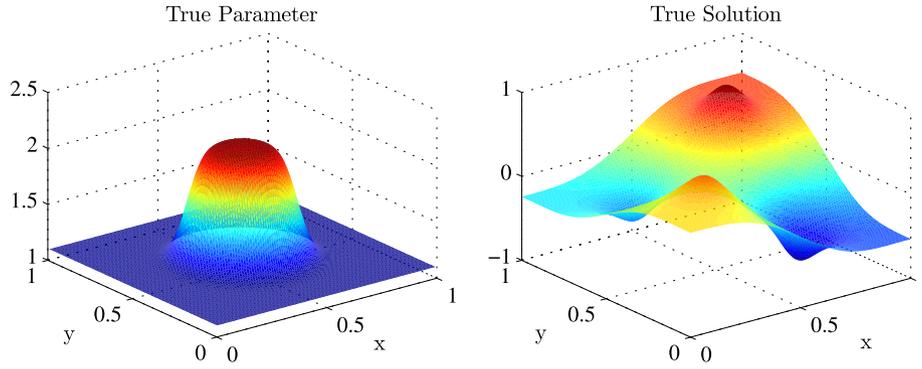


Figure 7. Setup of the test case for the high-dimensional example. Left: the true permeability used for generating the synthetic data sets. Right: the model outputs of the true permeability.

Table II. Comparison of the computational efficiency of the full target algorithm, the  $\epsilon$ -approximate algorithm, and the reference algorithm for the high-dimensional problem.

	Reference	Full target		$\epsilon$ -approximate	
Error threshold $\epsilon$	—	$10^{-1}$	$10^{-1}$	$10^{-2}$	$10^{-3}$
Full model evaluations	$5 \times 10^5$	$10^4$	62	129	209
Reduced basis vectors	—	64	62	129	209
CPU time (s)	75,300	1011	302	660	1226
ESS	2472	2221	2468	2410	2445
ESS/CPU time	$3.3 \times 10^{-2}$	2.2	8.2	3.7	2.0
Speedup factor	1	67	249	111	61
$\mu(\Omega_{\perp}^{(m)}(\epsilon))$	—	0	0	$1.0 \times 10^{-3}$	$1.6 \times 10^{-4}$

$\epsilon = 10^{-1}$  is used in the full target algorithm. For the  $\epsilon$ -approximate algorithm, three examples with  $\epsilon = \{10^{-1}, 10^{-2}, 10^{-3}\}$  are given. The posterior measure of the complement of the  $\epsilon$ -feasible set,  $\mu(\Omega_{\perp}^{(m)}(\epsilon))$ , is given in Definition 4.

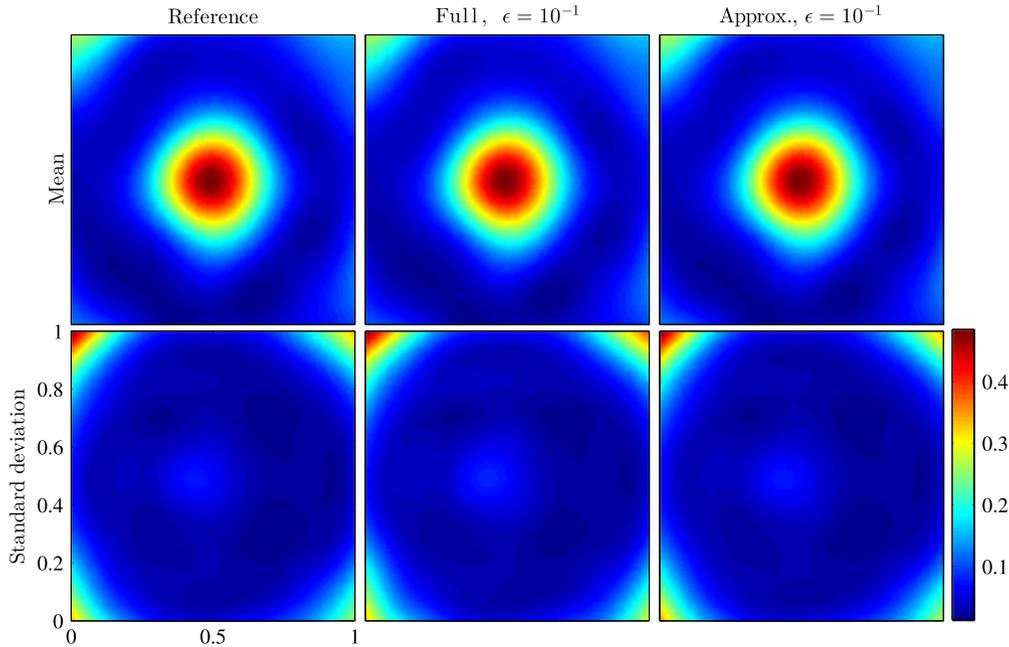


Figure 8. Mean (top row) and standard deviation (bottom row) at each spatial location of the permeability field. From left to right: the reference algorithm, the full target algorithm with  $\epsilon = 10^{-1}$ , and the  $\epsilon$ -approximate algorithm with  $\epsilon = 10^{-1}$ .

