LAWRENCE
LIVERMORE
NATIONAL
LABORATORY

LLNL-TR-740300

Efficient Stochastic Inversion Using Adjoint Models and Kernel-PCA

C. Thimmisetty , W. Zhao, X. Chen, C. H. Tong, J. A. White

October 19, 2017

This work performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344.

Efficient Stochastic Inversion Using Adjoint Models and Kernel-PCA

Charanraj A. Thimmisetty[1], Wenju Zhao[3], Xiao Chen[1],
Charles H. Tong[1], Joshua A. White[2]

[1] Center for Applied Scientific Computing,
Lawrence Livermore National Laboratory, Livermore, California, USA
[2] Atmospheric, Earth and Energy Division
Lawrence Livermore National Laboratory, Livermore, California, USA
[3]Department of Scientific Computing, Florida State University
Tallahassee, Florida, USA

This work was performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344.

October 18, 2017

Abstract

Performing stochastic inversion on a computationally expensive forward simulation model with a high-dimensional uncertain parameter space (e.g. a spatial random field) is computationally prohibitive even when gradient information can be computed efficiently. Moreover, the 'nonlinear' mapping from parameters to observables generally gives rise to non-Gaussian posteriors even with Gaussian priors, thus hampering the use of efficient inversion algorithms designed for models with Gaussian assumptions. In this paper, we propose a novel Bayesian stochastic inversion methodology, which is characterized by a tight coupling between the gradient-based Langevin Markov Chain Monte Carlo (LMCMC) method and a kernel principal component analysis (KPCA). This approach addresses the 'curse-of-dimensionality' via KPCA to identify a low-dimensional feature space within the high-dimensional and nonlinearly correlated parameter space. In addition, non-Gaussian posterior distributions are estimated via an efficient LMCMC method on the projected low-dimensional feature space. We will demonstrate this computational framework by integrating and adapting our recent data-driven statistics-on-manifolds constructions and reduction-through-projection techniques to a linear elasticity model.

Contents

Chapter 1

Introduction

The advent of computational science and engineering has enabled researchers to model complex and large-scale physical processes such as elasticity and plasticity simulation [1], climate and energy projection [2], subsurface flow and reactive transport [3], seismic wave propagation [4, 5], and power grid simulation and planning [6]. However, uncertainty in the model parameters renders corresponding modeling and simulation to be essentially stochastic. Applying uncertainty quantification (UQ) to improve model predictability usually requires modelers to solve an inverse problem (e.g., inverse UQ) by 'fusing' prior knowledge, modeling and simulation, and experimental observations. Deterministic approaches to solve the inverse problems, such as regularized weighted nonlinear least square methods, are capable of providing an optimal statistical estimator with the associated error bars for the inverse solutions. However, these approaches by their deterministic nature cannot produce solutions with a full description of the posterior probabilistic density functions (PDFs). Unlike deterministic inversion, stochastic inversion aims to provide a full PDF representation of the inverse solutions. This full PDF representation of the inverse solutions is critical to model prediction of the extreme events and event probabilities so that appropriate decisions can be made by decision makers according to the probability and risk associated with that specific event.

Alternatively, Bayesian inference provides a systematic framework for integrating prior knowledge and measurement uncertainties to compute detailed posteriors [7]. However, it can be computationally intractable [8] to compute the full PDF of the inverse solutions for each grid point (i.e., curse of dimensionality) resulting from the discretized parametric random field by solving a large-scale stochastic inversion problem. Moreover, unreasonable choices of prior knowledge due to ignorance of the abundant information embedded in the underlying dataset for model parameters can have

major effects on inferring posterior PDFs. In addition, the nonlinear mapping between the observables and parameters leads to non-Gaussian posteriors even with additive noise and Gaussian prior assumptions [8]. In general, it is computationally expensive to sample from the non-Gaussian and multi-modal posterior except for a few simple cases, where MCMC methods are considered as relevant techniques for sampling the non-standard posteriors. Despite the computational intensity encountered in MCMC, they have grown in rigor and sophistication with recent technical developments such as delayed rejection (DR) [9, 10], adaptive Metropolis (AM) [11, 12, 13], delayed rejection adaptive Metropolis (DRAM) [14], Langevin [15], stochastic Newton [8] and transport map accelerated MCMC [16].

The gradient-free MCMC methods, e.g., random walk MCMC, DR, AM, and DRAM, become computationally intractable as the dimension of the parameter space increases just moderately. Even though the gradient-enhanced MCMC algorithms such as Langevin [Citation] and stochastic Newton methods [Citation] have decreased the computational complexity of MCMC to $O(n^{1/3})$, expensive high-fidelity forward model runs, mesh-dependent high-dimensional parameter space, and multi-modal non-Gaussianity cause significant computational challenges in practice, thus making these algorithms not suitable for computationally intensive and large-scale real-world problems. One way to address the computational complexity of MCMC is through a construction of low-fidelity surrogate models using design of experiments (DOE) with the help of machine learning techniques, e.g., global polynomials [17, 18, 19], radial basis functions [20, 21], Gaussian processes [22], neural networks [23, 24], and/or proper orthogonal decomposition (POD) based reduced modeling. The use of low-fidelity model, based on surrogate and/or reduced-order modeling, greatly helps reduce the computational cost of the stochastic inversion. The low-fidelity model-based stochastic inversion, however, tend to produce entirely different inverse solutions or sub-optimal solutions compared to the true posterior obtained by high-fidelity model-based stochastic inversion.

Instead of performing forward model reduction, another way to reduce MCMC complexity is through control reduction by performing Bayesian inference in a low-dimensional subspace embedded in the high-dimensional parameter space while still controlling the high-fidelity forward model constrained onto the low-dimensional space. Karhunen-Loéve or principal component analysis (PCA) is a well-known choice for such parametric control dimension reduction. Traditionally, PCA is designed for the representation of linear correlation of the underlying data. Most of the realistic parametric random fields like channelized subsurface, however, exhibit non-linear correlations in the underlying data. The subspace obtained by PCA might not even cover the solution

domain. Furthermore, one has to perform exhaustive search to reach to the true posteriors due to the widely scattered reduced space represented by linear PCA-extracted subspace.

In general our method builds on unsupervised learning techniques to obtain relevant subspaces. Recent advances in unsupervised machine learning algorithms have provided ways to explore non-linear datasets using manifold learning techniques. For instance, Kernel PCA [25] (KPCA) as one such technique has been demonstrated to perform better clustering than linear PCA on complex non-linear data. Recently, Sarma [26] and Ma [27] demonstrated the efficiency and benefits of KPCA for deterministic inverse analysis and forward uncertainty propagation.

In the current work, we propose a novel framework for efficient stochastic inversion using adjoint partial differential equations (PDEs), automatic differentiation (AD), and KPCA. We will demonstrate this framework by integrating and adapting our recent data-driven statistics-on-manifolds constructions and reduction-through-projection approaches to the linear elasticity model. Essentially, a full statistical analysis in the high-dimensional ambient space spanned by the model parameters due to spatial discretization is computationally prohibitive. In addition, the model output is typically represented as very high-dimensional vectors defining the solution variables over the whole spatial discretization. Thus, we have a picture of an ambient space where each point is a high-dimensional vector obtained as an expensive model evaluation. The solution, however, is constrained: it does not occupy the whole ambient space, but merely a low-dimensional manifold within it. Examples of simple manifolds are: a spiral embedded in a 2-dimensional space, or a doughnut embedded in a 3-dimensional space.

Specially, we first derive a system of self-adjoint PDEs, which facilitates computation of the gradient with only one additional run besides the forward model run. The self-adjoint nature of the PDEs allows us to compute derivative of the cost functional with respect to the model parameters, i.e., first differentiation then discretization. Next, we use geostatistical methods such as the single normal equation simulation (SNESIM) algorithm [28] and intrinsic Gaussian process model [29] to generate a series of realizations of the complex structural model for building a prior model. Then, a low-dimensional feature space is obtained by performing non-linear control reduction in the high-dimensional ambient space through KPCA on the generated realizations. The feature random variables obtained from the KPCA are uncorrelated but not Gaussian, and Bayesian framework requires frequent sampling on these feature random variables. In this work, we sample them using polynomial chao expansion (PCE) constructed based on ICDF transformation. We then construct AD-based discretized adjoint model

(i.e., first discretization then differentiation) of the KPCA-based ICDF transformation and couple the discretized adjoint model with the continuous adjoint PDE model to obtain gradients of the objective functional with respect to the low-dimensional feature random variables. Bayesian inference is performed on the low-dimensional feature space using an efficient Langevin MCMC scheme performed on the high-fidelity forward models. The convergence rate of this KPCA-based and gradient-based stochastic inversion through MCMC has been greatly improved, thanks to the nonlinear control reduction with good classification and clustering retained. Unlike traditional machine learning problems, this process in each MCMC iteration step requires the projection of the low-dimensional feature space back to the high-dimensional parameter, since the high-fidelity forward models are functions of the mesh-dependent parameters. The projection is obtained by using both local fixed-point iteration and non-iterative algebra approaches with their efficiency and reliability discussed. Finally, this projection from the manifold back to parameter space gives us access to posterior PDFs of the mesh-dependent high-dimensional model parameters.

The remainder of this paper is organized as follows: Section 2 provides the mathematical framework of our procedure, which gives a detailed formulation of the proposed method. In Section 3, we apply the proposed method to identify material properties of geologically complex channelized subsurface. Section 4 gives some insights on advantages of KPCA and the implementation of the proposed method on the stochastic inversion. Finally, the conclusions are given in the Section 5 with an outline of the future work.

Chapter 2

Mathematical formulation

2.1 Preliminaries and notations

The following mathematical notation is used throughout the paper. Let the domain $\mathcal{D} \subset \mathbb{R}^d$, $d = 2, 3$ be a bounded, connected, open and Lipschitz continuous physical domain with a boundary $\Gamma = \partial\Omega$. Assume $\Gamma_{\mathcal{D}}$ and Γ_N are two subsets of Γ such that $\Gamma_{\mathcal{D}} \cap \Gamma_N = \emptyset$ and $\Gamma_{\mathcal{D}} \cup \Gamma_N = \Gamma$. Let the Dirichlet and Neumann boundary conditions be specified along the $\Gamma_{\mathcal{D}}$ and Γ_N, respectively. For an integer $m \geq 0$, we follow the classical notation of a standard Sobolev space $H^m(\mathcal{D})$ with norm $||\cdot||_m$ in accordance with Adams et al. [30]. Let Ω be a sample space associated with probability triplet $(\Omega, \mathcal{F}, \mathbb{P})$ where $\mathcal{F} \subset 2^{\Omega}$ is a σ-algebra of the events in Ω and $\mathbb{P}$ is the probability measure $\mathbb{P} : \mathcal{F} \to [0, 1]$. In addition, vector variables are represented by boldfaced letters, e.g. $\mathbf{x} = [x_1, x_2, \ldots, x_d]^T$.

In this paper we consider a system of stochastic PDEs with random coefficients. The PDE coefficients in Equation (2.1) $\boldsymbol{\mu}(\boldsymbol{x}, \omega) : \mathcal{D} \times \Omega \to \mathbb{R}$ and $\boldsymbol{\lambda}(\boldsymbol{x}, \omega) : \mathcal{D} \times \Omega \to \mathbb{R}$ are assumed to be random fields belonging to an infinite-dimensional probability space. The goal of the present work is to estimate these random coefficients based on prior knowledge and sparse measurements.

$$\begin{aligned} -\nabla \cdot (\mu(\nabla \mathbf{u} + \nabla \mathbf{u}^T)) + \nabla p &= \mathbf{0} \text{ in } \Omega \times \mathcal{D}, \\ \nabla \cdot \mathbf{u} + p/\lambda &= 0 \text{ in } \Omega \times \mathcal{D}, \\ (-p\mathbf{1} + \mu(\nabla \mathbf{u} + \nabla \mathbf{u}^T))\mathbf{n} &= \mathbf{h} \text{ on } \Omega \times \Gamma_D, \\ \mathbf{u} &= \mathbf{r} \text{ on } \Gamma_N. \end{aligned} \tag{2.1}$$

Here, $\boldsymbol{\mu}(\boldsymbol{x}, \omega)$ and $\boldsymbol{\lambda}(\boldsymbol{x}, \omega)$ are Lamé parameters; $\mathbf{u}(\boldsymbol{x}, \omega)$ and $p(\boldsymbol{x}, \omega)$ are displacement and pressure fields; $\boldsymbol{h}(\boldsymbol{x})$ and $\boldsymbol{r}(\boldsymbol{x})$ are prescribed traction and displacement vectors

on Ω_h and Ω_r, respectively; $\boldsymbol{n}$ is the unit outward normal on $\partial\Omega$; and the superscript T denotes the transpose. The set of PDEs in (2.1) represents the balance of linear momentum within an incompressible elastic solid with prescribed traction and displacement boundary conditions.

