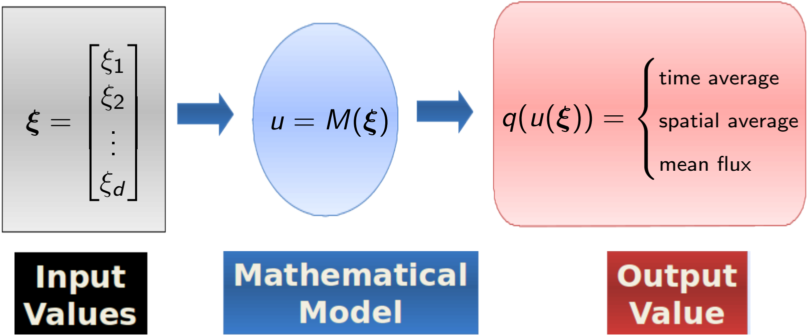
Compressive Sensing based Polynomial Chaos Expansions for Uncertainty Quantification

John D. Jakeman, M.S. Eldred, K. Sargsyan

Optimization and Uncertainty Quantification Department
Sandia National Laboratories
Albuquerque, NM USA

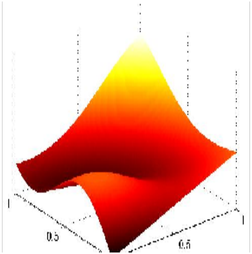
Sandia National Laboratories is a multi program laboratory managed and operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin Corporation, for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000.



$\xi = \begin{bmatrix} \xi_1 \\ \xi_2 \\ \vdots \\ \xi_d \end{bmatrix}$
$u = M(\xi)$
$q(u(\xi)) = \begin{cases} \text{time average} \\ \text{spatial average} \\ \text{mean flux} \end{cases}$
Input Values
Mathematical Model
Output Value

$$\xi_1 \in I_{\xi_1}$$
$$\xi_2 \in I_{\xi_2}$$
$$\vdots$$
$$\xi_d \in I_{\xi_d}$$

$$u = M(\xi)$$



Input Ranges

Mathematical Model

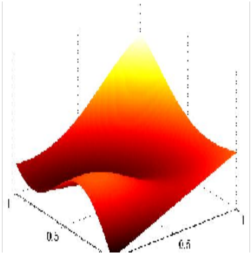
Output Response Surface

Every value of ξ will produce the same value in the QOI.

I

$(\Omega, \mathcal{F}, P)$
$\rho(\boldsymbol{\xi}) = \prod_{i=1}^{d} \rho_i(\xi_i)$

$u = M(\boldsymbol{\xi})$



Input Random Variables

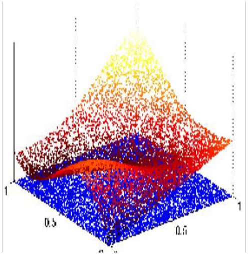
Mathematical Model

Output Values

Give distributions on the input data we can calculate statistical moments, distributions, etc. of the QOIs

We can sample from the input distributions to calculate statistics

$$\boldsymbol{\xi} = \begin{bmatrix} \xi_1 \\ \xi_2 \\ \vdots \\ \xi_d \end{bmatrix} \quad \Rightarrow \quad u = M(\boldsymbol{\xi}) \quad \Rightarrow$$



Input Random Variables | **Mathematical Model** | **Output Values**

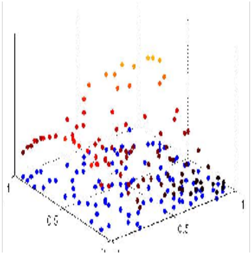
but...

Simulation models are computationally and financially expensive

$(\Omega, \mathcal{F}, P)$

$\rho(\boldsymbol{\xi}) = \prod_{i=1}^{d} \rho_i(\xi_i)$

$u = M(\boldsymbol{\xi})$



Input Random Variables

Mathematical Model

Output Values

Approximation issue: how do we use samples of $\mathbf{x}$ to learn f. (focus of this talk).

Design issue: how to choose the instances of $\mathbf{x}$ to maximize information gain whilst minimizing cost.

Ice-sheet modelling

Model seal-level rise due to ice-sheet loss in Greenland

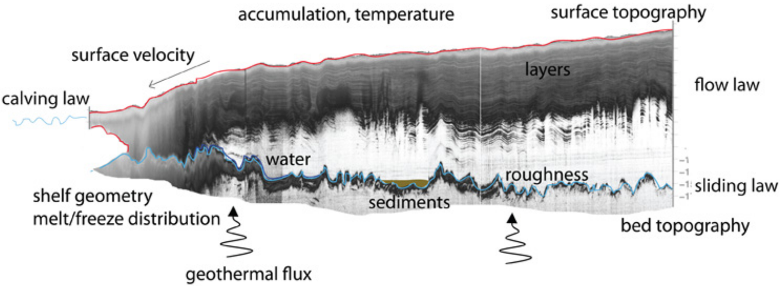


Figure 1: Schematic of observations, boundary conditions, and processes affecting ice sheet initialization.