deviation at each spatial location of the permeability field, generated from the reference algorithm, the full target algorithm, and the least accurate setting ( $\epsilon = 10^{-1}$ ) of the  $\epsilon$ -approximate algorithm. We observe that all algorithms produce similar estimates of mean and standard deviation in this test case.

The  $\mu\left(\Omega_{\perp}^{(m)}(\epsilon)\right)$  values estimated from samples generated by the reference algorithm for all three  $\epsilon$  values are also recorded in Table II. In this test example, we have  $\mu\left(\Omega_{\perp}^{(m)}(\epsilon)\right) < \epsilon$  for all three  $\epsilon$  values, and thus, the Monte Carlo estimator provided by the  $\epsilon$ -approximate algorithm can be characterized by the  $\epsilon$  values. We note that some of the estimated posterior measures  $\mu\left(\Omega_{\perp}^{(m)}(\epsilon)\right)$  have zero values in Table II, but these values do not necessarily mean that the posterior measures  $\mu\left(\Omega_{\perp}^{(m)}(\epsilon)\right)$  are exactly zero, because these are Monte Carlo estimates.

#### 6.4. Remarks on the $\epsilon$ -approximate algorithm

In the first case study, the  $\epsilon$ -approximate algorithm offers some speedup compared with the full target algorithm (range from 4.7 to 7.4). In the second case study, the speedup factor of the  $\epsilon$ -approximate algorithm compared with the full target algorithm drops to at most 3.7 (with  $\epsilon = 10^{-1}$ ), and it performs slightly worse than the full target algorithm for  $\epsilon = 10^{-3}$ . The speedup factor of the  $\epsilon$ -approximate algorithm decreases with  $\epsilon$  in both cases. This result is to be expected, as the computational cost of the reduced-order model depends on the dimensionality of the reduced basis, which grows as  $\epsilon$  decreases. For  $\epsilon = 10^{-3}$  in the second test case, the reduced-order model becomes computationally too expensive relative to the full model, and thus, we lose the efficiency gain of the  $\epsilon$ -approximate algorithm over the full target algorithm.

For problems that require only a limited number of effective samples, using the  $\epsilon$ -approximate algorithm can be more advantageous. This is because we can use a relatively large  $\epsilon$  value to keep the computational cost of the reduced-order model low, while the resulting MSE is still dominated by the variance of the estimator rather than the bias. If the goal is to obtain an accurate Monte Carlo estimator, in which the variance of the estimator is small compared with the bias resulting from sampling the approximate posterior distribution, we should use the full target algorithm. We also note that the accuracy and efficiency of the  $\epsilon$ -approximate algorithm depend on reliable error indicators or error estimators, while the full target algorithm always samples the full posterior distribution, regardless of the error indicator.

## 7. CONCLUSION

We have introduced a new data-driven model reduction approach for solving statistical inverse problems. Our approach constructs the reduced-order model using adaptively-selected posterior samples to compute the snapshots. The reduced-order model construction process is integrated into the posterior sampling, to achieve simultaneous posterior exploration and model reduction.

Based on the data-driven reduced-order model, we have also developed two MCMC algorithms to sample the posterior distribution more efficiently than standard full-model MCMC algorithms. The full target algorithm aims to accelerate sampling of the full posterior distribution by coupling the full and approximate posterior distributions together. The  $\epsilon$ -approximate algorithm samples an approximate posterior distribution and attempts to reduce the MSE of the resulting Monte Carlo estimator, compared with a standard MCMC algorithm. Both algorithms adaptively construct the reduced-order model online through the MCMC sampling. The full target algorithm preserves ergodicity with respect to the true posterior. The  $\epsilon$ -approximate algorithm does not sample the full posterior but can provide further speedups for some problems.