2.2 Discretization of the random field and kernel principal component analysis

Let $Y(\boldsymbol{x},\omega) := \ln(\boldsymbol{\mu}(\boldsymbol{x},\omega))$ be a random field, then the covariance function can be defined as $C_Y(\boldsymbol{x},\boldsymbol{y}) =< \tilde{Y}(\boldsymbol{x},\omega)\tilde{Y}(\boldsymbol{y},\omega) >_\omega$, where $\tilde{Y}(\boldsymbol{x},\omega) := Y(\boldsymbol{x},\omega) - < Y(\boldsymbol{x},\omega) >_\omega$ and $< . >_\omega$ is the expectation operator. Assuming C_Y is bounded, symmetric and positive definite, it can be represented as [31] (Charan, what does this mean?). Here, for the sake of simplicity we assume Lamé parameters are equal (Charan, what does 'equal' mean? Do you mean it is a constant?).

$$C_Y(\boldsymbol{x},\boldsymbol{y}) = \sum_{i=1}^{\infty} \gamma_i e_i(\boldsymbol{x}) e_i(\boldsymbol{y}), \tag{2.2}$$

where $\gamma_1 \geq \gamma_2 \geq \cdots$ are the eigenvalues , $e_i(\boldsymbol{x})$ and $e_j(\boldsymbol{y})$ are deterministic and mutually orthogonal functions i.e.,

$$\int_D e_i(\boldsymbol{x}) e_j(\boldsymbol{x})\, d\boldsymbol{x} = \delta_{ij}, \quad i,j \geq 1. \tag{2.3}$$

Using KarhunenLoève (KL) expansion, the random process $\overline{Y}(\boldsymbol{x},\omega)$ can be expressed in terms of $e_i(\boldsymbol{x})$ as

$$\overline{Y}(\boldsymbol{x},\omega) = \sum_{i=1}^{\infty} \xi_i(\omega)\sqrt{\gamma_i} e_i(\boldsymbol{x}), \tag{2.4}$$

where $\{\xi(\omega)_i\}$ are zero-mean and uncorrelated random variables, i.e., $< \xi_i(\omega) >= 0$ and $< \xi_i(\omega)\xi_j(\omega) >= \delta_{ij}$. The eigenvalues $\{\gamma_i\}$ and the eigenfunctions $\{f_n(\boldsymbol{x})\}$ are obtained by solving the following integral equation either analytically or numerically,

$$\int_D C_Y(\boldsymbol{x},\boldsymbol{y}) f(\boldsymbol{x})\, d\boldsymbol{x} = \gamma e(\boldsymbol{y}). \tag{2.5}$$

The attenuation of the eigenvalues $\{\gamma_i\}$ allows truncation of the infinite sum in Equation (2.4) up to N_R terms,

$$\overline{Y}(\boldsymbol{x},\omega) \approx \sum_{i=1}^{N_R} \xi_i(\omega)\sqrt{\gamma_i} e_i(\boldsymbol{x}), \tag{2.6}$$

where N_R is the stochastic dimension. The KL expansion is optimal [17] in the sense that it minimizes the mean-square error out of all possible orthonormal bases in $L^2(\mathcal{D}\times\Omega)$.

In practice, a closed form expression for the C_Y is rarely available. Instead, a numerical approximation to the $C_Y(\boldsymbol{x},\boldsymbol{y})$ is obtained using realizations of $Y(\boldsymbol{x},\omega)$ as:

$$C_Y(\boldsymbol{x},\boldsymbol{y}) \approx \frac{1}{M}\sum_{\omega}(Y(\boldsymbol{x},\omega) - < Y(\boldsymbol{x},\omega) >_\omega)(Y(\boldsymbol{y},\omega) - < Y(\boldsymbol{y},\omega) >_\omega)^T, \tag{2.7}$$

where M is number of realizations extracted from the random field $Y(\boldsymbol{x},\omega)$. Given C_Y, approximation to Equation (2.5) can be obtained using the Nystrom algorithm [32] as

$$\sum_{i=1}^{N_s} w_i C_Y(\boldsymbol{x}_i,\boldsymbol{y}) e(\boldsymbol{x}_i) = \gamma e(\boldsymbol{y}). \tag{2.8}$$

Here, N_s is the number of sample points where realizations $\boldsymbol{x}_i$'s are provided, and w_i's are weights of the quadrature rule. Assuming we have enough sample points and equal weights i.e., $w_i = \frac{1}{N_s}$, Equation (2.8) can be solved by simple eigen-decomposition of $C_Y(\boldsymbol{x}_i,\boldsymbol{y})$, for which principal component analysis (PCA) can be used to reduce the dimension.

The applications considered in this paper are related to elastic deformation of the subsurface due to self-weight or any external loads. The subsurface medium exhibits high levels of heterogeneity due to geological features such as channels. For the applications of interest, the prior models are generally generated based on their measurements (hard data) at a few sparse locations; and other data such as seismic logs (soft data) are often generated using geostatistical algorithms based on two point statistics such as kriging [33, 34, 35] or multi-point statistics (MPS). Here, we use MPS to generate elastic properties of the medium based on complex geological channelized structures.

The stochastic dimension of the prior model obtained using MPS is equal to the number of finite element grid points in the model. Equation (2.8), which is an equivalent of performing PCA of the covariance matrix, can be used to reduce the dimension.

However, in general, PCA can only obtain efficient embeddings for linear data. Recently, Sarma [26] and Ma [27] showed that KPCA can represent data better than PCA for the channelized subsurface structures.

We use two motivating examples shown in Fig 2.1 to demonstrate the efficiency of the KPCA. Figure 2.1 (top) depicts a classification problem and the objective is to classify the XOR dataset. It shows that KPCA with a second-order polynomial kernel can classify data perfectly, while PCA has 75% accuracy, thus demonstrating that KPCA has the capability to furnish a better representation for non-linear dataset. Figure 2.1 (bottom) shows another example [36], the goal of which is to reduce the dimension of a non-linear dataset that lies across a curve. It indicates that KPCA-based one-dimensional (1D) subspace is closer to true data than PCA-based 1D subspace. We take advantages of both dimension reduction and better feature representation properties of KPCA to increase the efficiency of stochastic inversion.

For the sake of completeness, we include a brief matrix derivation of KPCA below. More comprehensive derivations can be found in Schölkopf [37, 38] and Sarma [26]. Let N_R be a positive integer representing the dimension of the random field (in this case it is equal to the number of mesh grid points), and M be the number of observations of the random field. Given a set of discrete realizations $\{\mathbf{y}_l\}_{l=1}^{M}$ of the random field (which are also called snapshots), where each component (or snapshot) is $\mathbf{y}_l = [y_{1,l}, \ldots, y_{N,l}]^T \in \mathbb{R}^N$, $l = 1, 2, \ldots, M$, we define a linear or nonlinear mapping Φ as:

$$\Phi : \mathbb{R}^{N_R} \to \mathbb{R}^{N_F},\ y_l \to \Phi(y_l) \in \mathbb{R}^{N_F},\ l = 1, 2, \ldots, M, \tag{2.9}$$

where $\mathbb{R}^{N_F}$ is the new induced feature space. Here, $N_F \gg N_R$, and the feature space $\mathbb{R}^{N_F}$ in general contains much more information (that is, higher dimension) than the original space $\mathbb{R}^{N_R}$. For the the purpose of convenience, we also introduce matrix notations $\mathbf{Y} := [\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_M]$ and $\mathbf{\Phi} := [\Phi(\mathbf{y}_1), \Phi(\mathbf{y}_2), \ldots, \Phi(\mathbf{y}_M)]$. In addition, let $\mathbf{1}_M := \frac{1}{M}\mathbf{1}_{N_R \times M}$ be a matrix with all its elements equal to $\frac{1}{M}$; and let $\tilde{\mathbf{Y}} = \mathbf{Y} - \mathbf{Y}\mathbf{1}_M$ and $\tilde{\mathbf{\Phi}} := \mathbf{\Phi} - \mathbf{\Phi}\mathbf{1}_M$ be the centered matrix of $\mathbf{Y}$ and $\mathbf{\Phi}$, respectively.

In classical PCA, a discrete covariance matrix [39] is obtained as

$$\mathbf{C}_o := \frac{1}{M}\sum_{l=1}^{M} \tilde{\mathbf{y}}_l \tilde{\mathbf{y}}_l^T = \frac{1}{M}\tilde{\mathbf{Y}}\tilde{\mathbf{Y}}^T. \tag{2.10}$$

Here, the set $\{\tilde{\mathbf{y}}_l\}_{l=1}^{M}$ is a centered measurement vector given by $\tilde{\mathbf{y}}_l = \mathbf{y}_l - \bar{\mathbf{y}}$, where $\bar{\mathbf{y}} = \frac{1}{M}\sum_{l=1}^{M} \mathbf{y}_l$. Similar to the continuous version of the KL expansion with given mean and covariance kernel function, the KL expansion of the random fields for the discrete case can be characterized with following equation based on the Mercer's theorem:

$$\mathbf{y} = D_o \Lambda_o^{1/2} \boldsymbol{\xi} + \mathbf{Y}\mathbf{1}_1, \tag{2.11}$$

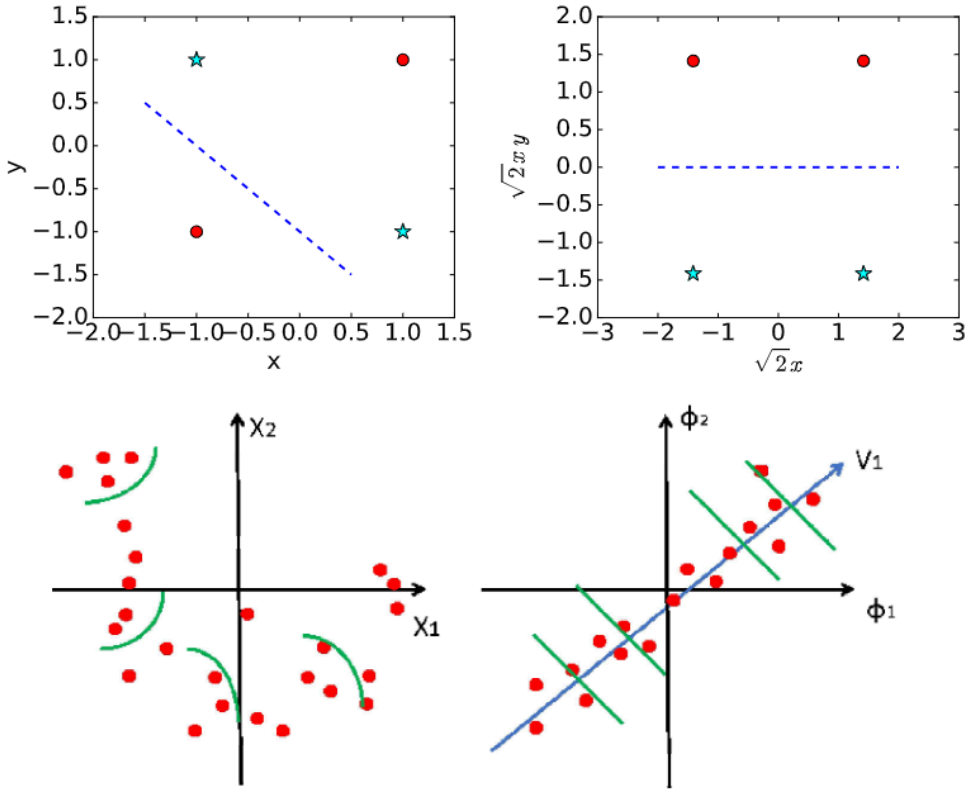


Fig. 2.1. KPCA motivating examples: XOR data classification (top) and non-linear dimension reduction of a non-linear dataset (bottom).

where D_o is a matrix of eigenvectors associated with $\mathbf{C}_o$; Λ_o is a diagonal matrix of the eigenvalues of $\mathbf{C}_o$; $\boldsymbol{\xi} = [\xi_1, \xi_2, \cdots, \xi_{N^R}]^T \in \mathbb{R}^{N_R}$ is a column random vector with statistical properties $\mathbb{E}[\xi_i \xi_j] = \delta_{i,j}$ and $\mathbb{E}[\xi_i] = 0$. A nonlinear choice for the Φ such as radial basis functions leads to the nonlinear form of PCA. Next, we compute the centralized form of the feature vectors $\{\tilde{\Phi}, (\mathbf{y}_l)\}_{l=1}^M$ where $\tilde{\Phi}(\mathbf{y}_l) = \Phi(\mathbf{y}_l) - \bar{\Phi}$, $\bar{\Phi} = \frac{1}{M}\sum_{l=1}^M \Phi(\mathbf{y}_l)$. Similar to PCA, we have the following discrete covariance after the nonlinear mapping

$$\mathbf{C}_f = \frac{1}{M}\sum_{j=1}^{M} \tilde{\Phi}(\mathbf{y}_j)\tilde{\Phi}(\mathbf{y}_i)^T = \frac{1}{M}\tilde{\mathbf{\Phi}}\tilde{\mathbf{\Phi}}^T. \tag{2.12}$$

Since N_F is usually much larger than N_R, it is infeasible in practice to perform PCA on the feature space due to the very high dimensionality of the covariance matrix. For

instance, for the polynomial kernel $(\mathbf{x} \cdot \mathbf{y})^d$ of order d, dimension of the feature space will be [37]

$$N_F = \frac{(N_R + d - 1)!}{d!(N_R - 1)!}. \tag{2.13}$$

Alternatively, the nonlinear mapping can be seen as a kernel map, thus allowing us to handle the high dimensionality by using a technique called "kernel trick". A kernel trick introduces a virtual mapping Φ, from beginning to the end, where the mapping Φ only acts as an intermediate functional, resulting in smaller dimension for $\mathbf{C}_f$. The eigen-problem of the covariance matrix $\mathbf{C}_f$ in the feature space is now given as:

$$\mathbf{C}_f \mathbf{V}_f = \mathbf{V}_f \Lambda_f. \tag{2.14}$$

Here, $\mathbf{V}_f$ is the matrix of eigenvectors and Λ_f is a diagonal eigenvalue matrix. The relationship between the eigenvectors $\{\mathbf{v}_l\}$ of $\mathbf{V}_f$ and the data set of $\{\tilde{\Phi}(\mathbf{y}_l)\}$, can be written as

$$\mathbf{C}_f \mathbf{v}_l = \frac{1}{M} \sum_{j=1}^{M} \tilde{\Phi}(\mathbf{y}_j) \tilde{\Phi}(\mathbf{y}_i)^T, \quad \mathbf{v}_l = \frac{1}{N_R} \sum_{j=1}^{M} (\tilde{\Phi}(\mathbf{y}_i)^T \mathbf{v}_l) \tilde{\Phi}(\mathbf{y}_j) = \gamma_l \mathbf{v}_l, \tag{2.15}$$

which shows that the eigenvectors $\{\mathbf{v}_l\}$ are elements in the space spanned by $\tilde{\Phi}(\mathbf{y}_l)$, $l = 1, \ldots, M$.