Initial problem develop steady state condition and quantify uncertainty due to uncertainty in basal friction.

$$\begin{cases} -\nabla \cdot (2\mu\, \dot{\varepsilon}_1) = -\rho g \dfrac{\partial s}{\partial x} \\ \\ -\nabla \cdot (2\mu\, \dot{\varepsilon}_2) = -\rho g \dfrac{\partial s}{\partial y}, \end{cases} \qquad 2\mu\dot{\varepsilon}_1 = \beta u_1, \quad 2\mu\dot{\varepsilon}_2 = \beta u_2$$

where $\dot{\varepsilon}_1$ and $\dot{\varepsilon}_2$ consist of a strain rate tensor and μ a nonlinear ice viscosity and an effective strain rate.

Ice-sheet modelling

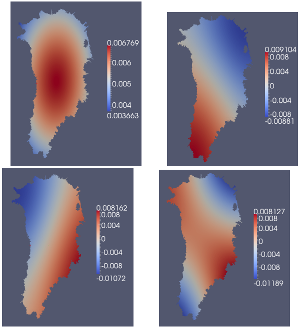
Model friction using Karhunen Loeve expansion

$$\log(\beta(\mathbf{x}, \boldsymbol{\xi})) = \bar{\beta}_0(\mathbf{x}) + \sum_{k=1}^{d} \sqrt{\lambda_k}\phi_k(\mathbf{x})\xi_k$$

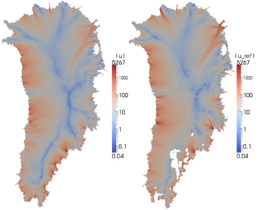
where $\{\lambda_k\}_{k=1}^{d}$ and $\{\phi_k(x)\}_{k=1}^{d}$ are, respectively, the eigenvalues and eigenfunctions of an assumed covariance kernel, e.g.

$$C_a(x_1, x_2) = \exp\left[-\frac{(x_1 - x_2)^2}{l_c^2}\right]$$

First 4 KLE bases



Mean field



Mike Eldred, Irina Kalashnikova, Mauro Perego, Andy Salinger, Laura Swiler

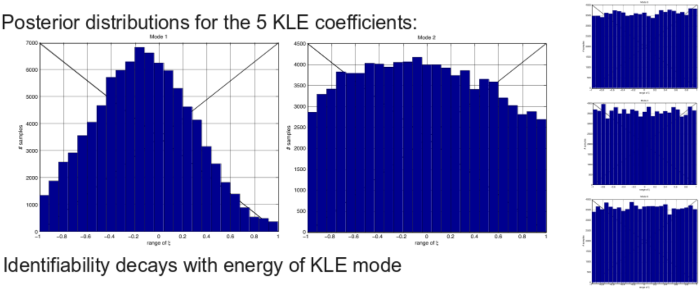
Infer distribution of friction using Bayesian inference

$$P(\boldsymbol{\xi}|\mathbf{d}) = \frac{P(\mathbf{d}|\boldsymbol{\xi})P_{\boldsymbol{\xi}}(\boldsymbol{\xi})}{\int P(\mathbf{d}|\boldsymbol{\xi})P_{\boldsymbol{\xi}}(\boldsymbol{\xi})\,d\xi}$$

where the likelihood the model is correct given the data is

$$P(\mathbf{d}|\boldsymbol{\xi}) = L(\boldsymbol{\xi}) = \prod_{i=1}^{N_d} P_\eta(d_i - M_i(\boldsymbol{\xi}))$$

M is the output of the model or a surrogate



Polynomial Chaos

Polynomial Chaos methods represent a function $f(\boldsymbol{\xi}) \in L_2(\rho(\boldsymbol{\xi}))$ as an expansion of orthogonal polynomials

$$f(\boldsymbol{\xi}) \approx \hat{f}(\boldsymbol{\xi}) = \sum_{\phi_{\boldsymbol{\lambda}} \in \mathcal{A}} \alpha_{\boldsymbol{\lambda}} \phi_{\boldsymbol{\lambda}}(\boldsymbol{\xi}), \quad \boldsymbol{\lambda} = (\lambda_1, \ldots, \lambda_d)$$

where $\{\phi_{\boldsymbol{\lambda}}(\boldsymbol{\xi})\}$ are tensor product of orthonormal polynomials which are chosen to be orthogonal to a distribution $\rho(\boldsymbol{\xi})$ of the random vector $\boldsymbol{\xi}$. That is

$$\left(\phi_{\boldsymbol{\lambda}_i}(\boldsymbol{\xi}), \phi_{\boldsymbol{\lambda}_j}(\boldsymbol{\xi})\right) = \int_{I_{\boldsymbol{\xi}}} \phi_{\boldsymbol{\lambda}_i}(\boldsymbol{\xi}) \phi_{\boldsymbol{\lambda}_j}(\boldsymbol{\xi}) \rho(\boldsymbol{\xi}) = \delta_{ij}$$