In the case studies, we have demonstrated that both algorithms are able to accelerate MCMC sampling of computationally expensive posterior distributions by up to two orders of magnitude and that the sampling accuracy of the  $\epsilon$ -approximate algorithm is comparable with that of a reference full-model MCMC. We have also used the first case study to show the numerical accuracy of the data-driven reduced-order model, compared with a reduced-order model that is built offline

with respect to the prior distribution. In this example, for the same number of reduced basis vectors, the posterior-averaged output error of the data-driven reduced-order model is several orders of magnitude smaller than that of the reduced-order model built with respect to the prior. Furthermore, we have demonstrated the impact of the amount of information carried in the observed data on the dimensionality of the reduced basis.

For solving statistical inverse problems, these results suggest that a data-driven reduced-order model is preferable to a reduced-order model built with respect to the prior, especially when the data are informative. Even though our approach is designed for constructing projection-based reduced-order models, the concept of building posterior-oriented surrogate models can be generalized to other approximation approaches such as Gaussian process regression and generalized polynomial chaos.

#### ACKNOWLEDGEMENTS

The authors thank Florian Augustin, Tan Bui-Thanh, Omar Ghattas, and Jinglai Li for many helpful comments and discussions. This work was supported by the United States Department of Energy, Office of Advanced Scientific Computing Research (ASCR), Applied Mathematics Program, awards DE-FG02-08ER2585 and DE-SC0009297, as part of the DiaMonD Multifaceted Mathematics Integrated Capability Center.

#### REFERENCES

1. Kaipio JP, Somersalo E. *Statistical and Computational Inverse Problems*, Vol. 160. Springer: New York, 2004.
2. Tarantola A. *Inverse Problem Theory and Methods for Model Parameter Estimation*. Society for Industrial Mathematics: Philadelphia, 2005.
3. Liu JS. *Monte Carlo Strategies in Scientific Computing*. Springer: New York, 2001.
4. Cui T, Fox C, O'Sullivan MJ. Bayesian calibration of a large-scale geothermal reservoir model by a new adaptive delayed acceptance Metropolis Hastings algorithm. *Water Resource Research* 2011; **47**:W10521.
5. Higdon D, Lee H, Holloman C. Markov chain Monte Carlo-based approaches for inference in computationally intensive inverse problems. In *Bayesian Statistics 7*, Bernardo JM, Bayarri MJ, Berger JO, Dawid AP, Heckerman D, Smith AFM, West M (eds). Oxford University Press: Oxford, UK, 2003; 181–197.
6. McKeague IW, Nicholls GK, Speer K, Herbei R. Statistical inversion of South Atlantic circulation in an abyssal neutral density layer. *Journal of Marine Research* 2005; **63**:683–704.
7. Haario H, Laine M, Lehtinen M, Saksman E, Tamminen J. Markov chain Monte Carlo methods for high dimensional inversion in remote sensing. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 2004; **66**:591–608.
8. Martin J, Wilcox LC, Burstedde C, Ghattas O. A stochastic Newton MCMC method for large-scale statistical inverse problems with application to seismic inversion. *SIAM Journal on Scientific Computing* 2012; **34**(3):A1460–A1487.
9. Marzouk YM, Najm HN, Rahn LA. Stochastic spectral methods for efficient Bayesian solution of inverse problems. *Journal of Computational Physics* 2007; **224**:560–586.
10. Marzouk YM, Najm HN. Dimensionality reduction and polynomial chaos acceleration of Bayesian inference in inverse problems. *Journal of Computational Physics* 2009; **228**:1862–1902.
11. Bayarri MJ, Berger J, Kennedy M, Kottas A, Paulo R, Sacks J, Cafeo J, Lin C, Tu J. Predicting vehicle crash-worthiness: validation of computer models for functional and hierarchical data. *Journal of the American Statistical Association* 2009; **104**:929–943.
12. Galbally D, Fidkowski K, Willcox KE, Ghattas O. Nonlinear model reduction for uncertainty quantification in large scale inverse problems. *International Journal for Numerical Methods in Engineering* 2008; **81**(12):1581–1608.
13. Lipponen A, Seppänen A, Kaipio JP. Electrical impedance tomography imaging with reduced-order model based on proper orthogonal decomposition. *Journal of Electronic Imaging* 2013; **22**:023008.
14. Lieberman C, Willcox KE, Ghattas O. Parameter and state model reduction for large-scale statistical inverse problems. *SIAM Journal on Scientific Computing* 2010; **32**(5):2523–2542.
15. Wang J, Zabarar N. Using Bayesian statistics in the estimation of heat source in radiation. *International Journal of Heat and Mass Transfer* 2005; **48**:15–29.
16. Sirovich L. Turbulence and the dynamics of coherent structures. Part 1: coherent structures. *Quarterly of Applied Mathematics* 1987; **45**:561–571.
17. Cover TM, Thomas JA. *Elements of Information theory*. Wiley-Interscience: New York, USA, 2006.
18. Arian E, Fahl M, Sachs EW. Trust-region proper orthogonal decomposition for flow control. *Technical Report ICASE-2000-25*, Institute for Computer Applications in Science and Engineering: Hampton, VA, USA, 2000.
19. Ravindran SS. Adaptive reduced-order controllers for a thermal flow system using proper orthogonal decomposition. *SIAM Journal on Scientific Computing* 2002; **23**(6):1924–1942.