Let $\boldsymbol{\alpha} = [\boldsymbol{\alpha}_1, \ldots, \boldsymbol{\alpha}_M]$ with $\boldsymbol{\alpha}_l = [\alpha_{l,1}, \alpha_{l,2}, \ldots, \alpha_{l,N_R}]^T$, and eigen matrix $\mathbf{V}_f = \tilde{\boldsymbol{\Phi}} \boldsymbol{\alpha}$ where each component of the eigenvector $\mathbf{v}_l = \sum_{j=1}^{N_R} \alpha_{l,j} \tilde{\Phi}(y_i) = \tilde{\boldsymbol{\Phi}} \boldsymbol{\alpha}_l$. Substituting this into Equation (2.15) leads to

$$\mathbf{C}_f \tilde{\boldsymbol{\Phi}} \boldsymbol{\alpha} = \tilde{\boldsymbol{\Phi}} \boldsymbol{\alpha} \Lambda_f. \tag{2.16}$$

Using the definition of $\mathbf{C}_f$ from Equation (2.12) and multiplying both sides by $\tilde{\Phi}^T$, and further setting $K_c = \tilde{\Phi}^T \tilde{\Phi}$, we have

$$\frac{1}{M} K_c^2 \mathbf{V} = K_c \boldsymbol{\alpha} \Lambda_f. \tag{2.17}$$

Assuming K_c is a nonsingular matrix, the equation above is equivalent to the following kernel eigenvalue problem

$$\frac{1}{M} K_c \mathbf{V} = \boldsymbol{\alpha} \Lambda_f, \tag{2.18}$$

where K_c is a matrix of $M \times M$. This 'kernel trick' allows us to perform KPCA in the high dimensional feature space, with almost similar computational expense as PCA.

Irrespective of the dimension of the feature space, we just need to perform eigen-decomposition on a relatively small space $\mathbb{R}^M$, which is independent of the selection of the nonlinear mapping and the feature space.

Solving Equation (2.18) leads to the eigenvector matrix $\mathbf{V}$, and the corresponding $\mathbf{V}_f$ in Equation (2.14) can be retrieved using,

$$\mathbf{V}_f = \tilde{\mathbf{\Phi}}\mathbf{V}. \tag{2.19}$$

Here, $\mathbf{V}_f$ has the property that

$$\mathbf{V}_f^T\mathbf{V}_f = \mathbf{V}^T\tilde{\Phi}^T\tilde{\Phi}\mathbf{V} = \mathbf{V}^T K_c \mathbf{V} = M\Lambda_f. \tag{2.20}$$

Using the same notation of $\mathbf{V}_f$, we have the orthonormal eigenvector matrix

$$\mathbf{V}_f = \frac{1}{\sqrt{M}}\tilde{\Phi}\mathbf{V}\Lambda_f^{-1/2}. \tag{2.21}$$

Let $K = \Phi^T\Phi$, then the centered K_c can be easily obtained using

$$\begin{aligned} K_c &= (\Phi - \bar{\Phi})^T(\Phi - \bar{\Phi}) = (\Phi - \Phi 1_{N_R})^T(\Phi - \Phi 1_{N_R}) \\ &= \Phi^T\Phi - \Phi^T\Phi 1_{N_R} - 1_{N_R}^T\Phi^T\Phi + 1_{N_R}\Phi^T\Phi 1_{N_R} \\ &= K - K1 - 1K + 1K1 \end{aligned}$$

Thus, we have the KL expansion in the feature space as

$$\mathbf{Y}_f = \mathbf{V}\Lambda^{1/2}\boldsymbol{\xi} + \bar{\Phi} = \frac{1}{\sqrt{M}}\tilde{\Phi}\mathbf{V}\Lambda_F^{-1/2}\Lambda^{1/2}\boldsymbol{\xi} + \bar{\Phi} = \frac{1}{\sqrt{M}}\tilde{\Phi}\mathbf{V}\boldsymbol{\xi} + \bar{\Phi}, \tag{2.22}$$

where $\boldsymbol{\xi} = [\xi_1, \ldots, \xi_{N_R}]^T$ is a random vector with properties $\mathbb{E}[\xi_i] = 0, \mathbb{E}[\xi_i\xi_j] = \delta_{i,j}$. The polynomial kernel and Gaussian kernel defined below are frequently used in practice, which are given by

$$k(\mathbf{x}, \mathbf{y}) = c + (\mathbf{x} \cdot \mathbf{y})^d, \ d \geq 1, \tag{2.23}$$

$$k(\mathbf{x}, \mathbf{y}) = \exp(-\tfrac{\|\mathbf{x}-\mathbf{y}\|^2}{\sigma}), \ \sigma > 0, \tag{2.24}$$

respectively. Kernel functions directly calculate dot product in the space of $\mathbb{R}^F$ using elements in the input space $\mathbb{R}^{N_R}$. Since there is no actual mapping of $\Phi(y)$, kernels play the role of the intermediate functional.

Although stochastic inversion is performed in the feature space, our interest is to obtain the snapshots from the posterior in the original space $\mathbb{R}^{N_R}$. In order to achieve

this, a pre-imaging problem is solved to project snapshots from the feature space back to the original space. In general, due to the non-linearity of the mapping Φ, neither existence nor uniqueness of the pre-image is guaranteed.

Pre-imaging problem involves solving the following optimization problem [37],

$$\min_{\mathbf{y}} \rho(\mathbf{y}) = ||\Phi(\mathbf{y}) - \boldsymbol{Y}||^2, \tag{2.25}$$

where the $\mathbf{y} \in \mathbb{R}^{N_F}$ and $\boldsymbol{Y} \in \mathbb{R}^{N_R}$ are points in the feature space and original space, respectively, and $\|\cdot\|$ is the Euclidean norm. The above minimization problem can be reduced to the following iterative fixed point problem [26, 37, 40]

$$\mathbf{y}^{k+1} = \frac{\sum_{l=1}^{N_R} \beta_i \sum_{j=1}^{d} j(\mathbf{y}_i \cdot y^k)^{j-1} \mathbf{y}_i}{\sum_{l=1}^{N_R} \beta_i \sum_{j=1}^{d} j(\mathbf{y}_i \cdot y^k)^{j-1}}. \tag{2.26}$$

2.3 Mapping non-Gaussian feature random variables to Gaussian random variables

Let $\boldsymbol{\xi}^d$ be the discrete observations of $\boldsymbol{\xi}$ obtained from the measurements of the snapshots $\{\mathbf{y}_l\}_{l=1}^M$. Letting $\mathbf{Y}_f = \boldsymbol{\Phi}$ and multiplying both sides of Equation (2.22) by $\boldsymbol{\Phi}^T$, we obtain

$$\tilde{\boldsymbol{\Phi}} = \boldsymbol{\Phi} - \boldsymbol{\Phi}\mathbf{1}_M = \frac{1}{\sqrt{M}}\tilde{\Phi}\mathbf{V}\boldsymbol{\xi}^d \Rightarrow K_c = \frac{1}{\sqrt{M}} K_c \mathbf{V}\boldsymbol{\xi}^d. \tag{2.27}$$

Assuming K_c is nonsingular, we have

$$\mathbf{V}\boldsymbol{\xi}^d = \sqrt{M}\mathbf{1}_M \tag{2.28}$$

which can be solved using a least-squares method or singular value decomposition (SVD).

Random variables $\boldsymbol{\xi}^d$ computed from Equation 2.28 act as a prior distribution for the Bayesian inversion framework. In general, $\boldsymbol{\xi}^d$ are non-Gaussian, uncorrelated and dependent random variables, which may complicate the Bayesian inversion procedures (e.g. more frequent sampling from their distributions).

Determination of a unique map from the dependent $\boldsymbol{\xi}^d$ to standard independent random variable space $\boldsymbol{\eta}$ is an active research and development problem. One way to achieve a non-unique mapping is using iso-probabilistic mappings such as the generalized Nataf transformation [41] and Rosenblatt transformation [42]. However, these transformations require information such as conditional distributions, which are hard

to construct from the limited observations. Hence, we assume $\{\xi_l^d\}_{l=1}^M$ are independent similar to [43, 44] and to facilitate the sampling, we construct a polynomial chaos expansion (PCE) for each ξ_l.

PCE, originally introduced by Wiener [45, 17], represents any L^2 random variable as a summation of series of polynomials over the centered normalized Gaussian variables. Since $\boldsymbol{\xi}^d$ are obtained from the measurements, they belong to L^2 space, thus allowing us to represent each component of $\{\xi_l^d\}_{l=1}^M$ obtained from Equation (2.28) using PCE as

$$\xi_l^d = \sum_{n=0}^{\infty} c_{n,l} \Psi_n(\eta_l(\omega)), l = 1, 2, ..., \tag{2.29}$$

where η_l's are i.i.d. standard Gaussian random variables, $\Psi_n(\eta_l(\omega))$ are Hermite polynomials, and $c_{n,l}$ are real valued deterministic coefficients. The associated orthogonal system $\{\Psi_n(\eta)\}_{n\in\mathbb{N}}$ forms the homogeneous polynomial chaos basis for the space $L^2(\Omega, \sigma(\eta), \mathbb{P})$. The coefficients in the equation above can be computed using Bayesian inference [46] or using non-intrusive projection method [47]. The method described below is based on empirical cumulative distributions to evaluate the coefficients.

Evaluating PCE coefficients using empirical inverse cumulative distribution functions

We use a projection method [48] to find a continuous parameterized representation similar to Equation (2.29) based on the discrete $\boldsymbol{\xi}^d$. Let $\{\eta_l\}$ be a standard Gaussian random variable, then by matching the cumulative density function (cdf) of ξ_l^d and η_l, each component of ξ_l can be expressed in terms of random variables η_l by following non-linear mapping:

$$\xi_l^d = F_{\xi_l^d}^{-1} \circ F_{\eta_l}(\eta_l), \tag{2.30}$$

where $F_{\xi_l^d}$ and F_{η_l} denote the cdfs of ξ_l^d and η_l. respectively. The coefficients of the PCE are then computed using the projection of $F_{\xi_i^d}^{-1} \circ F_{\eta_l}$ on orthonormal chaos basis system,

$$c_{n,l} =< \xi_l^d, \Psi_n >= \int_\Omega F_{\xi_l^d}^{-1} \circ F_{\eta_l} \Psi_n d\mathbb{P}_\eta(\omega), \tag{2.31}$$

However, the cdf $F_{\xi_l^d}$ is not known and it is to be estimated by the empirical cdf [49] based on the discrete observations of $\boldsymbol{\xi}^d$. Empirical cdf ($\tilde{F}_{\xi_l^d}$) of ξ_l^d can be estimated from sampling using,

$$\tilde{F}_{\xi_i^d}(x) = \frac{1}{M} \sum_{k=1}^{M} I(\xi_l^{d^{(k)}} \leq x), \tag{2.32}$$

where $I(A)$ is the indicator function of event A. We then introduce the following approximation

$$F^{-1}_{\xi_i^d} \sim \tilde{F}^{-1}_{\xi_i^d}, \text{ where } \tilde{F}^{-1}_{\xi_i^d} : [0,1] \to \mathbb{R} \tag{2.33}$$

which is uniquely defined as

$$\tilde{F}^{-1}_{\xi_i^d}(y) = \min\{x \in \{\xi^{(k)}_{l^d}\}_{k=1}^M; \tilde{F}_{\xi_i^d}(x) \geq y\}. \tag{2.34}$$

Then the coefficients of the polynomial chaos expansion can be computed using a numerical integration. Instead of using the indicator functions, we use kernel density estimation [50] to construct the empirical cdf,

$$\tilde{f(\xi)} = \frac{1}{M}\sum_{l=1}^{M} K_h(\xi - \xi_l), \tag{2.35}$$

where $K_h(\cdot)$ is the kernel function.

$$c_{n,l} =< \xi_l, \Psi_n >= \int_\Omega F^{-1}_{\xi_l^d} \circ F_{\eta_l^d} \Psi_n d\mathbb{P}_\eta(\omega), = \int_\Omega F^{-1}_{\xi_l^d} \circ F_{\eta_l^d} \Psi_n \frac{e^{-\eta^2/2}}{\sqrt{2\pi}} dx \tag{2.36}$$

The coefficients $c_{n,l}$ can be efficiently calculated using the Gauss-Hermite quadrature rules.

2.4 Bayesian Inference of the inverse problem

Bayesian inference treats the parameters $\boldsymbol{\mu} = \mu(\boldsymbol{x}), \boldsymbol{\lambda} = \mu(\boldsymbol{x})$ of the forward model (2.1) as a random process. Instead of performing Bayesian inference with respect to the parameters $\boldsymbol{\mu}$, $\boldsymbol{\lambda}$ of (2.1), we perform the inference in the parameterized feature space of $\boldsymbol{\eta}$. We denote the stochastic elasticity forward model (2.1) as $\mathbf{u} = f(\boldsymbol{\eta})$, which describes the relationship between the observed output state $\mathbf{u}_{obs}$ and the uncertain model parameters $\boldsymbol{\eta}$. As such, the posterior distribution from the Bayesian inference can be expressed as

$$\pi_{posterior}(\boldsymbol{\eta}) := \pi(\boldsymbol{\eta}|\mathbf{u}_{obs}) \propto \pi_{prior}(\boldsymbol{\eta})\pi_{likelihood}(\mathbf{u}_{obs}|\boldsymbol{\eta}) \tag{2.37}$$

The model above allows us to fuse modeling and measurement errors into the inversion framework. Unlike deterministic inversion, the expression (2.37) provides a probabilistic characterization of the solution [8] for the inverse problem. In this context, the likelihood function $\pi_{likehood}(\mathbf{u}_{obs}|\boldsymbol{\eta})$ is a conditional probability of the

model outputs with given model parameters $\boldsymbol{\eta}$. Also, the prior probability density function (pdf) $\pi_{prior}(\boldsymbol{\eta})$ allows us to infuse prior knowledge into the model. In our case, the prior density function π_{prior} is a multivariate Gaussian of the form:

$$\pi_{prior}(\boldsymbol{\eta}) \propto \exp(-\frac{1}{2}\|\boldsymbol{\eta} - \bar{\boldsymbol{\eta}}\|^2_{\Gamma^{-1}_{prior}}). \tag{2.38}$$

The simplification above is possible due to the independence of the $\boldsymbol{\eta}$. Specifically, the covariance matrix Γ_{prior} is an identity matrix and $\bar{\boldsymbol{\eta}}$ is a zero vector. The representation of likelihood function forms the core part to characterize the posterior density function $\pi_{posterior}$. In the limiting case where the measurement and the model are exactly unbiased, the Bayesian model can easily be reduced to

$$\pi_{posterior}(\boldsymbol{\eta}) := \pi(\boldsymbol{\eta}|\mathbf{u}_{obs}) \propto \pi_{prior}(\boldsymbol{\eta}). \tag{2.39}$$

For further simplification, we assume that the error between the measurement and the model is unbiased and additive; and the noise follows a Gaussian distribution. This leads to following expression for the likelihood function

$$\pi_{likelihood}(\mathbf{u}_{obs}|\boldsymbol{\eta}) \propto \exp(-\frac{1}{2}\|f(\boldsymbol{\eta}) - \mathbf{u}_{obs}\|^2_{\Gamma^{-1}_{noise}}). \tag{2.40}$$

We note that our procedure is still valid for other choices of likelihood functions. Our particular choice for likelihood is due to limited information on measurement and modeling errors. The choice of the likelihood function of the form Equation 2.40 leads to following log-likelihood function,

$$-\log(\pi(\mathbf{u}_{obs}|\eta)) = \frac{1}{2}\|f(\boldsymbol{\eta}) - \mathbf{u}_{obs}\|^2_{\Gamma^{-1}_{noise}}, \tag{2.41}$$

and the corresponding posterior density can be derived as

$$\pi_{posterior}(\boldsymbol{\eta}) \propto \exp(V(\boldsymbol{\eta})), \tag{2.42}$$

where $V(\boldsymbol{\eta})$ is given by

$$V(\boldsymbol{\eta}) := \frac{1}{2}\|f(\boldsymbol{\eta}) - \mathbf{u}_{obs}\|^2_{\Gamma^{-1}_{noise}} + \frac{1}{2}\|\boldsymbol{\eta} - \bar{\boldsymbol{\eta}}\|^2_{\Gamma^{-1}_{prior}}. \tag{2.43}$$

Due to the non-linear relation between the parameters $\boldsymbol{\eta}$ and the measurements, direct sampling from the posterior is not possible even with the chosen likelihood function [8]. MCMC methods provide a systematic way to sample from the corresponding posteriors.