Distribution	Density function	Polynomial	Weight function	Support range
Normal	$\frac{1}{\sqrt{2\pi}} e^{\frac{-x^2}{2}}$	Hermite $He_n(x)$	$e^{\frac{-x^2}{2}}$	$[-\infty, \infty]$
Uniform	$\frac{1}{2}$	Legendre $P_n(x)$	1	$[-1, 1]$
Beta	$\frac{(1-x)^{\alpha}(1+x)^{\beta}}{2^{\alpha+\beta+1}B(\alpha+1,\beta+1)}$	Jacobi $P_n^{(\alpha,\beta)}(x)$	$(1-x)^{\alpha}(1+x)^{\beta}$	$[-1, 1]$
Exponential	e^{-x}	Laguerre $L_n(x)$	e^{-x}	$[0, \infty]$
Gamma	$\frac{x^{\alpha}e^{-x}}{\Gamma(\alpha+1)}$	Generalized Laguerre $L_n^{(\alpha)}(x)$	$x^{\alpha}e^{-x}$	$[0, \infty]$

Polynomial Chaos: statistics

A number of important statistics can be computed directly from the PCE

$$\mathbb{E}[f(\hat{\boldsymbol{\xi}})] = \alpha_{\{0,\dots,0\}}, \quad \mathrm{Var}[f(\hat{\boldsymbol{\xi}})] = \sum_{\phi_{\boldsymbol{\lambda}} \in \mathcal{A}} \alpha_{\boldsymbol{\lambda}}^2 \langle \phi_{\boldsymbol{\lambda}}^2 \rangle - \alpha_{\{0,\dots,0\}}^2$$

Due to the orthogonality of the PCE basis we can also easily calculate the popular Sobol indices

$$\sigma_{\mathbf{u}} = \frac{1}{\mathrm{Var}[f(\hat{\boldsymbol{\xi}})]} \sum_{\mathbf{j} \in \mathcal{J}} c_{\mathbf{j}}^2 \langle \Phi_{\mathbf{j}}^2 \rangle, \quad \mathcal{J} = \{\mathbf{j} : \mathrm{nonzeros}(\mathbf{j}) = \mathbf{u}\}$$

where $\mathbf{u} \subseteq \{1, \dots, d\}$ is the set of active dimensions. The main effect indices are the sum of all indices involving only the ith variable and the total effect indices are just the sum of all the indices involving the ith, i.e.

$$S_{\mathrm{M}_i} = \sum_{\{\mathbf{u}:\mathbf{u}=\{i\}\}} \sigma_{\mathbf{u}}, \quad S_{\mathrm{T}_i} = \sum_{\{\mathbf{u}:i\in\mathbf{u}\}} \sigma_{\mathbf{u}}$$

Estimating the PCE coefficients

- Stochastic Galerkin method
  - Galerkin projection is used to solve the variational form of a set of differential equations producing a set of coupled equations that must be solved. Requires solver modification
- Pseudo-spectral methods
  - The PCE coefficients are determined by the following Fourier-type integrals

$$\alpha_{\boldsymbol{\lambda}_i} = (f(\boldsymbol{\xi}), \phi_{\boldsymbol{\lambda}_i}(\boldsymbol{\xi})) = \int_{I_{\boldsymbol{\xi}}} f(\boldsymbol{\xi})\, \phi_{\boldsymbol{\lambda}_i}(\boldsymbol{\xi})\, d\rho(\boldsymbol{\xi}), \quad i = 1, \ldots, P$$

    These integrals can be evaluated via sparse grid quadrature. Need to limit aliasing errors
- Sparse grid interpolation
  - Construct sparse grid interpolant using Lagrange basis functions and perform linear transformation into the desired PCE basis. Preferred

Compressive sensing

Given a small set of M of (possibly un-structured) realizations with corresponding model outputs

$$\boldsymbol{\Xi} = \{\boldsymbol{\xi}_1, \ldots, \boldsymbol{\xi}_M\}, \quad \mathbf{f} = (f(\boldsymbol{\xi}_1), \ldots, f(\boldsymbol{\xi}_M))^T$$

