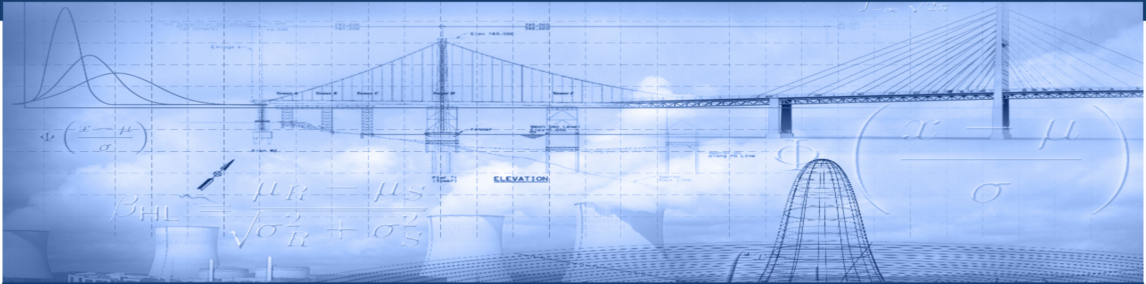
Polynomial chaos expansions in 90 minutes

Author(s):
Sudret, Bruno

Publication Date:
2021-10-07

Permanent Link:
https://doi.org/10.3929/ethz-b-000508852 →



Polynomial Chaos Expansions

B. Sudret

Chair of Risk, Safety and Uncertainty Quantification, ETH Zurich, Switzerland

How to cite?

This presentation has been delivered during a master class organized at the *4th National Conference on Multidisciplinary Design, Analysis and Optimization* of India, October 7-9, 2021 (https://www.ncmdao.org/).

How to cite

Sudret, B. *Polynomial chaos expansions* (2021), Master class, 4th National Conference on Multidisciplinary Design, Analysis and Optimization, Indian Institute of Technology Madras (India), October 7.

Chair of Risk, Safety and Uncertainty quantification

The Chair carries out research projects in the field of uncertainty quantification for engineering problems with applications in structural reliability, sensitivity analysis, model calibration, and reliability-based design optimization

Research topics

- Uncertainty modelling for engineering systems
- Structural reliability analysis
- Surrogate models (**polynomial chaos expansions**, Kriging, support vector machines)
- Bayesian model calibration and stochastic inverse problems
- **Global sensitivity analysis**
- Reliability-based design optimization

http://www.rsuq.ethz.ch

Computational models in engineering

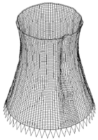
Complex engineering systems are designed and assessed using computational models, a.k.a simulators

A computational model combines:

- A mathematical description of the physical phenomena (governing equations), *e.g.* mechanics, electromagnetism, fluid dynamics, etc.

$$\operatorname{div} \boldsymbol{\sigma} + \boldsymbol{f} = \mathbf{0}$$
$$\boldsymbol{\sigma} = \mathbf{D} \cdot \boldsymbol{\varepsilon}$$
$$\boldsymbol{\varepsilon} = \frac{1}{2}\left(\nabla \boldsymbol{u} + {}^{\mathsf{T}}\nabla \boldsymbol{u}\right)$$

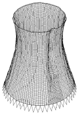
- Discretization techniques which transform continuous equations into linear algebra problems
- Algorithms to solve the discretized equations

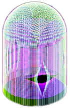


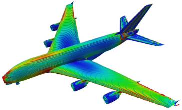
Computational models in engineering

Computational models are used:

- To explore the design space ("virtual prototypes")
- To optimize the system (*e.g.* minimize the mass) under performance constraints
- To assess its robustness w.r.t uncertainty and its reliability
- Together with experimental data for calibration purposes





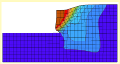


Computational models: the abstract viewpoint

A computational model may be seen as a black box program that computes quantities of interest (QoI) (a.k.a. model responses) as a function of input parameters

Vector of input parameters $\boldsymbol{x} \in \mathbb{R}^M$ — Computational model $\mathcal{M}$ — Model response $\boldsymbol{y} = \mathcal{M}(\boldsymbol{x}) \in \mathbb{R}^Q$

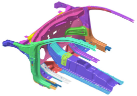
- Geometry
- Material properties
- Loading



- Analytical formula
- Finite element model
- Comput. workflow

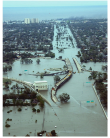
- Displacements
- Strains, stresses
- Temperature, etc.

