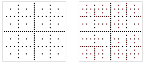
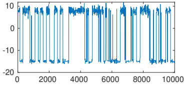
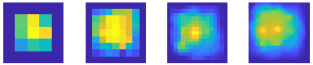
MS238 MS271 Theory and Application of Surrogate Models for Bayesian Inverse Problems

Minisymposium@**CSE19**, February 28th 2019, 9:45 AM-11:25 AM and 2:15 PM-3:55 PM, Location: Ballroom 100BC

1st session		2nd session	
09:45 AM	$\begin{bmatrix} * & \cdots & * \\ \vdots & \ddots & \vdots \\ * & \cdots & * \end{bmatrix} \to \begin{bmatrix} * & * \\ * & * \end{bmatrix}$	02:15 PM	$\mathcal{G}(\theta) \approx \sum_i \chi_{\Omega_i} \sum_k a_{i,k}\theta^k$
10:10 AM	(cancelled)	02:40 PM	GENEO
10:35 AM		03:05 PM	$\mathcal{G}(\theta) \sim \mathrm{N}(m^{(L)}(\theta), c^{(L)}(\theta,\theta))$
11:00 AM		03:30 PM	

Fast Sampling of parameterised Gaussian random fields

Jonas Latz

Input/Output: www.latz.io

Technical University of Munich
Department of Mathematics, Chair for Numerical Analysis
Email: jonas.latz@tum.de

February 28th 2019, SIAM Conference on Computational Science and Engineering, Spokane, WA, USA.

Joint work with

Marvin Eisenberger[1]   Daniel Kressner[2]   Stefano Massei[2]   Elisabeth Ullmann[3]

Acknowledgments

German research fund (DFG)   Technical University of Munich (TUM)

International Graduate School of Science and Engineering (IGSSE)

École Polytechnique Fédérale de Lausanne (EPFL)

Isaac Newton Institute, Cambridge, UK (INI)

[1] Computer Vision Group, Department of Informatics, TUM, Germany

[2] Chair for Numerical Algorithms and HPC, Mathematics Section, EPFL, Switzerland

[3] Chair for Numerical Analysis, Department of Mathematics, TUM, Germany

Outline

Motivation: Sampling from parameterised random fields

Reduced Basis for Eigenproblems and Reduced Basis Sampling

Numerical Experiments

Conclusions

Outline

Motivation: Sampling from parameterised random fields

- Why would someone do that?
- Why is that complicated?

Reduced Basis for Eigenproblems and Reduced Basis Sampling

Numerical Experiments

Conclusions

Motivation

Given an Inverse Problem

(IP) $$\text{Find } \theta \in X : \mathcal{G}(\theta) + \eta = y,$$

where

$\mathcal{G} : X \to Y$ is the forward response operator,

$\theta \in X$ is the unknown parameter,

$\eta \sim \mathrm{N}(0, \Gamma)$ is observational noise and

$y \in Y$ is observed data,

X, Y are Hilbert spaces

Motivation

Example (regression)

$\theta : D \to \mathbb{R}$ is a function

y are observations of this function

$\mathcal{G}$ maps any function θ to its values at the observed positions:

$$\theta \mapsto (\theta(x_i) : i = 1, ..., N_{\text{obs}}) \in Y.$$

Task: Identify the function θ given these point evaluations

Motivation

Example (groundwater flow)

$\theta : D \to \mathbb{R}$ models the log-conductivity of a groundwater reservoir

y are observations of the pressure in the groundwater reservoir

$\mathcal{G} = \mathcal{O} \circ G$ is the combination of

the solution operator $G(\theta) := p$ of the elliptic PDE:

$$\text{(PDE)} \qquad \begin{aligned} -\nabla \cdot \exp(\theta(x))\nabla p(x) &= f(x) \qquad && (x \in D) \\ p(x) &= 0 && (x \in \partial D) \end{aligned}$$

the observation operator $\mathcal{O}(p) := (p(x_i) : i = 1, ..., N_{\text{obs}}) \in Y$

Task: Identify the log-conductivity θ given the pressure data y. (PDE) links θ, y.

Motivation

We approach (PDE)-based (IP) Bayesian. So Assume,

(Prior) $$\theta \in \mathcal{L}^2(\Omega, \mathcal{A}, \mathbb{P}; X), \ \theta \sim \mu_0.$$

(BIP) $$\text{Find } \mu^y := \mathbb{P}(\theta \in \cdot | \mathcal{G}(\theta) + \eta = y),$$

given by

(Bayes' Rule) $$\frac{\mathrm{d}\mu^y}{\mathrm{d}\mu_0}(\theta) \propto L(y|\theta) := \exp\left(\underbrace{-\frac{1}{2} \| \Gamma^{-\frac{1}{2}} \left(\mathcal{G}(\theta) - y \right) \|_Y^2}_{=:\Phi(\theta)} \right).$$

Motivation

We approach (PDE)-based (IP) Bayesian. So Assume,

(Prior) $$\theta \in \mathcal{L}^2(\Omega, \mathcal{A}, \mathbb{P}; X), \; \theta \sim \mu_0.$$