2.5 Gradient-based adjoint MCMC

MCMC approaches require many simulations of the forward models, leading to computational intractability when the forward models are expensive to evaluate. Also, when the dimension of the parameter space is high, MCMC methods require many forward simulations for exploring the high dimensional probability space. In order to accelerate the MCMC sampling, adjoint of the posterior density function is computed and the Langevin MCMC (LMCMC) method is used instead of the Metropolis Hastings MCMC (MHMCMC) based on random walk. Theoretically, LMCMC has a computational complexity of $O(n^{1/3})$, while MHMCMC has the complexity of $O(n)$ where n is the dimension of the inference parameters. LMCMC considers the following overdamped Langevin Ito diffusion process,

$$dX = \nabla \log \pi_{posterior}(X)dt + \sqrt{2}dW. \tag{2.44}$$

The probability distribution $\rho(t)$ of $X(t)$ approaches a stationary distribution, which is invariant under diffusion and $\rho(t)$ approaches the true posterior($\rho_\infty = \pi_{poster}$) asymptotically. Approximate sample paths of the Langevin diffusion can be generated by many discrete-time methods. Using the Euler-Maruyama method with a fixed time step $\tau > 0$, the above Equation can be written as,

$$X_{k+1} = X_k + \tau \nabla \log \pi(X_k) + \sqrt{2\tau}\xi_k \tag{2.45}$$

where each ξ_k is an independent draw from a multivariate normal distribution on $\mathbb{R}^{N_F}$ with mean 0 and identity covariance matrix.

This proposal is accepted or rejected similar to the Metropolis-Hasting algorithm using α,

$$\alpha = \min\{1, \frac{\pi(X_{k+1})q(X_k|X_{k+1})}{\pi(X_k)q(X_{k+1}|X_k)}\} \tag{2.46}$$

where

$$q(x'|x) \propto \exp(-\frac{1}{4\tau}\|x' - x - \tau \nabla \log \pi(x)\|_2^2) \tag{2.47}$$

2.6 Adjoint Information of the posterior density function

In this section, we introduce a technique to compute the gradient information of the negative logarithm of the posterior function with respect to the random parameters

$\boldsymbol{\eta}$,

$$V(\boldsymbol{\eta}) := \tfrac{1}{2}\|f(\boldsymbol{\eta}) - \mathbf{u}_{obs}\|^2_{\Gamma^{-1}_{noise}} + \tfrac{1}{2}\|\boldsymbol{\eta} - \bar{\boldsymbol{\eta}}\|^2_{\Gamma^{-1}_{prior}} \tag{2.48}$$

$$= V_1(\boldsymbol{\eta}) + V_2(\boldsymbol{\eta}), \tag{2.49}$$

where $V_1(\boldsymbol{\eta}) = \frac{1}{2}\|f(\boldsymbol{\eta}) - \mathbf{u}_{obs}\|^2_{\Gamma^{-1}_{noise}}$ and $V_2(\boldsymbol{\eta}) = \frac{1}{2}\|\boldsymbol{\eta} - \bar{\boldsymbol{\eta}}\|^2_{\Gamma^{-1}_{prior}}$. It is nontrivial to obtain the functional derivative of $V(\boldsymbol{\eta})$. Here we use the adjoint model and automatic differentiation to compute the gradients. Using the mathematical derivations in the preceding sections, the relationship between the variables $\boldsymbol{\eta}, \boldsymbol{\xi}, \mathbf{y}, \mu, \lambda, \mathbf{u}$ can be summarized as,

$$\boldsymbol{\eta} \xrightarrow{\text{PCE}} \boldsymbol{\xi} \xrightarrow{\text{Pre-image}} \mathbf{y} \xrightarrow{\exp} \mu, \lambda \xrightarrow{\text{forward model}} \mathbf{u}. \tag{2.50}$$

The objective functional V can be expressed in terms of $\boldsymbol{\eta}$ by

$$V : \mathbb{R}^r \to \mathbb{R} \tag{2.51}$$

$$\boldsymbol{\eta} \to \tfrac{1}{2}(f(\boldsymbol{\eta}) - \mathbf{u}_{obs}, \Gamma^{-1}_{noise}(f(\boldsymbol{\eta}) - \mathbf{u}_{obs})) + \tfrac{1}{2}(\boldsymbol{\eta} - \bar{\boldsymbol{\eta}}, \Gamma^{-1}_{prior}(\boldsymbol{\eta} - \bar{\boldsymbol{\eta}})) \tag{2.52}$$

The second part of $V(\boldsymbol{\eta})$ is a quadratic form in the parameters $\boldsymbol{\eta}$. The expression for the gradient of $V_2(\boldsymbol{\eta})$ can directly be obtained as

$$\nabla_{\eta} V_2(\boldsymbol{\eta}) = \Gamma^{-1}_{prior}(\boldsymbol{\eta} - \bar{\boldsymbol{\eta}}) \tag{2.53}$$

To derive the gradient of V_1, we follow the procedure similar to Giering *et al.* [51]. Consider the Taylor expansion V_1 with respect to the control variables at a given point $\boldsymbol{\eta}_0$

$$V_1(\boldsymbol{\eta}) = V_1(\boldsymbol{\eta}_0) + (\nabla_{\eta} V_1(\boldsymbol{\eta}_0), \boldsymbol{\eta} - \boldsymbol{\eta}_0) + O(|\boldsymbol{\eta} - \boldsymbol{\eta}_0|), \tag{2.54}$$

or in short terms,

$$\delta V_1 = (\nabla_{\eta} V_1(\boldsymbol{\eta}_0), \delta\boldsymbol{\eta}). \tag{2.55}$$

We use the shorthand notation above whenever linear approximations are involved. Suppose V_1 is sufficiently regular, then for each parameter vector $\boldsymbol{\eta}_0$ variation of Y can be approximated using

$$\delta Y = A(\boldsymbol{\eta}_0)\delta\boldsymbol{\eta}. \tag{2.56}$$

where $A(\boldsymbol{\eta}_0)$ denotes the Jacobian of H at $\boldsymbol{\eta}_0$. Due to the symmetry of the inner product, applying the product rule of differentiation yields

$$\delta V_1 = (\Gamma^{-1}_{noise}(f(\boldsymbol{\eta}) - \mathbf{u}_{obs}), \nabla_{\eta} f(\boldsymbol{\eta}_0)\delta\boldsymbol{\eta}). \tag{2.57}$$

Using the definition of the adjoint operator A^* :

$$(v, Aw) = (A^*v, w), \tag{2.58}$$

we obtain

$$\delta V_1 = ((\nabla_\eta f(\boldsymbol{\eta}_0))^T \Gamma_{noise}^{-1}(f(\boldsymbol{\eta}) - \mathbf{u}_{obs}), \delta\boldsymbol{\eta}). \tag{2.59}$$

Therefore, according to the definition of gradient, the gradient of the V_1 with respect to $\boldsymbol{\eta}$ is

$$\nabla_\eta V_1(\boldsymbol{\eta}_0) = (\nabla_\eta f(\boldsymbol{\eta}_0))^T \Gamma_{noise}^{-1}(f(\boldsymbol{\eta}) - \mathbf{u}_{obs}), \tag{2.60}$$

Since the function $f := f_1 \circ f_2 \circ f_3 \circ f_4$, applying the chain rule yields

$$f' := f_1' \circ f_2' \circ f_3' \circ f_4' \tag{2.61}$$

$$= \nabla_{\boldsymbol{\lambda},\mu}\mathbf{u} \nabla_{\mathbf{y}} \lambda \nabla_{\boldsymbol{\xi}} \mathbf{y} \nabla_{\boldsymbol{\eta}} \boldsymbol{\xi}. \tag{2.62}$$

The gradient information can be rewritten as

$$\nabla_\eta V_1(\boldsymbol{\eta}_0) = (\nabla_{\boldsymbol{\eta}} \boldsymbol{\xi})^T (\nabla_{\boldsymbol{\xi}} \mathbf{y})^T (\nabla_{\mathbf{y}} \boldsymbol{\lambda})^T (\nabla_{\boldsymbol{\lambda},\mu}\mathbf{u})^T \Gamma_{noise}^{-1}(f(\boldsymbol{\eta}) - \mathbf{u}_{obs}), \tag{2.63}$$

The linear operator $\nabla_{\boldsymbol{\lambda},\mu}\mathbf{u}$ represents the tangent linear model of the forward problem and its adjoint operator is $(\nabla_{\boldsymbol{\lambda},\mu}\mathbf{u})^T$. Both operators depend on the point $\boldsymbol{\eta}_0$ at which the model is linearized. The linear operator $(\nabla_{\boldsymbol{\eta}} \boldsymbol{\xi})^T$ represents the adjoint model of the PCE, and $(\nabla_{\boldsymbol{\xi}} \mathbf{y})^T$ represents the adjoint model of the pre-image iteration mapping.

Following a similar procedure, the adjoint model $(\nabla_{\lambda,\mu}\mathbf{u})^T$ can easily be obtained as detailed in [52]. The PCE mapping in Equation (2.29) and the pre-image mapping methods are continuous smooth mappings. The adjoint models for these mappings are obtained with automatic differentiation [53].

2.7 Algorithms

In this section, we summarize the above derivations into two simple algorithms to facilitate the implementation of the proposed methodology.

Algorithm 1 Computation of posterior density function and gradients

Read the snapshots $\{\mathbf{Y}_l\}_{l=1}^{M}$ of the parameters μ, λ
Compute KPCA reduced model using Equation (2.22)
Parameterize the random variables $\boldsymbol{\xi}$ with PCE using Equation (2.29)
Compute prior density function π_{prior} as defined by Equation (2.38)
Compute likelihood function $\pi_{likelihood}$ as defined by Equation (2.40)
Compute the posterior density function using Equation (2.39)
Compute the gradient of the cost functional with respect to parameters λ and μ using adjoint model
Compute the gradient of the cost functional in the feature space using automatic differentiation

Algorithm 2 Posterior sampling using Langevin MCMC framework

Choose initial parameters $\boldsymbol{\eta}_0$
Compute $\pi_{posterior}(\boldsymbol{\eta}_0)$ using algorithm 1
for l=1 to N **do**
  Draw sample y from the proposal density function
  Compute $\pi_{posterior}(y)$ using algorithm 1
  Compute $\alpha(\boldsymbol{\eta}_l, y) = \min\{1, \frac{\pi_{posterior}(y)q(y|\boldsymbol{\eta}_l)}{\pi_{posterior}(\boldsymbol{\eta}_l)q(\boldsymbol{\eta}_l|y)}\}$, where $q(y|\boldsymbol{\eta}_l)$ and $q(\boldsymbol{\eta}_l|y)$ are computed using Equation 2.47
  Draw $u \sim U([0,1])$
  if $u < \alpha(\boldsymbol{\eta}_l, y)$ **then**
    Accept : Set $\boldsymbol{\eta}_{l+1} = y$
  else
    Reject : Set $\boldsymbol{\eta}_{l+1} = \boldsymbol{\eta}_k$
  end if
end for