we would like to find a solution to

$$\begin{bmatrix} f(\boldsymbol{\xi}^{(1)}) \\ f(\boldsymbol{\xi}^{(2)}) \\ \vdots \\ f(\boldsymbol{\xi}^{(M)}) \end{bmatrix} = \begin{bmatrix} \phi_{\boldsymbol{\lambda}_1}(\boldsymbol{\xi}^{(1)}) & \phi_{\boldsymbol{\lambda}_2}(\boldsymbol{\xi}^{(1)}) & \ldots & \phi_{\boldsymbol{\lambda}_P}(\boldsymbol{\xi}^{(1)}) \\ \phi_{\boldsymbol{\lambda}_1}(\boldsymbol{\xi}^{(2)}) & \phi_{\boldsymbol{\lambda}_2}(\boldsymbol{\xi}^{(2)}) & \ldots & \phi_{\boldsymbol{\lambda}_P}(\boldsymbol{\xi}^{(2)}) \\ \vdots & \vdots & & \vdots \\ \phi_{\boldsymbol{\lambda}_1}(\boldsymbol{\xi}^{(M)}) & \phi_{\boldsymbol{\lambda}_2}(\boldsymbol{\xi}^{(M)}) & \ldots & \phi\boldsymbol{\lambda}_P(\boldsymbol{\xi}^{(M)}) \end{bmatrix} \begin{bmatrix} \boldsymbol{\alpha}_{\boldsymbol{\lambda}_1} \\ \boldsymbol{\alpha}_{\boldsymbol{\lambda}_2} \\ \vdots \\ \vdots \\ \vdots \\ \boldsymbol{\alpha}_{\boldsymbol{\lambda}_P} \end{bmatrix} + \begin{bmatrix} e_1 \\ e_2 \\ \vdots \\ e_M \end{bmatrix}$$

Must choose truncation. Typically

$$\mathcal{A} = \mathcal{A}^d_{p,q} = \{\phi_{\boldsymbol{\lambda}} : \|\boldsymbol{\lambda}\|_q \leq p\}$$

with $q = 1$. The number of terms in this total degree basis

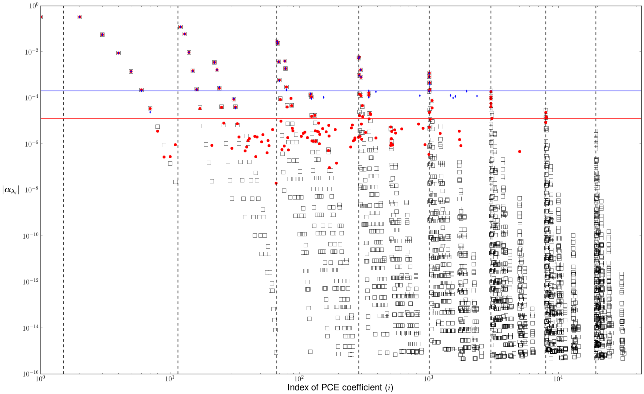
$$\text{card } \mathcal{A}^d_{p,1} \equiv P = \binom{d+p}{d}$$

Basis grows exponentially with dimension. So often $M \ll P$

Sparsity

A polynomial chaos expansion (signal) is defined to be s-sparse when $\|\boldsymbol{\alpha}\|_0 \leq s$. In practice, not many PCE will be truly sparse, but rather the magnitude of the PCE coefficients will decay rapidly

$$s = \#\{\boldsymbol{\lambda} : |\alpha_{\boldsymbol{\lambda}}| > \tau\}$$



When PCE is compressible it is still well approximated by a sparse signal.

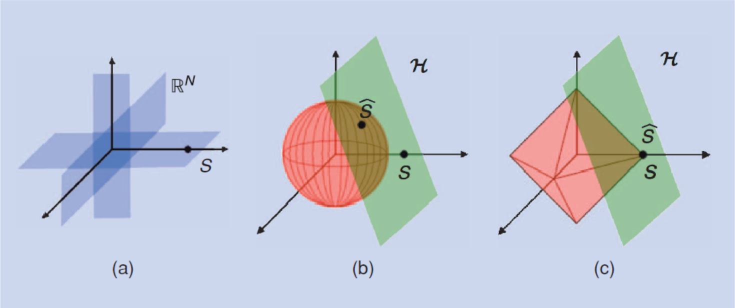
Compressive sensing

Compressive sensing attempts to find the dominant PCE coefficients by solving the following optimization problem

$$\mathbf{c} = \arg\min_{\mathbf{c}} \ \|\mathbf{c}\|_1 \quad \text{such that} \quad \|\boldsymbol{\Phi}\mathbf{c} - \mathbf{f}\|_2 \leq \varepsilon$$

$$\|\boldsymbol{\alpha}\|_1 = \sum_{i=1}^{P} |\alpha_i|$$

Geometric interpretation



Baraniuk (2007)

Mutual Coherence

The mutual coherence effects the performance of compressive sensing

$$\mu(\mathbf{\Phi}) = \max_{1<j<k\leq P} \frac{\left|\tilde{\phi}_j^T \tilde{\phi}_k\right|}{\left\|\tilde{\phi}_j\right\|_2 \left\|\tilde{\phi}_k\right\|_2}$$

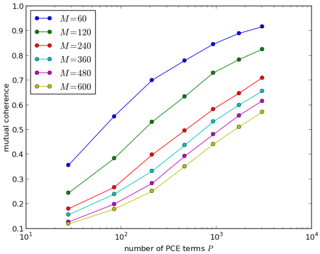
Compressive sensing will obtain a better estimate of the PCE coefficients if the mutual coherence of $\mathbf{\Phi}$ is small.