(BIP) Find $$\mu^y := \mathbb{P}(\theta \in \cdot | \mathcal{G}(\theta) + \eta = y),$$

given by

(Bayes' Rule) $$\frac{\mathrm{d}\mu^y}{\mathrm{d}\mu_0}(\theta) \propto L(y|\theta) := \exp\left(\underbrace{-\frac{1}{2} \|\Gamma^{-\frac{1}{2}} \left(\mathcal{G}(\theta) - y\right) \|_Y^2}_{=:\Phi(\theta)} \right).$$

Motivation

We approach (PDE)-based (IP) Bayesian. So Assume,

(Prior) $\theta \in \mathcal{L}^2(\Omega, \mathcal{A}, \mathbb{P}; X), \ \theta \sim \mu_0.$

(BIP) Find $\mu^y := \mathbb{P}(\theta \in \cdot | \mathcal{G}(\theta) + \eta = y),$

given by

(Bayes' Rule) $$\frac{\mathrm{d}\mu^y}{\mathrm{d}\mu_0}(\theta) \propto L(y|\theta) := \exp\left(- \underbrace{\frac{1}{2} \| \Gamma^{-\frac{1}{2}} \left(\mathcal{G}(\theta) - y \right) \|_Y^2}_{=: \Phi(\theta)} \right).$$

Prior measure μ_0

θ is a function in $C^0(D)$,
e.g. model θ as a (mean-zero) Gaussian random field (RF), where

$$\mathrm{Cov}(\theta) := \mathcal{C}(\ell, \sigma)$$

is an exponential covariance operator

$$\varphi \mapsto \mathcal{C}(\ell, \sigma)\varphi := \int_D \sigma^2 \exp(-\|x - \cdot\|_2/\ell)\varphi(x)\mathrm{d}x.$$

Typical assumption: ℓ, σ are given a priori

What if they are unknown?

Prior measure μ_0

θ is a function in $C^0(D)$,
e.g. model θ as a (mean-zero) Gaussian random field (RF), where

$$\mathrm{Cov}(\theta) := \mathcal{C}(\ell, \sigma)$$

is an exponential covariance operator

$$\varphi \mapsto \mathcal{C}(\ell, \sigma)\varphi := \int_D \sigma^2 \exp(-\|x - \cdot\|_2/\ell)\varphi(x)\mathrm{d}x.$$

Typical assumption: ℓ, σ are given a priori

What if they are unknown?

What if...?

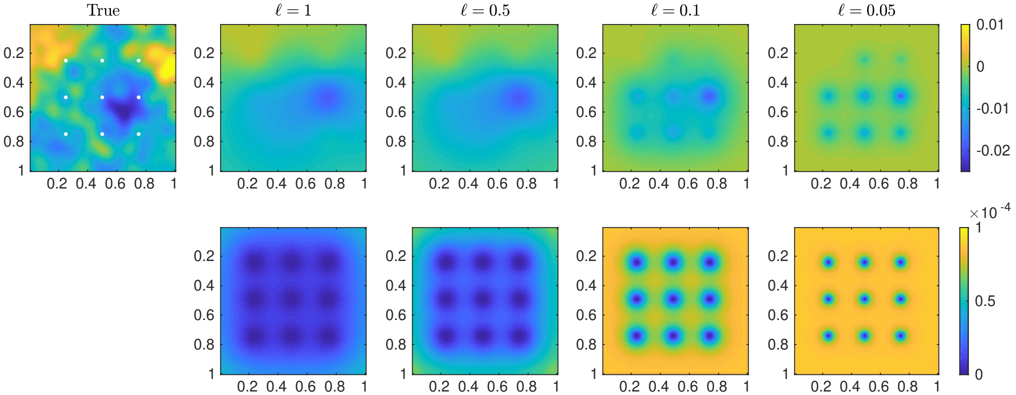


Figure: Estimating the **true value** with a Bayesian approach given noisy (noise level = 1%) evaluations of the random field at the 9 '●'. The three upper (lower) right plots show posterior mean (variances) estimates of the true values given a prior with exponential covariance with $\ell = 1, 0.5, 0.1, 0.05$. The true random field is a sample from an exponential covariance with $\ell = 0.5$.

What if...?

> “It ain’t what you don’t know that gets you into trouble.
> It’s what you know for sure that just ain’t so.
>
> – Mark Twain, 1835-1910

Use a hierarchical approach, i.e.

$$(\ell, \sigma) \sim \mu'$$

are also uncertain and shall be estimated

we will now only focus on $\ell \sim \mu'$ and assume that σ is known.

What if...?

> “It ain’t what you don’t know that gets you into trouble.
> It’s what you know for sure that just ain’t so.
>
> – Mark Twain, 1835-1910

Use a hierarchical approach, i.e.

$$(\ell, \sigma) \sim \mu'$$

are also uncertain and shall be estimated

we will now only focus on $\ell \sim \mu'$ and assume that σ is known.

What if...?

> “It ain’t what you don’t know that gets you into trouble.
> It’s what you know for sure that just ain’t so.

– Mark Twain, 1835-1910

Use a hierarchical approach, i.e.

$$(\ell, \sigma) \sim \mu'$$

are also uncertain and shall be estimated

we will now only focus on $\ell \sim \mu'$ and assume that σ is known.