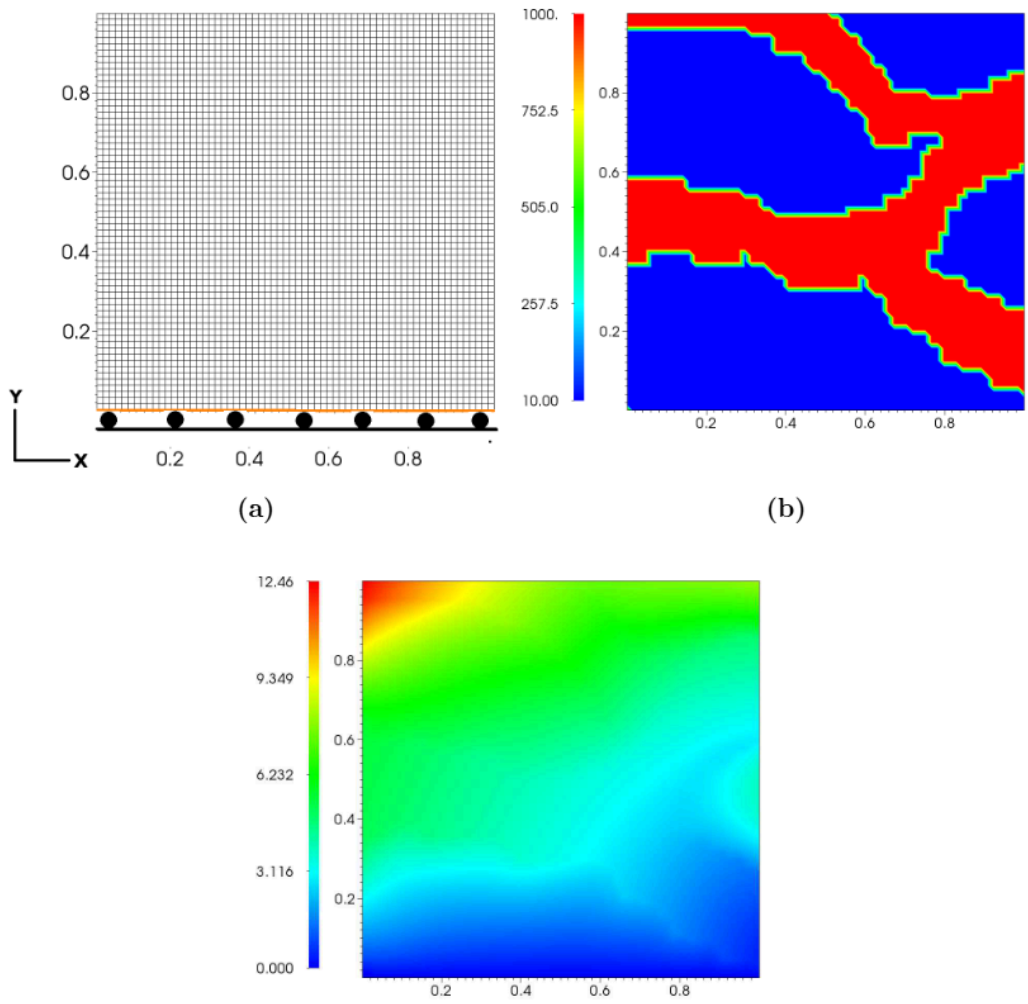
Chapter 3

Numerical Simulations

In this section, we demonstrate the computational efficiency of the proposed method for the stochastic inversion of an elasticity model (2.1) through a synthetic numerical example. The objective is to recover the subsurface elasticity parameters of the complex geological channelized field. Fig. 3.1 (a) shows the physical setup such as mesh and boundary conditions of the numerical example to be used for our demonstration. This setup mimics a compression test where the bottom boundary is supported by a horizontal roller to curtail the vertical motion and other boundaries are free to expand. Measurements of the displacements due to self weight (gravity) are assumed to be available at the top, left and right boundaries. For the sake of simplicity, both elasticity fields $\boldsymbol{\lambda}$ and $\boldsymbol{\mu}$ are assumed to be equal. Fig. 3.1 (b) depicts a realization $\boldsymbol{\lambda}_1$ of the elasticity parameters and Fig. 3.1 (c) shows the corresponding displacement in the y-direction (u_y) due to self weight. The cost function J, which is the term V_1 in Equation (2.49), is defined as

$$J = \frac{1}{2}||U_{mes} - U_{pred}||_2^2, \tag{3.1}$$

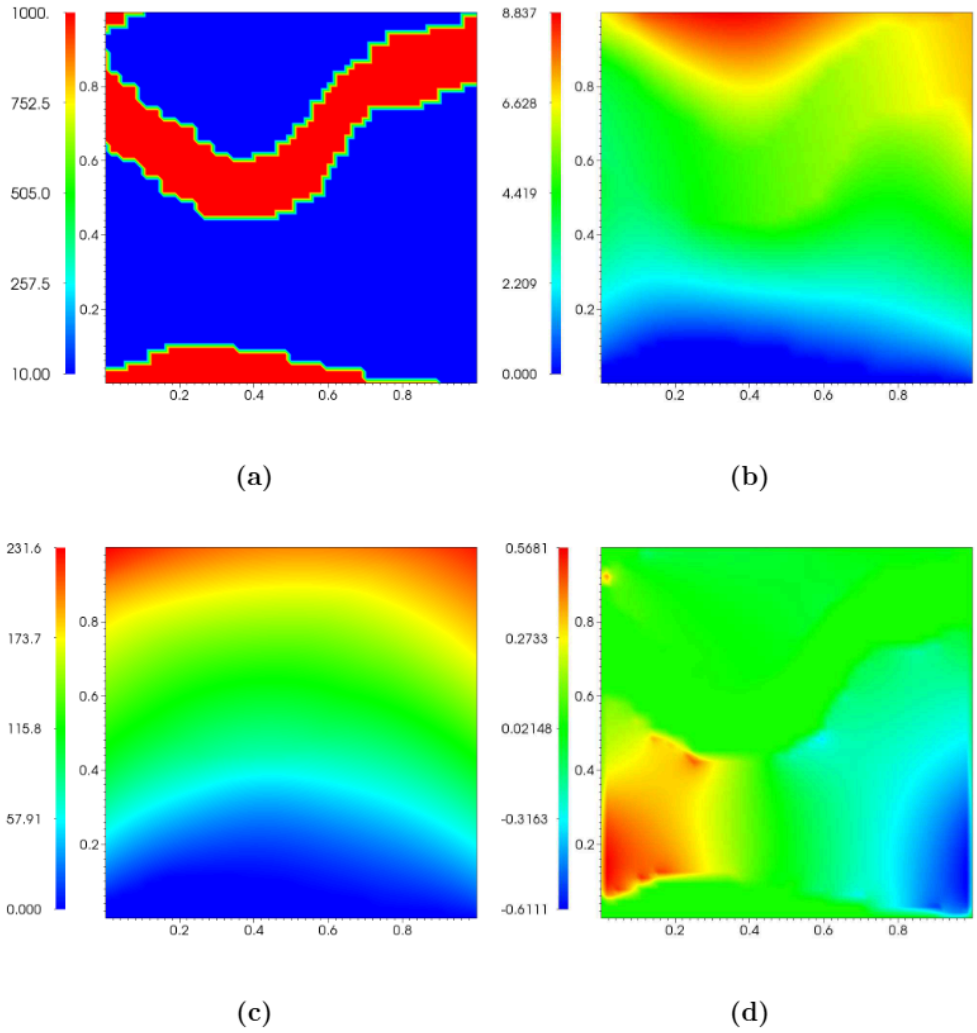
where U_{mes} and U_{pred} are measured and predicted displacement vectors, and $||.||_2$ is the L^2 norm. Fig. 3.2 (a) shows a different realization of the elasticity parameter $\boldsymbol{\lambda}_2$, which is used to compute the adjoint solution. Fig. 3.2 (b) shows the corresponding forward displacement in the y-direction (u_y) due to self weight. Fig. 3.2 (c) depicts adjoint displacement and Fig. 3.2 (d) shows the gradient of the cost function with respect to $\boldsymbol{\lambda}_2$ based on the measurements obtained with elasticity parameters $\boldsymbol{\lambda}_1$.



(a) **(b)**

(c)

Fig. 3.1. a) Physical setup of the numerical example used for the demonstration b) a realization $\boldsymbol{\lambda}_1$ of the elasticity parameters c) corresponding displacement in the y-direction (u_y) due to self weight.



(a) (b)

(c) (d)

Fig. 3.2. (a) A realization of the elasticity parameter $\boldsymbol{\lambda}_2$ (b) shows corresponding forward displacement in the y-direction (u_y) due to self weight (c) depicts adjoint displacement (d) gradient of the cost function with respect to $\boldsymbol{\lambda}_2$ based on the measurements obtained with elasticity parameters $\boldsymbol{\lambda}_1$.

3.1 Snapshot generation

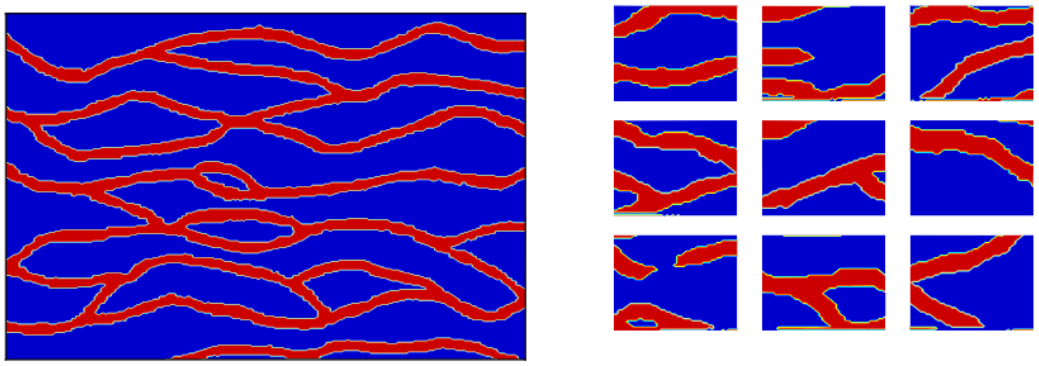


Fig. 3.3. A few snapshots generated (right) with the SNESIM algorithm and the training image used (left) for SNESIM.

In deep subsurface, elasticity parameters exhibit multiscale spatial fluctuations due to inherent geological heterogeneity [54]. In our numerical experiment, we rely on the single normal equation simulation (SNESIM) algorithm [28] based on training image shown in Fig. 3.3 (left) similar to Ma et al. and Sarma et al. [27, 26] and generate 1000 realizations of the dimension 45×45 channelized subsurface. Figure. 3.3 (right) depicts a few snapshots generated using the *snesnim* algorithm. Here, $\boldsymbol{\lambda}$ for the channelized zones and unchannelized zones are assumed to be 10 and 1000 MPa, respectively. In order to have the positive values for elasticity parameters, the inversion procedure is carried on $ln(\boldsymbol{\lambda})$ and $ln(\boldsymbol{\mu})$.

3.2 Efficiency of the kernel PCA and the pre-image

In contrast from the linear PCA, KPCA is preformed in a feature space instead of the original space. For the polynomial kernel $(\mathbf{x} \cdot \mathbf{y})^d$, an input space of realization in $\mathbb{R}^{N_R}$, will correspondingly have feature space of dimension N_F which is given by

$$N_F = \frac{(N_R + d - 1)!}{d!(N_R - 1)!} \approx (N_R)^d. \tag{3.2}$$

Compared to the original space, N_F is extremely large with higher order polynomial kernels. In our channelized model, we have $N_R = 10^3$ and $d = 5$, the $N_F \approx 10^{15}$, which allows kernel PCA to capture properties of the nonlinear data. The Kernel trick permits us to preform the eigendecomposition in low dimensional space instead of the high dimensional space.

Since our interest is inversion in the original space, a pre-imaging step is performed to transform the feature snapshot to the original snapshot. Unlike linear PCA, the solution to the pre-imaging is not unique and also suffers from the instability. In order to choose the best kernel for our procedure, we test Gaussian, linear, quadratic, cubic, 4th and 5th order polynomial kernels for their pre-imaging efficiency using a few selected snapshots. Fig. 3.5 depicts the results from this procedure for a particular snapshot. This figure shows that when the reduced dimension is 1000 (same as N_R), all the kernels are able to recover the original snapshot. In lower dimensions, increased polynomial kernel order (d) lead to efficient mapping. Also, determination of the pre-image became unstable for the polynomial kernels order greater than five.

Fig. 3.6 depicts eigenvalue decay of the snapshots for different kernels, which shows that the dimensionality reduced by linear PCA and KPCA is generally the same. Fig. 3.4 depicts a few snapshots generated using mean perturbation in KPCA space with Gaussian, linear, quadratic, cubic, fourth and fifth order kernels. This figure shows that as the order of the polynomial kernel increases, the mean perturbed data looks more like a channelized structure i.e., higher order kernels are able to represent data more efficiently. Based on Figs. 3.5, 3.6 and 3.4 we choose polynomial kernel with order 5 and dimension 20 (about 75% contribution) for our problem.

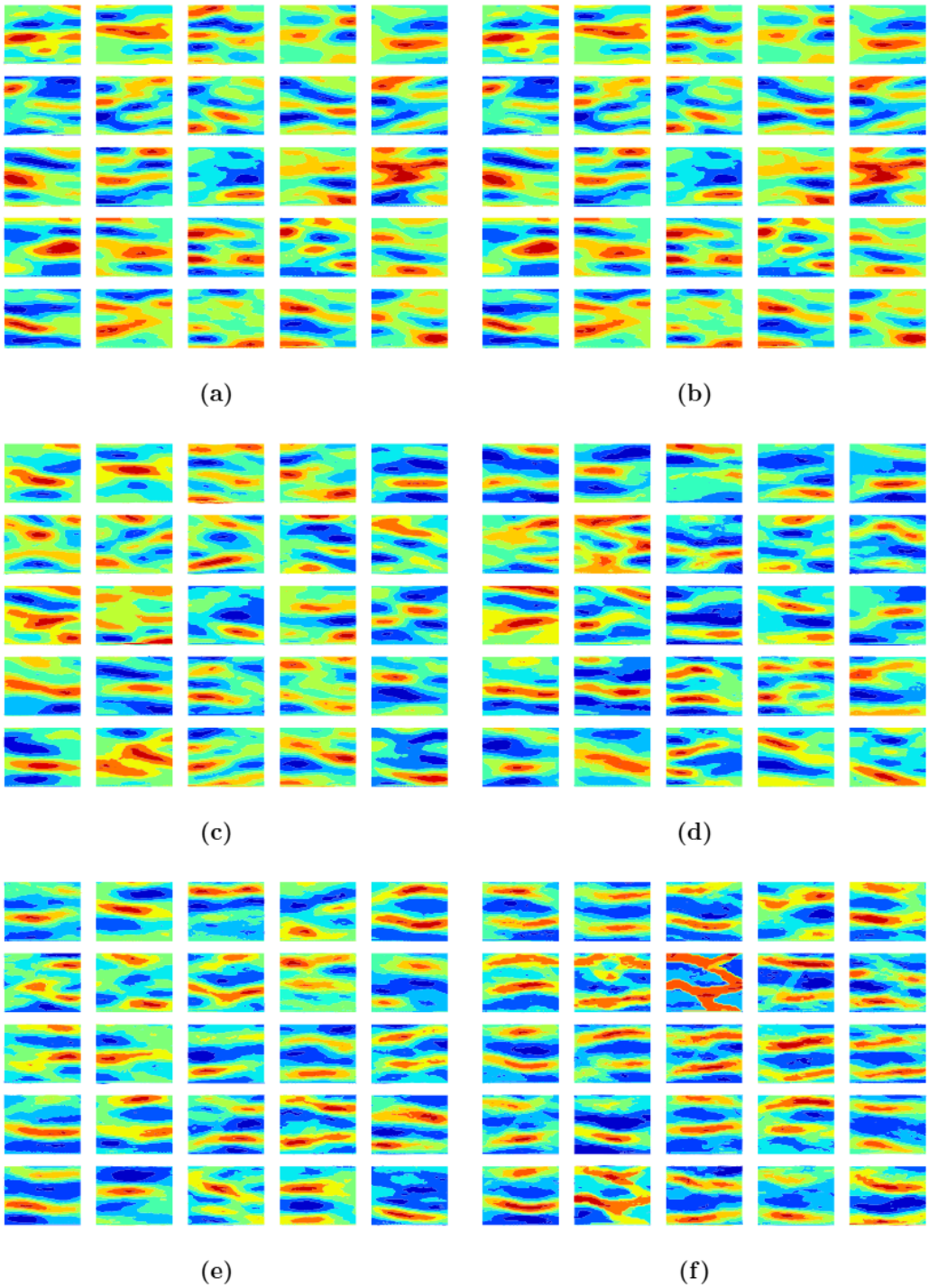


Fig. 3.4. A few snapshots generated using mean perturbation in KPCA space with a)Gaussian b)linear c)quadratic d)cubic e)fourth order and f)fifth order kernels.