Real world is uncertain



- Differences between the designed and the real system:
  - Dimensions (tolerances in manufacturing)
  - Material properties (*e.g.* variability of the stiffness or resistance)

- Unforecast exposures: exceptional service loads, natural hazards (earthquakes, floods, landslides), climate loads (hurricanes, snow storms, etc.), accidental human actions (explosions, fire, etc.)



Global framework for uncertainty quantification

Step B

Quantification of sources of uncertainty

Step A

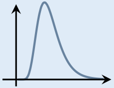
Model(s) of the system

Assessment criteria

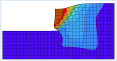
Step C

Uncertainty propagation

Random variables



Computational model



Moments

Probability of failure

Response PDF

Step C'

Sensitivity analysis

B. Sudret, *Uncertainty propagation and sensitivity analysis in mechanical models – contributions to structural reliability and stochastic spectral methods (2007)*

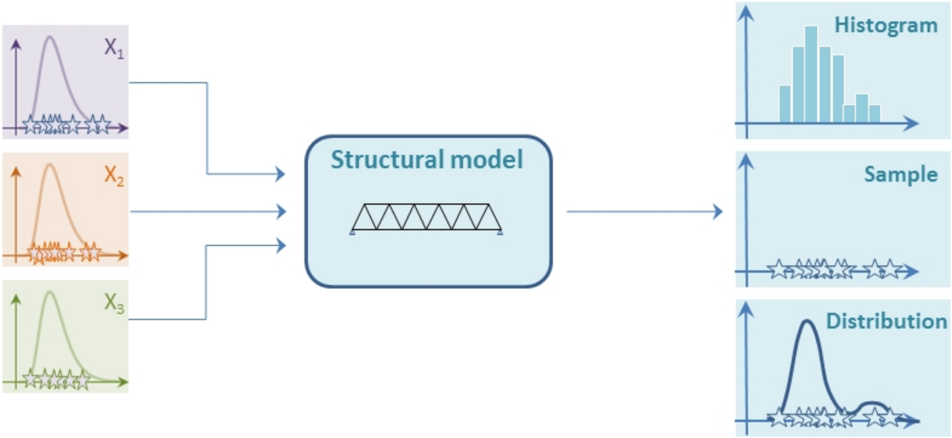
Monte Carlo simulation in UQ

- Monte Carlo simulation allows one to assess the performance of a large number of virtual systems featuring different realizations of the input parameters

Source: www.monaco.mc

- The input random variables are sampled according to their joint PDF $f_{\boldsymbol{X}}(\boldsymbol{x})$
- For each sample $\boldsymbol{x}^{(i)}$, the response $\mathcal{M}(\boldsymbol{x}^{(i)})$ is computed (possibly time-consuming)
- The response sample set $\mathbf{M} = \{\mathcal{M}(\boldsymbol{x}^{(1)}), \ldots, \mathcal{M}(\boldsymbol{x}^{(n)})\}^{\mathsf{T}}$ is used to compute statistical moments, probabilities of failure or estimate the response distribution (histograms, kernel densities)

Monte Carlo simulation in UQ



Monte Carlo simulation

Advantages

- It is a universal method, *i.e.* it does not depend on the type of model $\mathcal{M}$
- It is statistically well defined: convergence, confidence intervals, etc.
- It is non intrusive, *i.e.* it is based on repeated runs of the computational model as a black box
- It is suited to distributed computing (clusters of PCs)