Hierarchical Bayesian Inverse Problem

State of the art:

- Hierarchical Bayesian Estimation
  - Discussed in various standard textbooks[4]
- Hierarchical Bayesian Inverse Problem
  - Hierarchical level set inversion[5]
  - Theoretical considerations and applications in probabilistic numerics[6]
  - Polynomial chaos surrogates for hierarchical Bayesian inverse problems[7]

[4] Robert (2007): *The Bayesian Choice*, Springer.

[5] Dunlop, Iglesias, Stuart (2017): *Hierarchical Bayesian level set inversion*, Statistics and Computing: 27(6):1555-1584.

[6] Roinien, Girolami, Lasanen, Markkanen (2016): *Hyperpriors for Matérn fields with applications in Bayesian inversion*, ArXiv e-prints 1410.5522:1-27.

[7] Sraj, Le Maître, Knio, Hoteit (2016): *Coordinate Transformation and Polynomial Chaos for Bayesian Inference of a Gaussian Process with Parametrized Prior Covariance Function*, Comp. Meth. Appl. Mech. Engrg. 289:205–228.

Hierarchical Bayesian Inverse Problem: Setting

Given an Inverse Problem

(IP) $$\text{Find } \theta \in X : \mathcal{G}(\theta) + \eta = y,$$

where

$\mathcal{G} : X \to Y$ is the forward response operator,

$\theta \in X$ is the unknown parameter,

$\eta \sim \mathrm{N}(0, \Gamma)$ is observational noise,

$y \in Y$ is observed data and

X, Y are Hilbert spaces, $X := \mathcal{L}^2$, $Y := \mathbb{R}^n$.

Hierarchical Bayesian setting:

Given $\theta \sim K(\cdot|\ell) = \mathrm{N}(0, \mathcal{C}(\ell))$ (as a hyperprior) and

$\ell \sim \mu'$ (prior for ℓ).

Compute $\mu^y := \mathbb{P}((\ell, \theta) \in \cdot | \mathcal{G}(\theta) + \eta = y)$.

Hierarchical Bayesian Inverse Problem

Bayes' Rule for hierarchical Bayesian Inverse Problem

$$\mu^y(B) = \frac{1}{Z(y)} \iint_B \exp\left(-\Phi(\theta)\right) K(\mathrm{d}\theta|\ell)\mu'(\mathrm{d}\ell) \quad (B \in \mathcal{R} \otimes \mathcal{B}X)$$

$$Z(y) = \iint_{R \times X} \exp\left(-\Phi(\theta)\right) K(\mathrm{d}\theta|\ell)\mu'(\mathrm{d}\ell)$$

Generate posterior samples with Metropolis-within-Gibbs MCMC

Outline

Motivation: Sampling from parameterised random fields

- Why would someone do that?
- Why is that complicated?

Reduced Basis for Eigenproblems

Reduced Basis Sampling

Numerical Experiments

Conclusions

Random field sampling

MCMC sampling requires sampling from $\mathrm{N}(0, \mathcal{C}(\ell))$ for any $\ell \in R$

Expand $\theta \sim \mathrm{N}(0, \mathcal{C}(\ell))$ in a Karhunen-Loève (KL) expansion:

(KL)
$$\theta := \sum_{i=1}^{\infty} \sqrt{\lambda_i(\ell)}\psi_i(\ell)\xi_i,$$

where

$\xi_1, \xi_2, \ldots \sim \mathrm{N}(0,1)$ i.i.d.
$(\lambda_i(\ell), \psi_i(\ell))_{i=1}^{\infty}$ are eigenpairs of $\mathcal{C}(\ell)$,
where $\lambda_1(\ell) \geq \lambda_2(\ell) \geq \cdots$

Discretisation: Truncate after N_{sto} terms

(truncKL)
$$\theta_{KL} := \sum_{i=1}^{N_{\mathrm{sto}}} \sqrt{\lambda_i(\ell)}\psi_i(\ell)\xi_i,$$

Computational cost

Each MCMC sample requires evaluations of $\underbrace{\mathcal{G}}_{\text{PDE}}(\overbrace{\ \theta\ }^{\text{RF}})$.

PDE: N piecewise linear FEs (e.g. multigrid solver)

$$\frac{\text{Cost}}{\text{Sample}} : O(N; N \to \infty)$$

RF: N_{sto}-term KLE and N piecew. const. FEs (e.g. impl. rest. Lanczos method; `eigs`)

$$\frac{\text{Cost}}{\text{Sample}} : O(N^2 N_{\mathrm{sto}}; N \to \infty)$$

Toy example. Elliptic PDE in 2D, $N = 256^2$, $N_{\mathrm{sto}} = 400$ KL terms.

PDE solve $\sim 1\mathrm{s}$ (50000 MCMC Samples: $\sim 14\mathrm{h}$)

KL solve $\sim 50\mathrm{min}$ (50000 MCMC Samples: ~ 5 years)

Computational cost

Each MCMC sample requires evaluations of $\underbrace{\mathcal{G}}_{\text{PDE}}(\overbrace{\ \theta\ }^{\text{RF}})$.