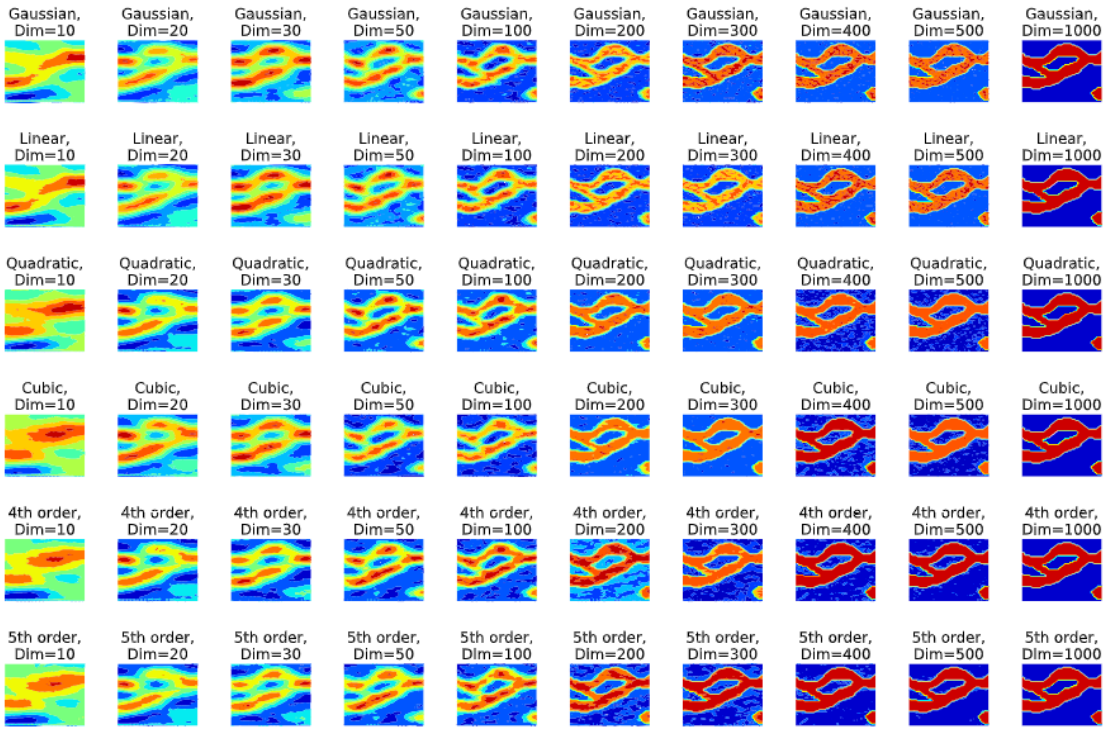


Fig. 3.5. KPCA with Gaussian, linear, quadratic, cubic, fourth and fifth order kernels in 10, 20, 30, 40, 50, 100, 200, 300, 400, 500, 1000 dimensions.

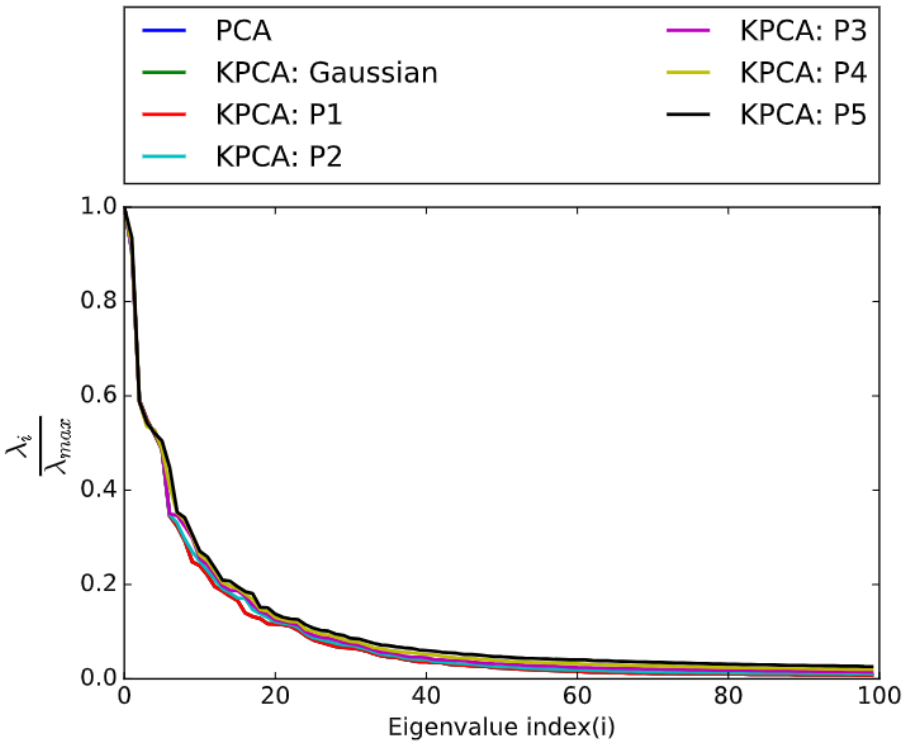


Fig. 3.6. Eigenvalue decay of the snapshots for different kernels.

3.3 Efficiency of the PCE

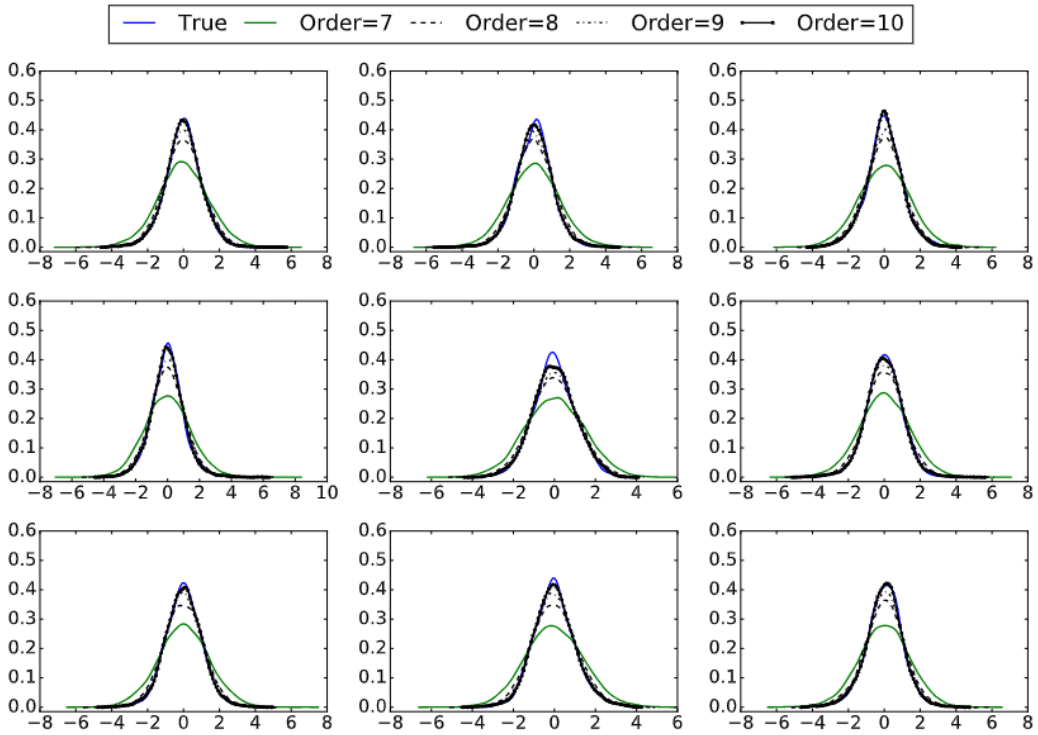


Fig. 3.7. Probability density function of a few ξ^d obtained using true samples and from the samples of PCE with different orders.

Nonlinear mapping of the parameter space $\Phi : \mathbb{R}^{N_R} \to \mathbb{R}^{N_F}$, $N_F \gg N_R$ and solving (2.28) leads to 1000 discrete realizations of the $\boldsymbol{\xi}^d$. In general, $\boldsymbol{\xi}^d$ are non-Gaussian, uncorrelated and dependent random variables. Assuming $\boldsymbol{\xi}^d$ are independent similar to [43, 44], we construct multiple PCEs for $\boldsymbol{\xi}^d$ using ICDF mapping. Fig. 3.7 depicts the probability density function of a few selected ξ^d constructed from the 1000 discrete realizations (true) and also samples obtained from the PCE with different orders. This figure shows that as the order of the PCE increases, PCE is able to capture true distribution of the ξ^d. Based on this plot PCE of order 10 is used to map ξ^d standard Gaussian variable η.

3.4 Numerical test for the gradient

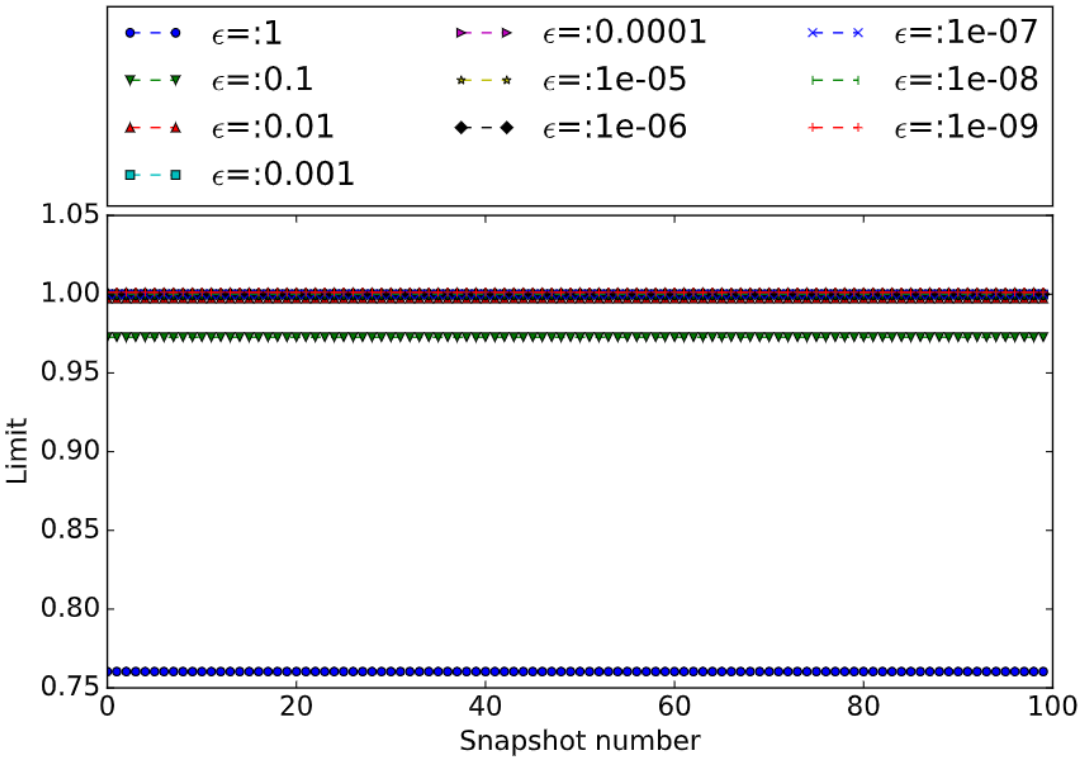


Fig. 3.8. Gradient limit for different choices of ϵ for 100 snapshots.

The gradients in the feature space are computed using adjoint PDE and TAPENADE [53], an automatic differentiation toolkit. In order to test the accuracy of computed gradients, we make use of the properties Gateaux differential $d_h f$. A Gateaux differential is defined as,

$$d_h J = \lim_{\epsilon \to 0} \frac{J(\eta + \epsilon h) - J(\eta)}{\epsilon}. \tag{3.3}$$

A property Gateaux derivative is if $h = \frac{\nabla_\eta J}{||\nabla_\eta J||^2}$, then $\epsilon \to 0$, $d_h J \to 1$. We use this property to test the accuracy of the derivative of the cost functional with respect to η. Fig. 3.8 depicts gradient limit for different choices of ϵ for 100 snapshots. This figure shows that for sufficiently small ϵ (> 0.1) the limit in Equation 3.3 goes to 1, thus verifies the accuracy of the gradient computation.

3.5 Stochastic inversion using MHMCMC and Langevin MCMC

The goal of our stochastic inversion framework is to recover the elastic parameters shown in Fig. 3.9 (a) based on the measurements of displacements at boundary grid points. Since truncated KPCA space consists of only 20 dimensions and measurements are only available at the boundaries, the goal here is to recover low dimension version Fig. 3.9 (b) of the original snapshot. Inversion is carried out using LMCMC and random walk MHMCMC algorithms. Adjoint PDE and automatic differentiation allowed us to compute gradient with only one extra run of the forward model.