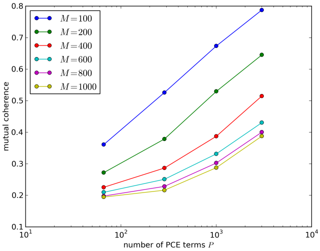
Intuitively, if two columns are closely correlated the mutual coherence will be large and it will be impossible in general to distinguish whether the energy in the signal comes from one or the other.

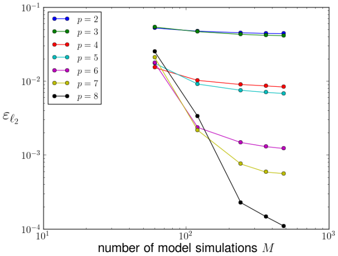
Blindly choosing a large degree p can cause a degradation in the accuracy of the PCE coefficients.

Mutual Coherence

Mutual coherence increases with degree p







Growth of PCE basis

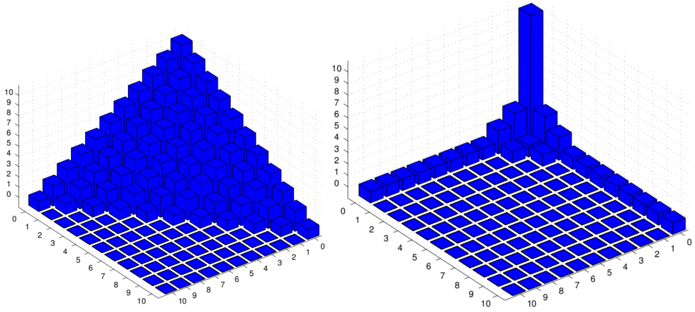


Figure: (Left) A total degree index set $\mathcal{A}^3_{10,1}$. (Right) A hyperbolic index set $\mathcal{A}^3_{10,2/5}$.

Total degree truncation grows to quickly with dimension limiting its ability to recover high-degree terms.

Hyperbolic sets will have difficulty recovering basis terms with many active variables.

Compressive sensing solvers

There are many compressive sensing algorithms used to find sparse solutions

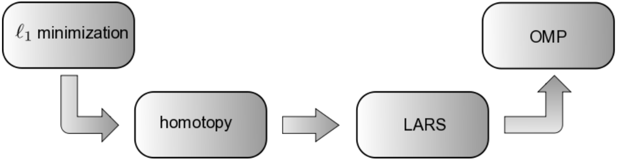


Figure: Bridging provably convergent ℓ_1 minimization algorithms and greedy algorithms such as OMP. Donoho 2008

- Quadratic cone optimization provably solves ℓ_1 minimization
- Homotopy provably solves ℓ_1 minimization problems.
- OMP and LARS are greedy approximations OMP solves a least-squares problem at each iteration, whereas LARS solves a linearly penalized least-squares problem.
- LARS is easily modified to solve the homotopy problem.

Orthogonal matching pursuit (OMP)

Algorithm 1 α=Orthogonal Matching Pursuit (OMP)[$\mathbf{\Phi}$,$\mathbf{f}$,ε]

1: Set $\mathbf{r}_0 = \mathbf{f}$, $\Lambda_0 = \emptyset$, $i = 0$
2: **while** $\|\mathbf{r}_i\|_2 > \varepsilon$ or $i < M$ **do**
3: $\quad i = i + 1$
4: $\quad \boldsymbol{\lambda}_i = \arg\max_{\boldsymbol{\lambda}_k \in \Lambda} \frac{\left|\mathbf{r}_{i-1}^T \tilde{\phi}_{\boldsymbol{\lambda}_k}\right|}{\left\|\tilde{\phi}_{\lambda_k}\right\|_2}, \quad \Lambda_i = \Lambda_{i-1} \cup \boldsymbol{\lambda}_i$
5: $\quad \boldsymbol{\alpha}_i = P_i \mathbf{f}, \quad \mathbf{r}_i = (I - P_i)\mathbf{f}$
6: **end while**

Cross validation

Let $\kappa : \{1, \ldots, N\} \to \{1, \ldots, K\}$ be an indexing function that determines the partition of the training data. Furthermore let $\hat{f}^{-\kappa}$ the approximation built on the data with the κ part removed, then the cross validation error is given by

$$CV(\gamma) = \frac{1}{N} \sum_{k=1}^{K} e_{\kappa(k)}, \quad e_{\kappa(k)} = \sum_{j \in \kappa(k)} (y_j - \hat{f}^{-\kappa(k)}(x_j))^2 \qquad (1)$$

where γ is a set of hyper-parameters, for example $\gamma = (t, \varepsilon)$, which can be estimated using cross validation.