PDE: N piecewise linear FEs (e.g. multigrid solver)

$$\frac{\text{Cost}}{\text{Sample}} : O(N; N \to \infty)$$

RF: N_{sto}-term KLE and N piecew. const. FEs (e.g. impl. rest. Lanczos method; `eigs`)

$$\frac{\text{Cost}}{\text{Sample}} : O(N^2 N_{\mathrm{sto}}; N \to \infty)$$

Toy example. Elliptic PDE in 2D, $N = 256^2$, $N_{\mathrm{sto}} = 400$ KL terms.

PDE solve $\sim 1\mathrm{s}$ (50000 MCMC Samples: $\sim 14\mathrm{h}$)
KL solve $\sim 50\mathrm{min}$ (50000 MCMC Samples: ~ 5 years)

Computational cost

Each MCMC sample requires evaluations of $\underbrace{\mathcal{G}}_{\text{PDE}}(\overbrace{\theta}^{\text{RF}})$.

PDE: N piecewise linear FEs (e.g. multigrid solver)

$$\frac{\text{Cost}}{\text{Sample}} : O(N; N \to \infty)$$

RF: N_{sto}-term KLE and N piecew. const. FEs (e.g. impl. rest. Lanczos method; `eigs`)

$$\frac{\text{Cost}}{\text{Sample}} : O(N^2 N_{\mathrm{sto}}; N \to \infty)$$

Toy example. Elliptic PDE in 2D, $N = 256^2$, $N_{\mathrm{sto}} = 400$ KL terms.

PDE solve $\sim 1\mathrm{s}$ (50000 MCMC Samples: $\sim 14\mathrm{h}$)

KL solve $\sim 50\mathrm{min}$ (50000 MCMC Samples: ~ 5 years)

Computational cost

Each MCMC sample requires evaluations of $\underbrace{\mathcal{G}}_{\text{PDE}}(\overbrace{\theta}^{\text{RF}})$.

PDE: N piecewise linear FEs (e.g. multigrid solver)

$$\frac{\text{Cost}}{\text{Sample}} : O(N; N \to \infty)$$

RF: N_{sto}-term KLE and N piecew. const. FEs (e.g. impl. rest. Lanczos method; `eigs`)

$$\frac{\text{Cost}}{\text{Sample}} : O(N^2 N_{\mathrm{sto}}; N \to \infty)$$

Toy example. Elliptic PDE in 2D, $N = 256^2$, $N_{\mathrm{sto}} = 400$ KL terms.

PDE solve $\sim 1\mathrm{s}$ (50000 MCMC Samples: $\sim 14\mathrm{h}$)

KL solve $\sim 50\mathrm{min}$ (50000 MCMC Samples: $\sim 5\ \mathrm{years}$)

Computational cost

Each MCMC sample requires evaluations of $\underbrace{\mathcal{G}}_{\text{PDE}}(\overbrace{\ \theta\ }^{\text{RF}})$.

PDE: N piecewise linear FEs (e.g. multigrid solver)

$$\frac{\text{Cost}}{\text{Sample}} : O(N; N \to \infty)$$

RF: N_{sto}-term KLE and N piecew. const. FEs (e.g. impl. rest. Lanczos method; `eigs`)

$$\frac{\text{Cost}}{\text{Sample}} : O(N^2 N_{\mathrm{sto}}; N \to \infty)$$

Toy example. Elliptic PDE in 2D, $N = 256^2$, $N_{\mathrm{sto}} = 400$ KL terms.

PDE solve $\sim 1\mathrm{s}$ (50000 MCMC Samples: $\sim 14\mathrm{h}$)
KL solve $\sim 50\mathrm{min}$ (50000 MCMC Samples: $\sim 5\ \mathrm{years}$)

Summary

Why would someone do that?

Fixing uncertain parameters can lead to wrong estimation results

Why is that complicated?

Recomputing the KL expansion is computationally very expensive

Outline

- Motivation: Sampling from parameterised random fields
- Reduced Basis for Eigenproblems and Reduced Basis Sampling[8]
  - Parameterised eigenproblems and reduced bases
  - Cost and memory requirement
- Numerical Experiments
- Conclusions

[8] L., Eisenberger, Ullmann (2019): *Fast Sampling of parameterised Gaussian random fields*, Comput. Methods in Appl. Mech. Engrg. (in Press, available online 15 Feb 2019).

Galerkin approximation of the eigenproblem

Problem. Compute eigenpairs of the covariance operators $\mathcal{C}(\ell)$, for various ℓ.