20. Kunisch K, Volkwein S. Proper orthogonal decomposition for optimality systems. *ESAIM: Mathematical Modelling and Numerical Analysis* 2008; **42**(1):1–23.
21. Carlberg K, Farhat C. A compact proper orthogonal decomposition basis for optimization-oriented reduced-order models. *AIAA Paper* 2008; **5964**:10–12.
22. Stuart AM. Inverse problems: a Bayesian perspective. *Acta Numerica* 2010; **19**:451–559.
23. Grenander U, Miller MI. Representations of knowledge in complex systems. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 1994; **56**(4):549–603.
24. Hastings W. Monte Carlo sampling using Markov chains and their applications. *Biometrika* 1970; **57**:97–109.
25. Metropolis N, Rosenbluth AW, Rosenbluth MN, Teller AH, Teller E. Equation of state calculations by fast computing machines. *Journal of Chemical Physics* 1953; **21**:1087–1092.
26. Haario H, Saksman E, Tamminen J. An adaptive Metropolis algorithm. *Bernoulli* 2001; **7**(2):223–242.
27. Roberts GO, Rosenthal JS. Coupling and ergodicity of adaptive Markov chain Monte Carlo algorithms. *Journal of Applied Probability* 2007; **44**(2):458–475.
28. Girolami M, Calderhead B. Riemann manifold Langevin and Hamiltonian Monte Carlo methods. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 2011; **73**(2):123–214.
29. Liu JS, Chen R. Sequential Monte Carlo methods for dynamic systems. *Journal of the American Statistical Association* 1998; **93**(443):1032–1044.
30. Christen JA, Fox C. MCMC using an approximation. *Journal of Computational and Graphical Statistics* 2005; **14**(4):795–810.
31. Astrid P, Weiland S, Willcox KE, Backx T. Missing point estimation in models described by proper orthogonal decomposition. *IEEE Transactions on Automatic Control* 2008; **53**(10):2237–2251.
32. Barrault M, Maday Y, Nguyen NC, Patera AT. An “empirical interpolation” method: application to efficient reduced-basis discretization of partial differential equations. *Comptes Rendus Mathématique* 2004; **339**(9):667–672.
33. Chaturantabut S, Sorensen DC. Nonlinear model reduction via discrete empirical interpolation. *SIAM Journal on Scientific Computing* 2010; **32**(5):2737–2764.
34. Haasdonk B, Dihlmann M, Ohlberger M. A training set and multiple bases generation approach for parameterized model reduction based on adaptive grids in parameter space. *Mathematical and Computer Modelling of Dynamical Systems* 2011; **17**:423–442.
35. Amsallem D, Zahr M, Farhat C. Nonlinear model order reduction based on local reduced-order bases. *International Journal Numerical Methods Engineering* 2012; **92**:891–916.
36. Eftang JL, Stamm B. Parameter multi-domain ‘hp’ empirical interpolation. *International Journal for Numerical Methods in Engineering* 2012; **90**(4):412–428.
37. Peherstorfer B, Butnaru D, Willcox KE, Bungartz HJ. Localized discrete empirical interpolation method. *SIAM Journal on Scientific Computing* 2014; **36**(1):A168–A192.
38. Meyer M, Matthies HG. Efficient model reduction in non-linear dynamics using the Karhunen–Loeve expansion and dual-weighted-residual methods. *Computational Mechanics* 2003; **31**(1):179–191.
39. Cui T. Bayesian calibration of geothermal reservoir models via Markov chain Monte Carlo, The University of Auckland, Auckland, New Zealand, 2010.
40. Bui-Thanh T, Willcox KE, Ghattas O. Model reduction for large-scale systems with high-dimensional parametric input space. *SIAM Journal on Scientific Computing* 2008; **30**(6):3270–3288.
41. Patera AT, Rozza G. Reduced basis approximation and a posteriori error estimation for parametrized partial differential equations, 2007. MIT Pappalardo monographs in mechanical engineering (to appear), Copyright MIT (2006–2007).