Drawbacks

- The "scattering" of Y is investigated point-by-point: if two samples $\boldsymbol{x}^{(i)}, \boldsymbol{x}^{(j)}$ are almost equal, two independent runs of the model are carried out
- The convergence rate is low ($\propto N^{-1/2}$)

Surrogate models for uncertainty quantification

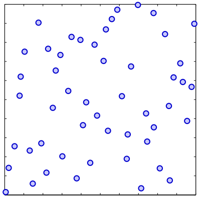
A surrogate model $\tilde{\mathcal{M}}$ is an approximation of the original computational model $\mathcal{M}$ with the following features:

- It is built from a limited set of runs of the original model $\mathcal{M}$ called the experimental design $\mathcal{X} = \left\{ \boldsymbol{x}^{(i)},\, i = 1, \, \dots \,, N \right\}$
- It assumes some regularity of the model $\mathcal{M}$ and some general functional shape

Name	Shape	Parameters
Polynomial chaos expansions	$\tilde{\mathcal{M}}(\boldsymbol{x}) = \sum_{\boldsymbol{\alpha} \in \mathcal{A}} a_{\boldsymbol{\alpha}} \, \Psi_{\boldsymbol{\alpha}}(\boldsymbol{x})$	$\boldsymbol{a}_{\boldsymbol{\alpha}}$
Low-rank tensor approximations	$\tilde{\mathcal{M}}(\boldsymbol{x}) = \sum_{l=1}^{R} b_l \left(\prod_{i=1}^{M} v_l^{(i)}(x_i) \right)$	$b_l ,\; z_{k,l}^{(i)}$
Kriging (a.k.a Gaussian processes)	$\tilde{\mathcal{M}}(\boldsymbol{x}) = \boldsymbol{\beta}^{\mathsf{T}} \cdot \boldsymbol{f}(\boldsymbol{x}) + Z(\boldsymbol{x}, \omega)$	$\boldsymbol{\beta} \,,\; \sigma_Z^2 \,,\; \boldsymbol{\theta}$
Support vector machines	$\tilde{\mathcal{M}}(\boldsymbol{x}) = \sum_{i=1}^{m} a_i \, K(\boldsymbol{x}_i, \boldsymbol{x}) + b$	$\boldsymbol{a} \,,\; b$

Ingredients for building a surrogate model

- Select an experimental design $\mathcal{X}$ that covers at best the domain of input parameters: Latin hypercube sampling (LHS), low-discrepancy sequences



- Run the computational model $\mathcal{M}$ onto $\mathcal{X}$ exactly as in Monte Carlo simulation

- Smartly post-process the data $\{\mathcal{X}\,,\,\mathcal{M}(\mathcal{X})\}$ through a learning algorithm

Name	Learning method
Polynomial chaos expansions	sparse grid integration, least-squares, compressive sensing
Low-rank tensor approximations	alternate least squares
Kriging	maximum likelihood, Bayesian inference
Support vector machines	quadratic programming

Advantages of surrogate models

Usage

$$\mathcal{M}(\boldsymbol{x}) \quad \approx \quad \tilde{\mathcal{M}}(\boldsymbol{x})$$

hours per run — seconds for 10^6 runs

Advantages

- Non-intrusive methods: based on runs of the computational model, exactly as in Monte Carlo simulation
- Suited to high performance computing: "embarrassingly parallel"

Challenges

- Need for rigorous validation
- Communication: advanced mathematical background

Efficiency: 2-3 orders of magnitude less runs compared to Monte Carlo

Outline

Polynomial chaos expansions in a nutshell

Ghanem & Spanos (1991; 2003); Xiu & Karniadakis (2002); Soize & Ghanem (2004)

- We assume here for simplicity that the input parameters are independent with $X_i \sim f_{X_i}$, $i = 1, \ldots, d$
- PCE is also applicable in the general case using an isoprobabilistic transform $\boldsymbol{X} \mapsto \boldsymbol{\Xi}$