Definition. $(\lambda_i(\ell), \psi_i(\ell))$ is an eigenpair of $\mathcal{C}(\ell)$ in the weak formulation, if

$$(\ell\text{EVP}) \qquad \gamma(\ell; \psi_i(\ell), v) = \lambda_i(\ell)\langle \psi_i(\ell), v\rangle_X \quad \forall v \in X.$$

Here, $\gamma(\ell; \cdot, \cdot)$ is a bilinear form, given by

$$\gamma(\ell; u, v) := \langle \mathcal{C}(\ell)u, v\rangle_X$$

Discretisation: replace X by a finite-dimensional ansatz space X_h

Galerkin approximation

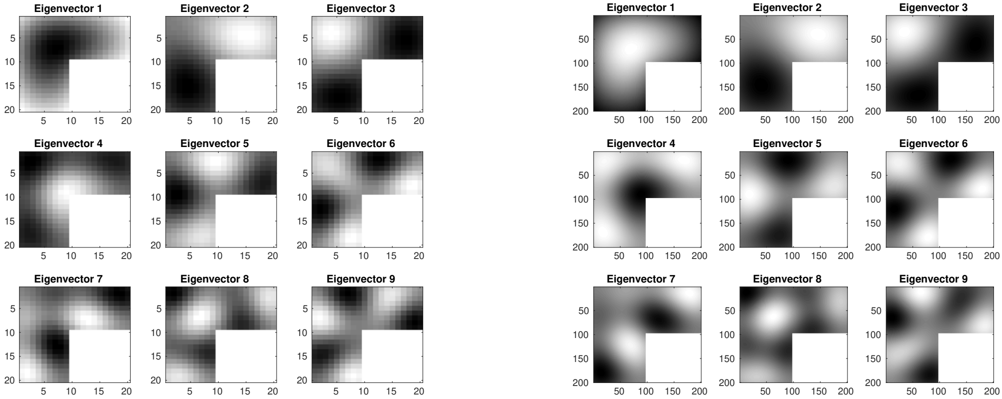


Figure: KL eigenfunctions on L-shaped domain, discretised with piecewise constant finite elements. Coarse grid (left) vs. fine grid (right).

Overview

At the moment: Solve the KL eigenproblem on a standard FE ansatz space.

high-dimensional and expensive

Reduced basis idea: Tailor a specific ansatz space for this eigenproblem.

changing the parameter may not influence the eigenproblem too severely

Reduced Basis

Offline phase:

Construct a reduced space $X_{RB} = \mathrm{span}(W)$.
W is orthonormal
Online-Offline-Decomposition with linearly separable operator

Online phase:

Construct reduced operator
Galerkin approximation on the reduced space
Project the reduced solution on X_h (resp. X)

Offline phase: Basis construction

1. Snapshot-based construction: Solve the full eigenproblem for some $\ell^{\text{snap}} \in R^{N_{\text{snap}}}$ and obtain

$$W^{\text{snap}} := (\psi_i(\ell_p^{\text{snap}}) : i = 1, ..., N_{\text{sto}}, p = 1, ..., N_{\text{snap}}),$$

   with eigenvectors $\psi_i(\ell)$ of $\mathcal{C}(\ell)$.

2. Consider all the snapshot eigenfunctions as basis functions and orthonormalise this basis with SVD (= proper orthogonal decomposition (POD)).

$$W^{\text{snap}} = U\Sigma V,$$

where

$$\Sigma = \text{diag}(\lambda_i^{\text{snap}} : i = 1, ..., N_{\text{sto}}N_{\text{snap}}), \quad U = (u_i : i = 1, ..., N_{\text{sto}}N_{\text{snap}})$$

Dimensionality reduction

Either of

- Set $W := (u_i : \lambda_i^{\mathrm{snap}} > 0)$.
  - Possibly still high dimensional
  - Uses all the information from the snapshots
- Set $W := (u_i : \lambda_i^{\mathrm{snap}} > \underline{\lambda})$ for some threshold $\underline{\lambda} > 0$.
  - This leads to a further dimensionality reduction, but to a larger reduced basis error.
  - Often: Full operator sparse, reduced operator dense
  - Here: All operators are dense.

Obtain a reduced basis W and
N_{RB}-dimensional reduced space $X_{\mathrm{RB}} := \mathrm{span}(W)$

Online-Offline Decomposition

The reduced operator is given by

$$\mathcal{C}^{RB}(\ell) = W^*\mathcal{C}(\ell)W$$

(Constr. costs online about $O(N^2)$)

Assume that we can decompose $\mathcal{C}(\ell)$ in the following way

$$\mathcal{C}(\ell) := \sum_{m=1}^{N_{\mathrm{lin}}} F_m(\ell)\mathcal{C}_m.$$

Then, we can construct the

$$\mathcal{C}^{RB}(\ell) := \sum_{m=1}^{N_{\mathrm{lin}}} F_m(\ell)\mathcal{C}_m^{RB} := \sum_{m=1}^{N_{\mathrm{lin}}} F_m(\ell) \quad \underbrace{W^*\mathcal{C}_m W}_{\text{Keep in the memory.}}$$

(Constr. cost online $O(N_{RB}^2 N_{\mathrm{lin}})$)

Online phase: Reduced Basis Sampling

Use the KL expansion to sample from the random field

Compute the KL eigenfunctions on a reduced basis

Algorithm:

1. Sample $\ell \sim \mu'$
2. Compute $(\Lambda(\ell), \Psi^{RB}(\ell)) \leftarrow \mathrm{eigs}(\mathcal{C}^{RB}(\ell), N_{\mathrm{sto}})$
3. $\theta_{RB} \leftarrow W\Psi^{RB}(\ell)\Lambda(\ell)^{1/2}\xi, \xi \sim \mathrm{N}(0, \mathrm{Id})$.