Fig. 3.9 (c) and (d) show the posterior mean and standard deviation snapshots obtained using MHMCMC. Fig. 3.9 (e) and (f) show the posterior mean and standard deviation snapshots obtained using LMCMC. Fig. 3.10 shows the posterior distribution of the η for the random walk MHMCMC and LMCMC. In both cases, posteriors are concentrated near the original parameter showing that stochastic inversion leads to the right solution. Three MCMC chains with initial guess for η as -2, 0 and 2 are used to check the global convergence of the MCMC algorithms. Fig. 3.11 shows the convergence MCMC chains for random walk MHMCMC and LMCMC. This figure shows that chains started converging around 100th and 500th sample in case of LMCMC and MHMCMC respectively, i.e., gradient information allows for a faster convergence.

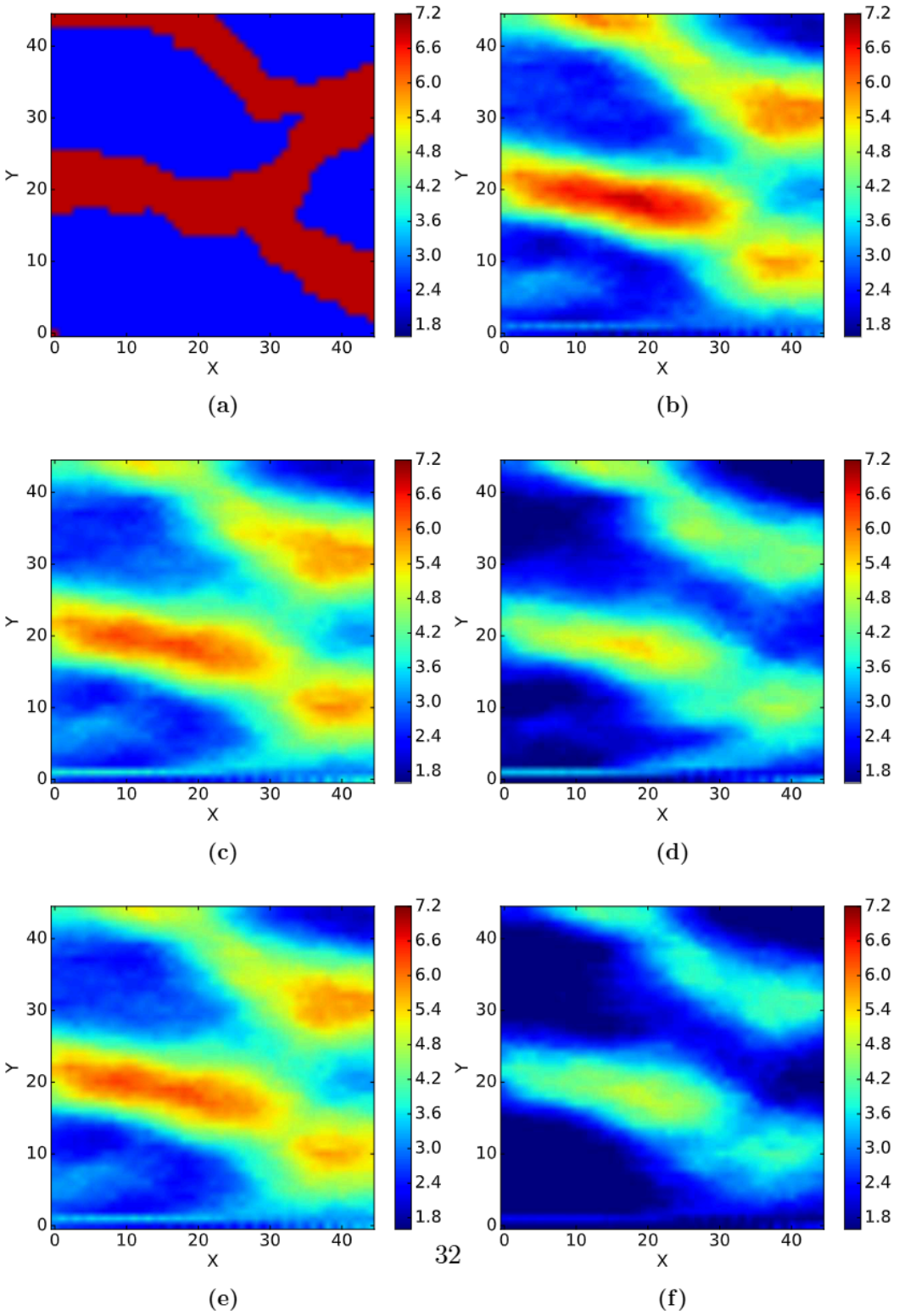
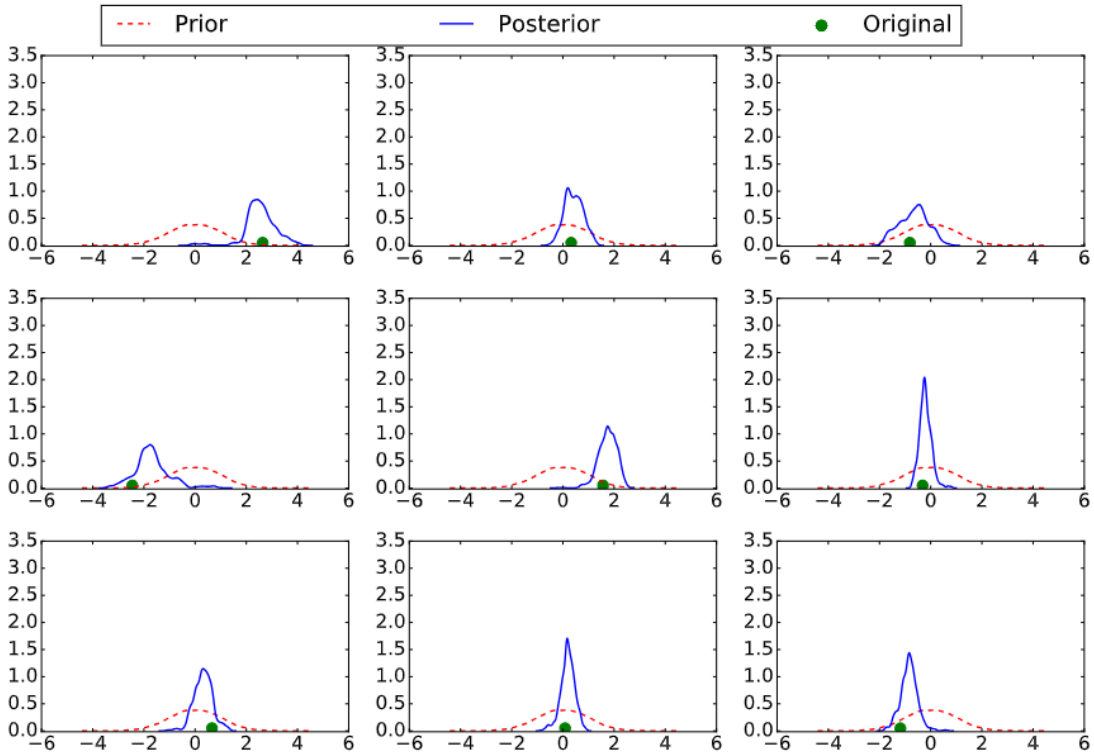
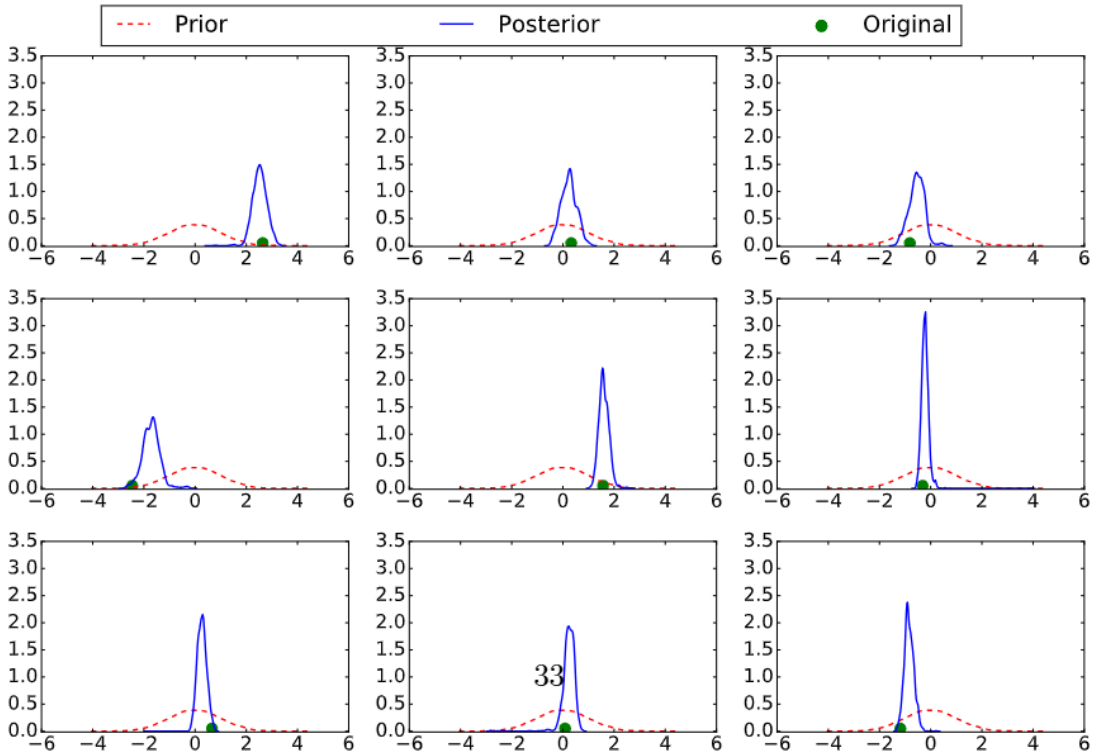


Fig. 3.9. a) Original b) KPCA projected c) MHMCMC posterior mean d) MHMCMC posterior standard deviation e) Langevin MCMC posterior mean and f) Langevin MCMC posterior standard deviation snapshots.

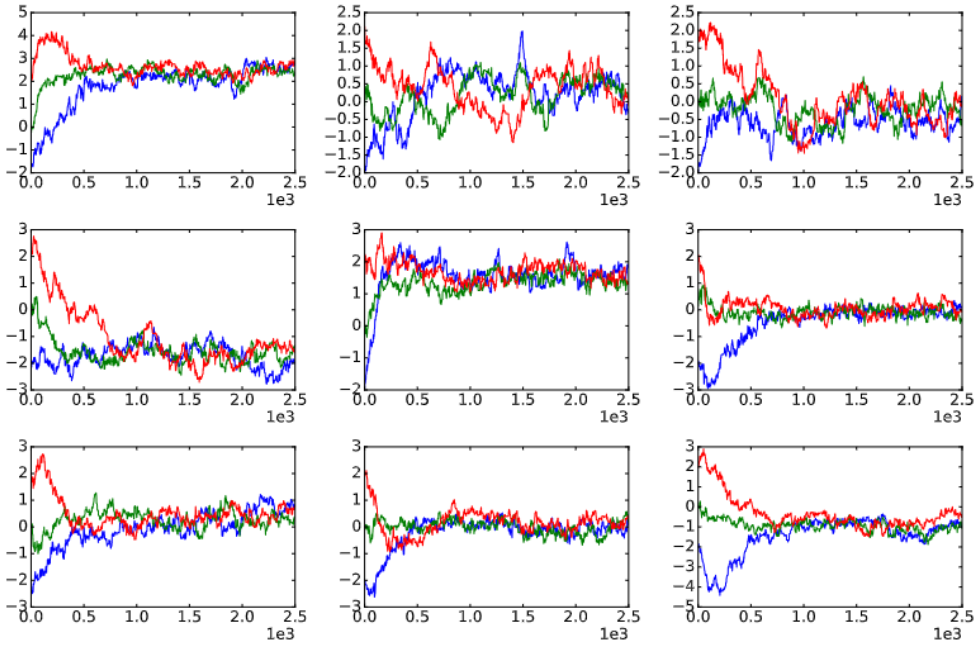


(a)

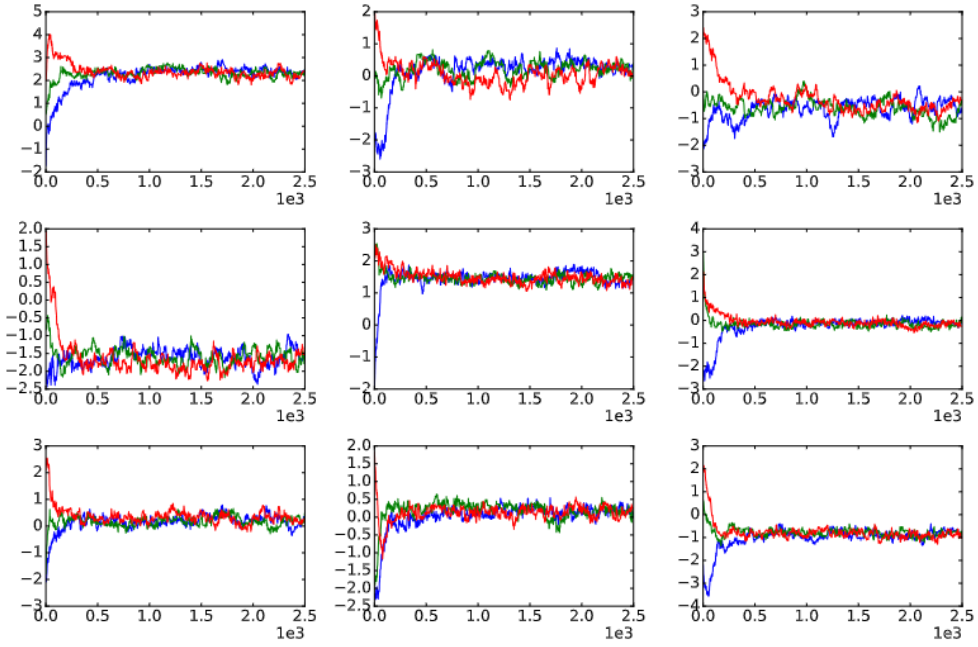


(b)

Fig. 3.10. Prior and posterior probability density functions for a few η's with original value for a) MHMCMC b) Langevin MCMC.



(a)



(b)

Fig. 3.11. Posterior MCMC chains for a few η's with staring at -2 (red), 0 (green) and 2(blue) for a)MCMC b)Langevin MCMC.