Cross validation: estimating hyper-parameters

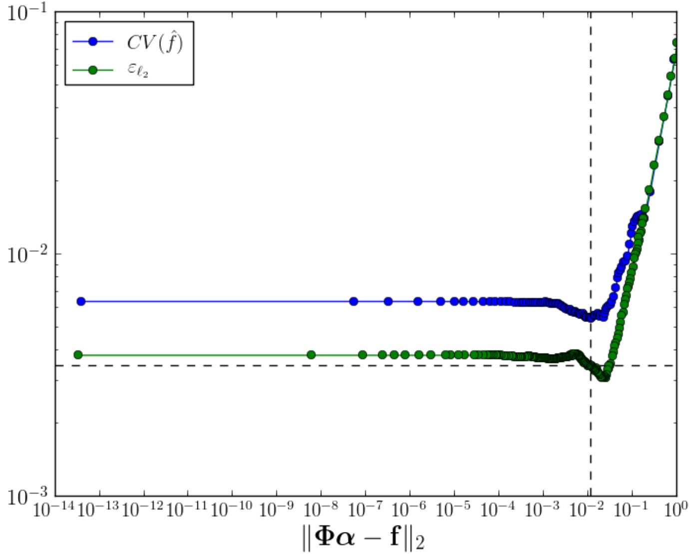


Figure: The use of cross validation to select the truncation tolerance ε

Cross validation: estimating hyper-parameters

Cross validation can be used to choose the degree p. (non-adapted method)

In high dimensions we can only consider 2nd or 3rd degree basis

p	3	4	5
$\lvert\mathcal{A}^{40}_{3,1}\rvert$	$12,341$	$135,751$	$1,221,759$

What if the function is anisotropic and some of the largest coefficients correspond to higher degree terms $p > 3$?

Lets try and determine the basis iteratively

Basis selection

Algorithm 2 $\Lambda^\star$,$\boldsymbol{\alpha}^\star$=BASIS_SELECTION[$\boldsymbol{\Phi}$,$\mathbf{f}$,$\varepsilon$]

Set $\Lambda^\star = \Lambda_0 = A^d_{p,1} = \arg\min_{A^d_{p,1} \in \{A^d_{1,1}, A^d_{2,1}, \ldots\}} \left| |A^d_{p,1}| - 10M \right|$
Set $t^\star = 3$, $CV^\star = \infty$, $i = 1$
while TRUE **do**
  $\boldsymbol{\alpha}_i$, CV_i = CS[$\boldsymbol{\Phi}(\Lambda_{i-1})$,$\mathbf{f}$]
  $\Lambda_i^\varepsilon = \{\boldsymbol{\lambda} : \boldsymbol{\lambda} \in \Lambda_{i-1}, \boldsymbol{\alpha}_{\boldsymbol{\lambda}} \neq 0\}$
  for $t \in \{1, \ldots, t^\star\}$ **do**
    $\Lambda_{i,t}$ = EXPAND[$\Lambda_i^\varepsilon(\boldsymbol{\alpha}_i)$,$t$]
    $\boldsymbol{\alpha}_{i,t}$, $CV_{i,t}$ = CS[$\boldsymbol{\Phi}(\Lambda_{i,t})$,$\mathbf{f}$]
    if $CV_{i,t} < CV_i$ **then**
      $CV_i = CV_{i,t}$, $\boldsymbol{\alpha}_i = \boldsymbol{\alpha}_{i,t}$, $\Lambda_i = \Lambda_{i,t}$, $t^\star = t$
    end if
  end for
  if $CV_i^\star > CV^\star$ **then**
    TERMINATE
  end if
  $CV^\star = CV_i$, $\Lambda^\star = \{\boldsymbol{\lambda} : \boldsymbol{\lambda} \in \Lambda_i, \boldsymbol{\alpha}_{\boldsymbol{\lambda}} \neq 0\}$, $\boldsymbol{\alpha}^\star = \{\boldsymbol{\alpha}_{\boldsymbol{\lambda}} : \boldsymbol{\lambda} \in \Lambda_i, \boldsymbol{\alpha}_{\boldsymbol{\lambda}} \neq 0\}$
  $i = i + 1$
end while

Index set expansion

Choose indices satisfying the following admissibilty criterion

$$\boldsymbol{\lambda} - \mathbf{e}_k \in \Lambda \text{ for } 1 \leq k \leq d,\ \lambda_k > 1$$

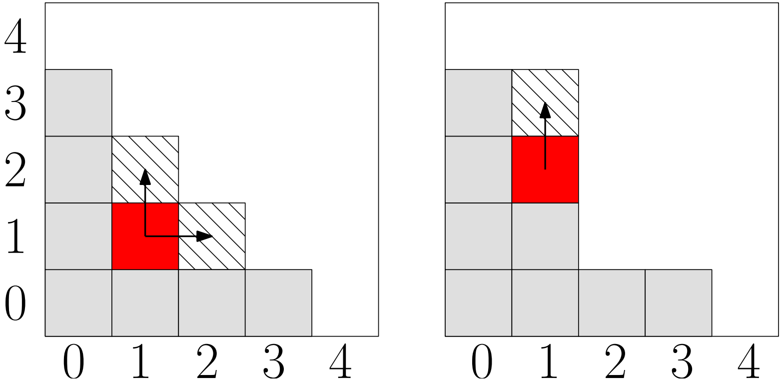


Figure: Identifying an admissible index. The non-zero indices identified by CS, Λ_i^{ε}, are in gray, admissible indices are stripped, the index being is red.