Outline

Motivation: Sampling from parameterised random fields

Reduced Basis for Eigenproblems and Reduced Basis Sampling[9]

- Parameterised eigenproblems and reduced bases
- Cost and memory requirement

Numerical Experiments

Conclusions

[9] L., Eisenberger, Ullmann (2019): *Fast Sampling of parameterised Gaussian random fields*, Comput. Methods in Appl. Mech. Engrg. (in Press, available online 15 Feb 2019).

Computational cost

Standard Sampling: $O(N^2 N_{\mathrm{smpl}} N_{\mathrm{sto}})$

Reduced Basis Sampling:

Offline:

$$O(\underbrace{N_{\mathrm{snap}} N^2}_{(1)} + \underbrace{N_{\mathrm{snap}} N^2 N_{\mathrm{sto}}}_{(2)} + \underbrace{N_{\mathrm{snap}} N^2 N_{\mathrm{sto}}}_{(3)}; N \to \infty)$$

(1) Construct full operator, (2) Solve full eigenproblem, (3) POD.

Online:

$$O(\underbrace{N_{\mathrm{smpl}} N_{\mathrm{RB}}^2 N_{\mathrm{lin}}}_{(4)} + \underbrace{N_{\mathrm{smpl}} N_{\mathrm{RB}}^2 N_{\mathrm{sto}}}_{(5)} + \underbrace{N N_{\mathrm{smpl}}}_{(6)} N_{\mathrm{RB}}; N \to \infty),$$

(4) Construct reduced operator, (5) Solve reduced eigenproblem, (6) Project the reduced solution onto the full space

In the online phase, the cost of one KL solve is equivalent to the cost of one PDE solve, linear in N.

Computational cost

Standard Sampling: $O(N^2 N_{\mathrm{smpl}} N_{\mathrm{sto}})$

Reduced Basis Sampling:

Offline:

$$O(\underbrace{N_{\mathrm{snap}} N^2}_{(1)} + \underbrace{N_{\mathrm{snap}} N^2 N_{\mathrm{sto}}}_{(2)} + \underbrace{N_{\mathrm{snap}} N^2 N_{\mathrm{sto}}}_{(3)}; N \to \infty)$$

(1) Construct full operator, (2) Solve full eigenproblem, (3) POD.

Online:

$$O(\underbrace{N_{\mathrm{smpl}} N_{\mathrm{RB}}^2 N_{\mathrm{lin}}}_{(4)} + \underbrace{N_{\mathrm{smpl}} N_{\mathrm{RB}}^2 N_{\mathrm{sto}}}_{(5)} + \underbrace{N N_{\mathrm{smpl}}}_{(6)} N_{\mathrm{RB}}; N \to \infty),$$

(4) Construct reduced operator, (5) Solve reduced eigenproblem, (6) Project the reduced solution onto the full space

In the online phase, the cost of one KL solve is equivalent to the cost of one PDE solve, linear in N.

Computational cost

Standard Sampling: $O(N^2 N_{\mathrm{smpl}} N_{\mathrm{sto}})$
Reduced Basis Sampling:

Offline:

$$O(\underbrace{N_{\mathrm{snap}} N^2}_{(1)} + \underbrace{N_{\mathrm{snap}} N^2 N_{\mathrm{sto}}}_{(2)} + \underbrace{N_{\mathrm{snap}} N^2 N_{\mathrm{sto}}}_{(3)}; N \to \infty)$$

(1) Construct full operator, (2) Solve full eigenproblem, (3) POD.

Online:

$$O(\underbrace{N_{\mathrm{smpl}} N_{\mathrm{RB}}^2 N_{\mathrm{lin}}}_{(4)} + \underbrace{N_{\mathrm{smpl}} N_{\mathrm{RB}}^2 N_{\mathrm{sto}}}_{(5)} + \underbrace{N N_{\mathrm{smpl}}}_{(6)} N_{\mathrm{RB}}; N \to \infty),$$

(4) Construct reduced operator, (5) Solve reduced eigenproblem, (6) Project the reduced solution onto the full space

In the online phase, the cost of one KL solve is equivalent to the cost of one PDE solve, linear in N.

Memory requirement and random field expansion

How do we keep the collected data?

- keep only (ℓ, ξ) - Memory efficient ($O(N_{\mathrm{sto}} + 1)$), but time consuming when the full random field is needed.
- keep the full random field θ - Memory inefficient $O(N)$, but fast.

Neither is actually useful.
Furthermore:

- When using MCMC (Gibbs move of ℓ), we need to construct $\mathcal{C}(\ell)$ in any step. (that is $O(N)$ in time and $O(N^2)$ in memory)

Reduced basis expansion

Consider the KL expansion

$$\begin{aligned}
\theta &= \sum_{i=1}^{N_{\mathrm{sto}}} \lambda_i(\ell)^{1/2}\psi_i(\ell)\xi_i \\
&\overset{\mathrm{RB}}{\approx} \sum_{i=1}^{N_{\mathrm{sto}}} \lambda_i(\ell)^{1/2}\xi_i W\psi_i^{RB}(\ell) \\
&= \sum_{i=1}^{N_{\mathrm{sto}}}\sum_{j=1}^{N_{RB}} \lambda_i(\ell)^{1/2}\psi_{i,j}^{RB}(\ell)w_j \\
&= \sum_{j=1}^{N_{RB}} \underbrace{\left(\sum_{i=1}^{N_{\mathrm{sto}}} \lambda_i(\ell)^{1/2}\psi_{i,j}^{RB}(\ell)\right)}_{\text{Multivar. Gaussian random variable, for fixed } \ell} w_j =: Wb(\ell)
\end{aligned}$$

Reduced Basis expansion

$b(\ell) \sim \mathrm{N}(0, \mathcal{C}^{RB}(\ell))$ contains the full covariance information of θ.