Chapter 4

Discussion

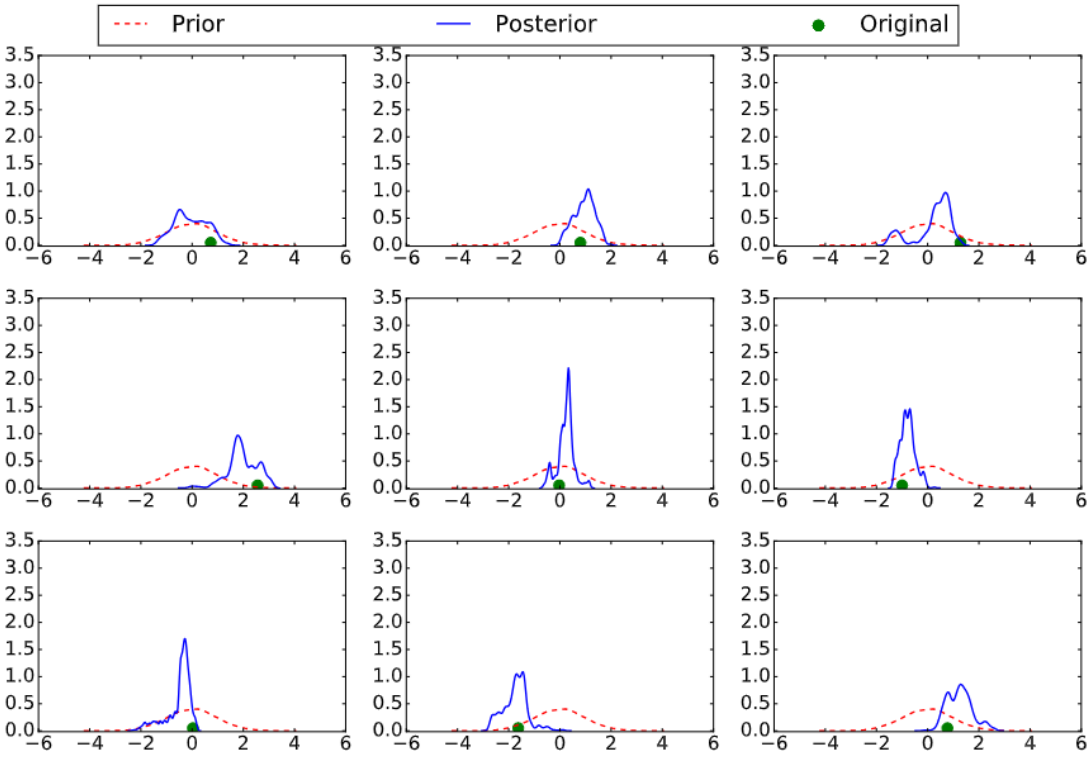


Fig. 4.1. Prior and posterior probability density functions for a few η's with original value for obtained using PCA-based Langevin MCMC.

As showed in figure 3.6, the dimensionality reduced by linear PCA and KPCA is

generally the same based on the existing data points but the reduced-order space they represent could be very different. Since the proposed method is based on LMCMC which has computational complexity of $O(n^{1/3})$ compared to MHMCMC complexity of $O(n)$, thus its computational cost scales better compared to MCMC. Also, for the numerical examples considered, the KPCA-based LMCMC and PCA-based LMCMC have 33.66% and 10.10% acceptance rate respectively. We will explain why KPCA is more useful than PCA in the background of KPCA-based stochastic inversion; The posterior probability density functions (PDFs) inverted by PCA-based MCMC (as seen in Fig. 4.1) are generally non-Gaussian and possibly multi-modal. By contrast, those inverted using KPCA-based MCMC are nearly Gaussian-distributed and are generally unimodal as a result of the embedded nonlinear mapping from the feature space to the parameter space. It is generally harder thus takes more iterations to converge the non-standard PDFs with many peaks, especially when a gradient-based MCMC is implemented, where the posterior is approximated by a local Gaussian during the inversion process; For deterministic parameter estimation or calibration, the performance between linear PCA-based optimizations and those utilizing KPCA are generally not significantly different. However, KPCA-based MCMC (stochastic inversion) achieves much better performance than PCA-based MCMC. The reason is that, compared to the linear PCA, the embedded manifold identified by the data-driven KPCA contains a more concentrated distribution of the underlying parameters that need to be inverted. Even though inverting any particular point (deterministic inversion) in the concentrated manifold may not be very distinguishable, inverting a distribution of points (stochastic inversion) in such a manifold will be critical for achieving high performance and accuracy. In specific, the neighborhood identified by linear PCA for any given channelized material parameter point may contain very few channelized structures, which can cause great difficulties for a high-dimensional random field inversions especially when considering stochastic inversions. Hence, the KPCA-based MCMC will demonstrate improved efficiency even without gradient information, thus making it useful for the applications where the adjoint model cannot be derived easily.

Chapter 5

Conclusions

We have presented an efficient stochastic inversion framework for the linear elasticity problem based on an adjoint model, automatic differentiation, and Kernel PCA. An improved, reduced representation of the complex elastic properties of the subsurface is captured using the low-dimensional feature space obtained from KPCA. Different kernels such as Gaussian, first, second, third, fourth and fifth polynomial kernels were tested and the kernel of KPCA is chosen based on snapshots obtained from the pre-imaging and mean perturbation. The efficiency of the proposed method is demonstrated through a synthetic numerical example with the objective of recovering the subsurface elastic parameters of the complex geological channelized field. Gradient-free MCMC and Langevin MCMC were able to sample from the true posterior after 500 and 100 forward model runs, respectively. The KPCA-based MCMC results in a higher acceptance rate compared to the PCA-based MCMC since the neighborhood identified by KPCA for any given channelized material parameter point contains more channelized structures.

Acknowledgment

This work was funded by the laboratory directed research and development (LDRD; 16-ERD-023) program and conducted under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under contract DE-AC52-7NA27344.

Bibliography

[1] A. Anandarajah, Computational methods in elasticity and plasticity: solids and porous media, Springer Science & Business Media, 2011.

[2] B. Kirtman, S. Power, A. Adedoyin, G. Boer, R. Bojariu, I. Camilloni, F. Doblas-Reyes, A. Fiore, M. Kimoto, G. Meehl, et al., Near-term climate change: projections and predictability.

[3] G. Dagan, S. P. Neuman, Subsurface flow and transport: a stochastic approach, Cambridge University Press, 2005.

[4] B. Kennett, Seismic wave propagation in stratified media, ANU Press, 2013.

[5] R. W. Graves, Simulating seismic wave propagation in 3d elastic media using staggered-grid finite differences, Bulletin of the Seismological Society of America 86 (4) (1996) 1091–1106.

[6] P. Kundur, N. J. Balu, M. G. Lauby, Power system stability and control, Vol. 7, McGraw-hill New York, 1994.

[7] A. Tarantola, Inverse problem theory and methods for model parameter estimation, SIAM, 2005.

[8] J. Martin, L. C. Wilcox, C. Burstedde, O. Ghattas, A stochastic newton mcmc method for large-scale statistical inverse problems with application to seismic inversion, SIAM Journal on Scientific Computing 34 (3) (2012) A1460–A1487.

[9] P. J. Green, A. Mira, Delayed rejection in reversible jump metropolis-hastings, Biometrika (2001) 1035–1053.

[10] A. Mira, Ordering and improving the performance of monte carlo markov chains, Statistical Science (2001) 340–350.

[11] H. Haario, E. Saksman, J. Tamminen, Adaptive proposal distribution for random walk metropolis algorithm, Computational Statistics 14 (3) (1999) 375–396.

[12] H. Haario, E. Saksman, J. Tamminen, An adaptive metropolis algorithm, Bernoulli (2001) 223–242.

[13] L. Tierney, A. Mira, Some adaptive monte carlo methods for bayesian inference, Statistics in medicine 18 (1718) (1999) 2507–2515.

[14] G. O. Roberts, J. S. Rosenthal, Optimal scaling of discrete approximations to langevin diffusions, Journal of the Royal Statistical Society: Series B (Statistical Methodology) 60 (1) (1998) 255–268.

[15] H. Haario, M. Laine, A. Mira, E. Saksman, Dram: efficient adaptive mcmc, Statistics and computing 16 (4) (2006) 339–354.

[16] M. Parno, Y. Marzouk, Transport map accelerated markov chain monte carlo, arXiv preprint arXiv:1412.5492.

[17] R. G. Ghanem, P. D. Spanos, Stochastic finite elements: a spectral approach (2003).

[18] Y. Marzouk, D. Xiu, A stochastic collocation approach to bayesian inference in inverse problems.

[19] Y. M. Marzouk, H. N. Najm, L. A. Rahn, Stochastic spectral methods for efficient bayesian solution of inverse problems, Journal of Computational Physics 224 (2) (2007) 560–586.

[20] N. Bliznyuk, D. Ruppert, C. A. Shoemaker, Local derivative-free approximation of computationally expensive posterior densities, Journal of Computational and Graphical Statistics 21 (2) (2012) 476–495.

[21] V. R. Joseph, Bayesian computation using design of experiments-based interpolation technique, Technometrics 54 (3) (2012) 209–225.

[22] C. E. Rasmussen, Gaussian processes for machine learning.

[23] K.-I. Funahashi, On the approximate realization of continuous mappings by neural networks, Neural networks 2 (3) (1989) 183–192.

[24] K. Hornik, M. Stinchcombe, H. White, Multilayer feedforward networks are universal approximators, Neural networks 2 (5) (1989) 359–366.

[25] B. Schölkopf, A. Smola, K.-R. Müller, Kernel principal component analysis, in: International Conference on Artificial Neural Networks, Springer, 1997, pp. 583–588.

[26] P. Sarma, L. J. Durlofsky, K. Aziz, Kernel principal component analysis for efficient, differentiable parameterization of multipoint geostatistics, Mathematical Geosciences 40 (1) (2008) 3–32.

[27] X. Ma, N. Zabaras, Kernel principal component analysis for stochastic input model generation, J. Comput. Phys. 230 (19) (2011) 7311–7331.

[28] S. Strebelle, Conditional simulation of complex geological structures using multiple-point statistics, Mathematical Geology 34 (1) (2002) 1–21.

[29] C. A. Thimmisetty, R. G. Ghanem, J. A. White, X. Chen, High-dimensional intrinsic interpolation using gaussian process regression and diffusion maps, Mathematical Geosciences.

[30] R. A. Adams, J. J. F. Fournier, Sobolev spaces, 2nd Edition, Vol. 140 of Pure and Applied Mathematics (Amsterdam), Elsevier/Academic Press, Amsterdam, 2003.

[31] R. Courant, D. Hilbert, Methods of mathematical physics, Vol. 1, CUP Archive, 1966.

[32] W. H. Press, Numerical recipes 3rd edition: The art of scientific computing, Cambridge university press, 2007.

[33] N. Cressie, The origins of kriging, Mathematical geology 22 (3) (1990) 239–252.

[34] E. H. Isaaks, et al., Applied geostatistics, Tech. rep., Oxford University Press (1989).

[35] G. Matheron, Principles of geostatistics, Economic geology 58 (8) (1963) 1246–1266.

[36] C. M. Bishop, Pattern recognition, Machine Learning 128 (2006) 1–58.

[37] B. Schölkopf, A. Smola, K.-R. Müller, Nonlinear component analysis as a kernel eigenvalue problem, Neural Comput. 10 (5) (1998) 1299–1319.

[38] B. Schölkopf, A. Smola, K.-R. Müller, Kernel principal component analysis, Springer Berlin Heidelberg, Berlin, Heidelberg, 1997, pp. 583–588.

[39] I. T. Jolliffe, Principal component analysis, 2nd Edition, Springer Series in Statistics, Springer-Verlag, New York, 2002.

[40] J. T. Kwok, I. W. Tsang, The pre-image problem in kernel methods, in: ICML, 2003, pp. 408–415.

[41] R. Lebrun, A. Dutfoy, A generalization of the nataf transformation to distributions with elliptical copula, Probabilistic Engineering Mechanics 24 (2) (2009) 172–178.

[42] M. Rosenblatt, Remarks on a multivariate transformation, The annals of mathematical statistics 23 (3) (1952) 470–472.

[43] R. G. Ghanem, A. Doostan, On the construction and analysis of stochastic models: characterization and propagation of the errors associated with limited data, Journal of Computational Physics 217 (1) (2006) 63–81.

[44] G. Stefanou, A. Nouy, A. Clement, Identification of random shapes from images through polynomial chaos expansion of random level set functions, International Journal for Numerical Methods in Engineering 79 (2) (2009) 127–155.

[45] N. Wiener, The homogeneous chaos, American Journal of Mathematics 60 (4) (1938) 897–936.

[46] M. Arnst, R. Ghanem, C. Soize, Identification of bayesian posteriors for coefficients of chaos expansions, Journal of Computational Physics 229 (9) (2010) 3134–3154.

[47] M. Eldred, J. Burkardt, Comparison of non-intrusive polynomial chaos and stochastic collocation methods for uncertainty quantification, AIAA paper 976 (2009) (2009) 1–20.

[48] G. Stefanou, A. Nouy, A. Clement, Identification of random shapes from images through polynomial chaos expansion of random level set functions, International Journal for Numerical Methods in Engineering 79 (2) (2009) 127–155.

[49] J. L. Jin Qin, Empirical likelihood and general estimating equations, The Annals of Statistics 22 (1) (1994) 300–325.

[50] M. Jones, The performance of kernel density functions in kernel distribution function estimation, Statistics & Probability Letters 9 (2) (1990) 129–132.

[51] R. Giering, T. Kaminski, Recipes for adjoint code construction, ACM Transactions on Mathematical Software (TOMS) 24 (4) (1998) 437–474.

[52] A. A. Oberai, N. H. Gokhale, G. R. Feijóo, Solution of inverse problems in elasticity imaging using the adjoint method, Inverse problems 19 (2) (2003) 297.

[53] L. Hascoët, V. Pascual, The Tapenade Automatic Differentiation tool: Principles, Model, and Specification, ACM Transactions On Mathematical Software 39 (3).

[54] C. Thimmisetty, A. Khodabakhshnejad, N. Jabbari, F. Aminzadeh, R. Ghanem, K. Rose, J. Bauer, C. Disenhof, Multiscale stochastic representation in high-dimensional data using gaussian processes with implicit diffusion metrics, in: Dynamic Data-Driven Environmental Systems Science, Springer International Publishing, 2015, pp. 157–166.