Numerical study

- To numerically demonstrate the properties of basis selection we will build PCE using maxi-min Latin hypercube samples $\boldsymbol{\Xi} = \{\boldsymbol{\xi}_i\}_{i=1}^M$ of size M
- For the small sample sizes M the selection of $\boldsymbol{\Xi}$ significantly effects the performance of any approximation method
- We will use the RMSE to measure the accuracy of the PCEs constructed. Specifically given a set of $Q = 100,000$ LHD samples $\boldsymbol{\Xi}_{\text{test}} = \{\boldsymbol{\xi}^{(i)}\}_{i=1}^Q \in [0,1]^d$ and samples of the true function $f(\boldsymbol{\xi}^{(i)})$ and the PCE approximation $\hat{f}(\boldsymbol{\xi}^{(i)})$ we compute

$$\varepsilon_{\ell^2} = \left(\frac{1}{Q} \sum_{i=1}^{Q} |\hat{f}(\boldsymbol{\xi}^{(i)}) - f(\boldsymbol{\xi}^{(i)})|^2 \right)^{1/2}$$

Algebraic Function

Consider the algebraic test function

$$f_{\mathrm{CP}}(\mathbf{x}) = \left(1 + \sum_{k=1}^{d} c_k\, \xi_k\right)^{-(d+1)}, \quad \boldsymbol{\xi} \in [0,1]^d$$

$$c_k^{(1)} = \frac{k - \frac{1}{2}}{d}, \quad c_k^{(2)} = \frac{1}{k^2} \quad \text{and} \quad c_k^{(3)} = \exp\left(\frac{k \log(10^{-8})}{d}\right)$$

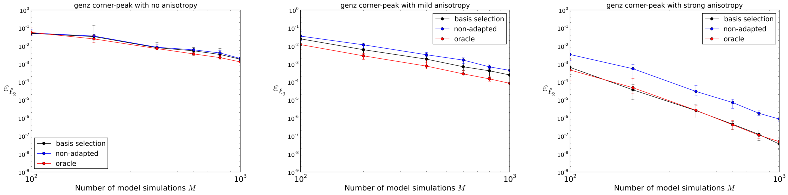
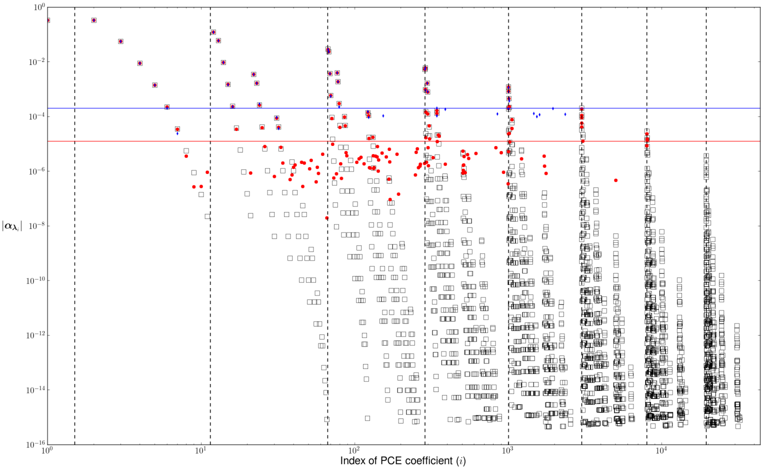


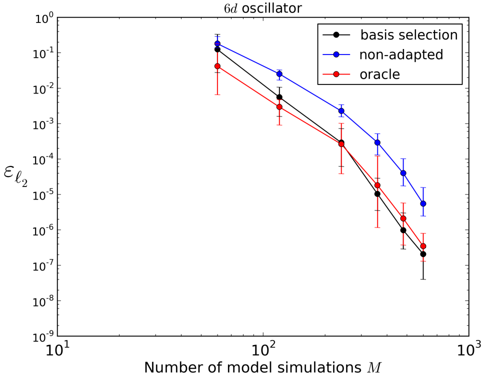
Figure: Errors in the PCE for increasing design sizes M. top-left: $c^{(1)}$, top-right: $c^{(2)}$, bottom: $c^{(3)}$