The polynomial chaos expansion of the (random) model response reads:

$$Y = \sum_{\boldsymbol{\alpha} \in \mathbb{N}^d} y_{\boldsymbol{\alpha}} \, \Psi_{\boldsymbol{\alpha}}(\boldsymbol{X})$$

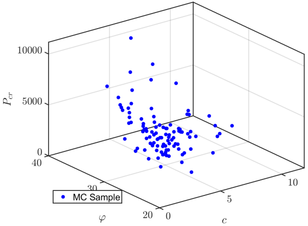
where:

- $\Psi_{\boldsymbol{\alpha}}(\boldsymbol{X})$ are basis functions (multivariate orthonormal polynomials)
- $y_{\boldsymbol{\alpha}}$ are coefficients to be computed (coordinates)

Sampling (MCS) vs. spectral expansion (PCE)

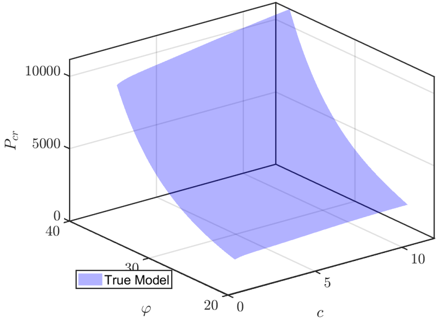
Whereas MCS explores the output space /distribution point-by-point, the polynomial chaos expansion assumes a generic structure (polynomial function), which better exploits the available information (runs of the original model)

Example: load bearing capacity as a function of (c, φ)



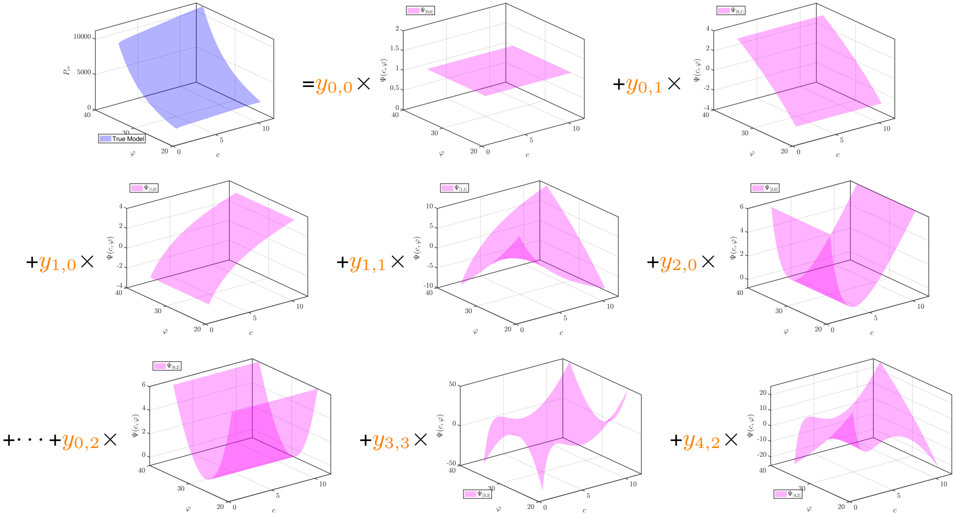
Thousands (resp. millions) of points are needed to grasp the structure of the response (resp. capture the rare events)

Visualization of the PCE construction



= "Sum of coefficients × basic surfaces"

Visualization of the PCE construction

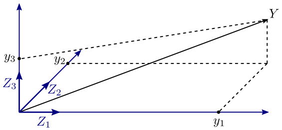


Polynomial chaos expansion: procedure

$$Y^{\text{PCE}} = \sum_{\boldsymbol{\alpha} \in \mathcal{A}} y_{\boldsymbol{\alpha}} \, \Psi_{\boldsymbol{\alpha}}(\boldsymbol{X})$$

Four steps

- How to construct the polynomial basis $\Psi_{\boldsymbol{\alpha}}(\boldsymbol{X})$ for given $X_i \sim f_{X_i}$?
- How to compute the coefficients $y_{\boldsymbol{\alpha}}$?
- How to check the accuracy of the expansion ?
- How to answer the engineering questions:
  - Mean, standard deviation
  - PDF, quantiles
  - Sensitivity indices