We can use $b(\ell) \in \mathbb{R}^{N_{RB}}$ to represent the full random field.

Replaces the full random field in the Gibbs Move of ℓ in MCMC sampling algorithm.

$\Rightarrow$ computational cost of the Gibbs Move of ℓ is independent of N.

Reduced Basis expansion

$b(\ell) \sim \mathrm{N}(0, \mathcal{C}^{RB}(\ell))$ contains the full covariance information of θ.

We can use $b(\ell) \in \mathbb{R}^{N_{RB}}$ to represent the full random field.

Replaces the full random field in the Gibbs Move of ℓ in MCMC sampling algorithm.

$\Rightarrow$ computational cost of the Gibbs Move of ℓ is independent of N.

Reduced Basis expansion

$b(\ell) \sim \mathrm{N}(0, \mathcal{C}^{RB}(\ell))$ contains the full covariance information of θ.

We can use $b(\ell) \in \mathbb{R}^{N_{RB}}$ to represent the full random field.

Replaces the full random field in the Gibbs Move of ℓ in MCMC sampling algorithm.

$\Rightarrow$ computational cost of the Gibbs Move of ℓ is independent of N.

Outline

Motivation: Sampling from parameterised random fields

Reduced Basis for Eigenproblems and Reduced Basis Sampling

Numerical Experiments

- Timings and Accuracy
- Bayesian Inverse Problem: Observations from a random field

Conclusions

Timings and accuracy

Spatial domain: $D = (0,1)^2$, resolved with piecewise constant finite elements

$\mathcal{C}(\ell)$ is the exponential covariance operator for $\ell = 0.1, 0.5, 1.4$

$N_{\mathrm{sto}} = 100$ KL eigenpairs

Reduced Bases are computed with a POD given 10 snapshots.

We measure:

The reduced basis error (in the eigenvalues) using $N_{RB} = 2^1, ..., 2^{13}$, given a fixed FE dimension $N = 100^2$.

The speed-up of the reduced basis with $N_{RB} = 256$, when increasing the FE dimension $N = 2^{2\cdot 4}, ..., 2^{2\cdot 7}$.

Accuracy of RB approximation

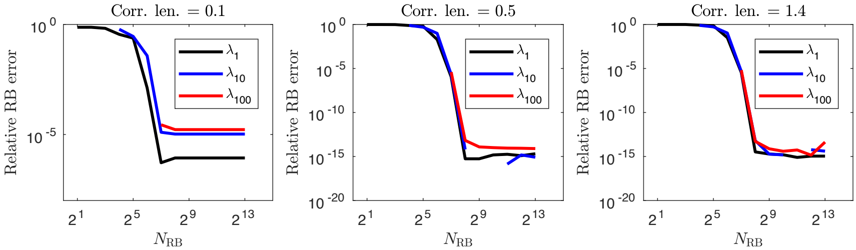


Figure: Relative Error of Eigenvalues, exponential covariance, $N = 100^2$, $N_{\mathrm{sto}} = 100$

If $N_{\mathrm{RB}} \geq 256$, the RB error is constantly small.

Timings

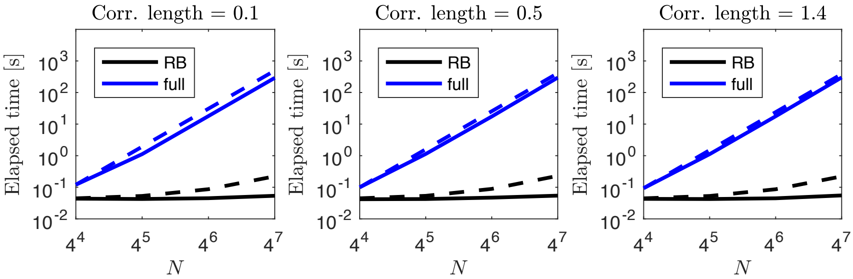


Figure: Timings of Eigenproblems, $N_{RB} = 256$, $N_{\mathrm{lin}} = 39$, $N_{\mathrm{sto}} = 100$, Code ran in `MATLAB` with Intel i7 (2.6 Ghz) CPU and 16 GB Ram. Each timing is averaged over 3 runs.

RB surrogate appears to be almost constant in N (actually, the RB eigenproblem dominates the cost)

Outline

Motivation: Sampling from parameterised random fields

Reduced Basis for Eigenproblems and Reduced Basis Sampling

Numerical Experiments

- Timings and Accuracy
- Bayesian Inverse Problem: Observations from a random field

Conclusions

Bayesian Inverse Problem: Random field estim.