Algebraic Function



Damped Harmonic Oscillator

$$\frac{d^2x}{dt^2}(t,\boldsymbol{\xi}) + \gamma\frac{dx}{dt} + kx = f\cos(\omega t),$$
$$x(0) = x_0, \quad \dot{x}(0) = x_1,$$



$$\boldsymbol{\xi} = (\gamma, k, f, \omega, x_0, x_1)$$

Damped Harmonic Oscillator

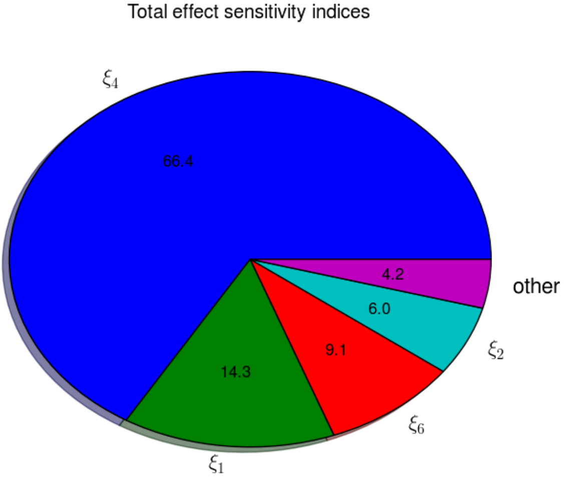


Figure: Total effect sensitivity indices for the random oscillator.

Diffusion Equation

$$-\frac{d}{dx}\left[a(x,\boldsymbol{\xi})\frac{du}{dx}(x,\boldsymbol{\xi})\right] = 1, \quad (x,\boldsymbol{\xi}) \in (0,1) \times I_{\boldsymbol{\xi}} \tag{2}$$

subject to the physical boundary conditions

$$u(0,\boldsymbol{\xi}) = 0, \quad u(1,\boldsymbol{\xi}) = 0 \tag{3}$$

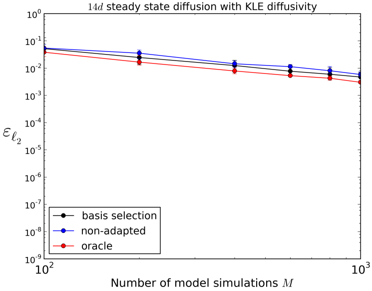
Furthermore assume that the random diffusivity satisfies

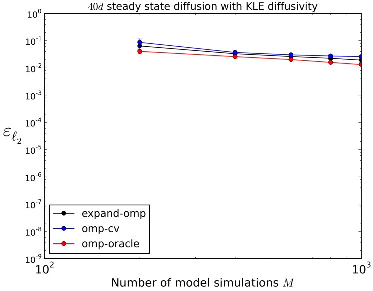
$$a(x,\boldsymbol{\xi}) = \bar{a} + \sigma_a \sum_{k=1}^{d} \sqrt{\lambda_k}\phi_k(x)\xi_k \tag{4}$$

where $\{\lambda_k\}_{k=1}^{d}$ and $\{\phi_k(x)\}_{k=1}^{d}$ are, respectively, the eigenvalues and eigenfunctions of the covariance kernel

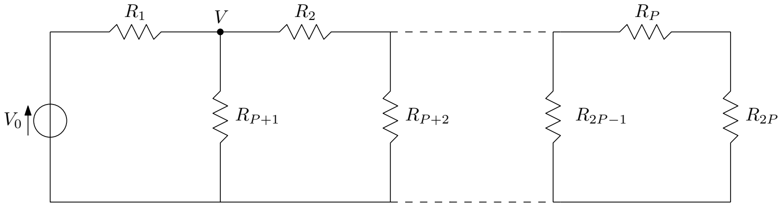
$$C_a(x_1,x_2) = \exp\left[-\frac{(x_1-x_2)^2}{l_c^2}\right]$$

Diffusion Equation

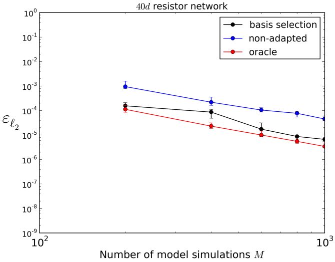


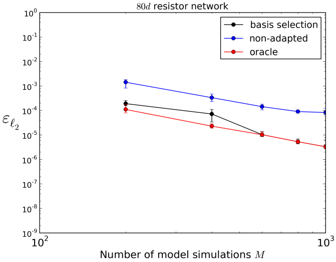


Resistor Network



$$R_k = \xi_k \in [1-\varepsilon, 1+\varepsilon], k = 1, \ldots, d$$





Plot sobol indices

Challenges

- Performing MCMC sampling in high dimensions
- Reducing the number of random variables used to represent the random friction field
- Efficiently building surrogates (today's topic)
- Constructing adaptive sampling strategies for small computational budgets
- Leveraging multi-fidelity models to build surrogates
- Determining the identifiability of the KLE modes: balance sampling of the model with the information content of the observational data
- Model structure error

Recent improvements to MCMC still require many model runs
Make comment about choosing correlation lengths