Basis and coordinates in a 3D space

Outline

Univariate orthogonal polynomials

- Suppose the input random vector has independent components:

$$f_{\boldsymbol{X}}(\boldsymbol{x}) = \prod_{i}^{M} f_{X_i}(x_i)$$

- For each marginal distribution $f_{X_i}(x_i)$, we define the inner product:

$$\langle \phi_1(x_i), \phi_2(x_i) \rangle = \int_{\mathcal{D}_i} \phi_1(x_i)\, \phi_2(x_i)\, f_{X_i}(x_i)\, dx_i$$

- By classical algebra one can build a family of orthogonal polynomials $\{P_k^{(i)},\ k \in \mathbb{N}\}$:

$$\left\langle P_j^{(i)}(x_i), P_k^{(i)}(x_i) \right\rangle = \int P_j^{(i)}(x_i)\ P_k^{(i)}(x_i)\ f_{X_i}(x_i)\, dx_i = \gamma_j^{(i)}\ \delta_{jk}$$

e.g. using the Gram-Schmit orthogonalization procedure of $\left\{1,\, x,\, x^2,\, x^3, \dots\right\}$

Classical orthogonal polynomials

Xiu & Karniadakis (2002)

- Classical families of orthogonal polynomials have been discovered historically when solving various problems of physics, quantum mechanics, etc.
- The name of the researcher who first investigated their properties is attached to them.

Type of variable	Weight function	Orthogonal polynomials	PCE basis $\psi_k(x)$
Uniform	$\mathbf{1}_{]-1,1[}(x)/2$	Legendre $P_k(x)$	$P_k(x)/\sqrt{\frac{1}{2k+1}}$
Gaussian	$\frac{1}{\sqrt{2\pi}} e^{-x^2/2}$	Hermite $He_k(x)$	$He_k(x)/\sqrt{k!}$
Gamma	$x^a\, e^{-x}\, \mathbf{1}_{\mathbb{R}^+}(x)$	Laguerre $L_k^a(x)$	$L_k^a(x)/\sqrt{\frac{\Gamma(k+a+1)}{k!}}$
Beta	$\mathbf{1}_{]-1,1[}(x)\, \frac{(1-x)^a (1+x)^b}{B(a)\, B(b)}$	Jacobi $J_k^{a,b}(x)$	$J_k^{a,b}(x)/\mathfrak{J}_{a,b,k}$
		$\mathfrak{J}^2_{a,b,k} = \frac{2^{a+b+1}}{2k+a+b+1}$	$\frac{\Gamma(k+a+1)\Gamma(k+b+1)}{\Gamma(k+a+b+1)\Gamma(k+1)}$

Legendre

Hermite

Laguerre

Jacobi

See details in Appendix

Multivariate polynomials

Tensor product of 1D polynomials

- One defines the multi-indices $\boldsymbol{\alpha} = \{\alpha_1, \ldots, \alpha_M\}$, of degree $|\boldsymbol{\alpha}| = \sum_{i=1}^{M} \alpha_i$
- The associated multivariate polynomial reads:

$$\Psi_{\boldsymbol{\alpha}}(\boldsymbol{x}) \stackrel{\text{def}}{=} \prod_{i=1}^{M} \Psi_{\alpha_i}^{(i)}(x_i)$$

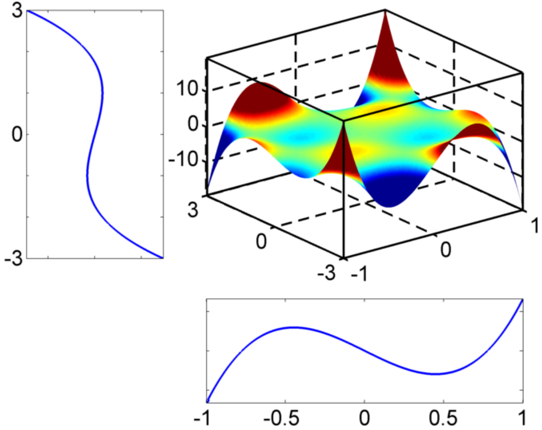
where $\Psi_{\alpha_i}^{(i)}(x_i)$ is the univariate polynomial of degree α_i from the orthonormal family associated to variable x_i