We observe a Gaussian random field discretised on 256^2 with piecewise constant finite elements

- Spatial domain: $D = [0, 1]^2$
- Prior $K(\cdot|\ell, \sigma) = \mathrm{N}(0, \mathcal{C}(\ell, \sigma))$, where
  - $\mathcal{C}(\ell, \sigma)$ is the exponential covariance operator for $\ell \in [0.1, \sqrt{2}], \sigma \in [0.1, 1]$ (mean-zero)
    $\ell^{-1} \sim \mathrm{Unif}[2^{-1/2}, 10]$,
    $\sigma \sim \mathrm{N}(0.5, 0.1^2)(\cdot \cap [0.1, 1])$
- $N_{\mathrm{sto}} = 800$
- Observations generated with random fields with $\ell = 0.2, 1.1$, $\sigma^2 = 0.5$
- 2500 observations, assume noise level 6%

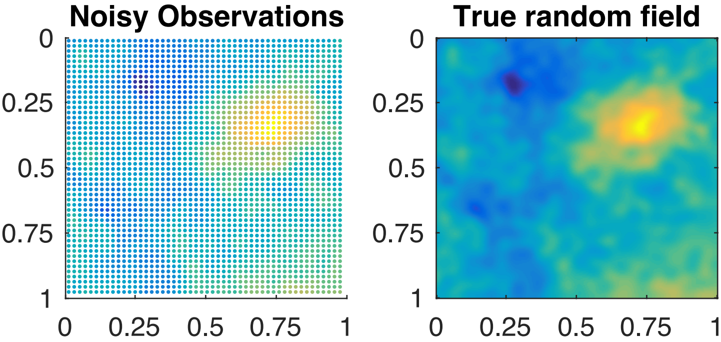


Figure: Correlation length: $\ell := 0.2$. (Left) Observations of the random fields. (Right) True random field

Reduced Basis Construction

POD with

$$\ell^{\text{snap}} = (0.1148, 0.1491, 0.2124, 0.3694, 1.4142)$$

(Chosen according to the prior of ℓ and to account for the degeneration of the exponential kernel)

Dimensionality reduction: Cut-off singular values $\leq 1\mathrm{E}-5$.

Resulted in $N_{RB} = 1153$.

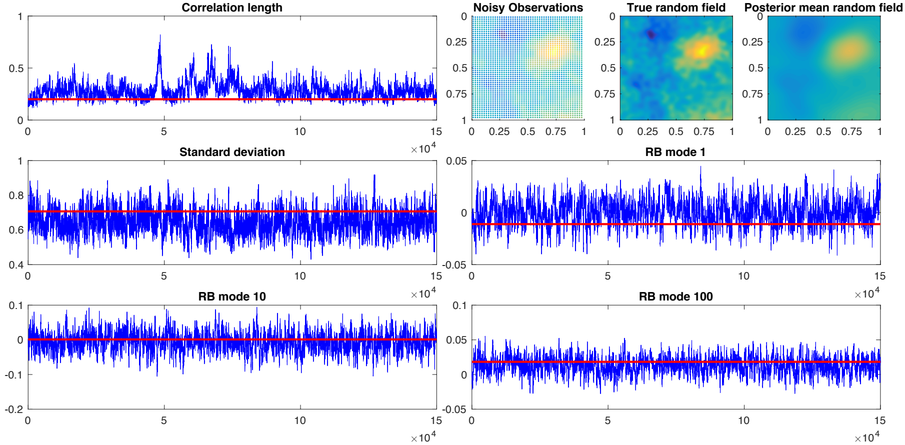


Figure: Estimation with RB-MCMC. True Correlation length: $\ell := 0.2$. Markov Chain with $1.5\mathrm{E}5$ Samples.

Discussion

Assumed to have a good initial value for the Markov Chain (hardly any burn-in)

Assess convergence of the MCMC with 24 shorter Markov Chains and 4 of the same length

Standard deviation is difficult to estimate (it heavily influences the MCMC proposal)

Random field estimates are fairly good

Correlation length estimates are heavily influenced by the high noise level

Outline

Motivation: Sampling from parameterised random fields

Reduced Basis for Eigenproblems

Reduced Basis Sampling

Numerical Experiments

Conclusions

Take-home messages

Using RBM in the online phase reduces the computational cost to $O(N)$

Can be used for efficient hierarchical Bayesian inverse problems

Significant reduction of memory requirement

Moreover:

Can also be used in a hierarchical forward problem

Approximate Offline-Online decomposition possible for any Matérn Covariance Operator

Current[10] and future research

Reducing cost in offline phase

Do we need to construct full covariance matrices?
Do we need to solve full eigenproblems?

RB error management with error indicator and greedy methods

How do we choose ℓ^{snap}?

Other linearisation techniques for covariance kernels

Yet, we use a Taylor approximation that is numerically unsuitable for small ℓ. Can we do better?

Real-world applications:

Deep Gaussian process regression

Hierarchical Bayesian hydraulic tomography in 3D

[10] Kressner, L., Massei, Ullmann: *Low-rank approximation of parameterised dense covariance matrices*, in preparation.

Jonas Latz

Input/Output: www.latz.io