The set of multivariate polynomials $\{\Psi_{\boldsymbol{\alpha}},\ \boldsymbol{\alpha} \in \mathbb{N}^M\}$ forms a basis of the appropriate space:

$$Y = \sum_{\boldsymbol{\alpha} \in \mathbb{N}^M} y_{\boldsymbol{\alpha}}\, \Psi_{\boldsymbol{\alpha}}(\boldsymbol{X})$$

Example: multivariate polynomials in 2D ($M = 2$)

$\boldsymbol{\alpha} = [3\,,\,3]$ $\Psi_{(3,3)}(\boldsymbol{x}) = \tilde{P}_3(x_1) \cdot \tilde{He}_3(x_2)$



Outline

Dealing with complex input distributions

- Classical orthogonal polynomials are defined for reduced variables, *e.g.* :
  - Standard normal variables $\mathcal{N}(0,1)$
  - Standard uniform variables $\mathcal{U}(-1,1)$
- In practical UQ problems the physical parameters are modelled by random variables that are:
  - Not necessarily reduced, *e.g.* $X_1 \sim \mathcal{N}(\mu,\sigma)$, $X_2 \sim \mathcal{U}(a,b)$, etc.
  - Not necessarily from a classical family, *e.g.* lognormal variable
  - May show dependence modelled by a joint PDF

How to handle these cases?

Dealing with complex input distributions

Independent variables

Input parameters with given marginal CDFs $X_i \sim F_{X_i}, \quad i = 1, \ldots, M$

- Arbitrary PCE: orthogonal polynomial computed numerically on-the-fly

  Wan & Karniadakis (2006); Oladyshkin & Nowak (2012)

- Isoprobabilistic transform through a one-to-one mapping to reduced variables, *e.g.* :

$$X_i = F_{X_i}^{-1}\left(\frac{\xi_i + 1}{2}\right) \qquad \text{if } \xi_i \sim \mathcal{U}(-1\,,\,1)$$

$$X_i = F_{X_i}^{-1}\left(\Phi(\xi_i)\right) \qquad \text{if } \xi_i \sim \mathcal{N}(0,1)$$

General case: addressing dependence

Sklar's theorem (1959)

- The joint CDF is defined through its marginals and copula

$$F_{\boldsymbol{X}}(\boldsymbol{x}) = \mathcal{C}\left(F_{X_1}(x_1), \ldots, F_{X_M}(x_M)\right)$$

- Rosenblatt or Nataf isoprobabilistic transform is used

Standard truncation scheme

Premise

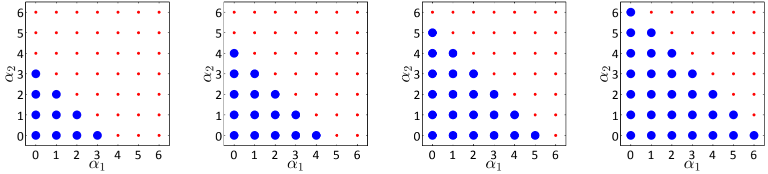
- The infinite series expansion cannot be handled in pratical computations
- A truncated series must be defined

Standard truncation scheme

Consider all multivariate polynomials of total degree $|\boldsymbol{\alpha}| = \sum_{i=1}^{M} \alpha_i$ less than or equal to p:

$$\mathcal{A}^{M,p} = \{\boldsymbol{\alpha} \in \mathbb{N}^M \; : \; |\boldsymbol{\alpha}| \leq p\} \qquad P \equiv \text{card } \mathcal{A}^{M,p} = \binom{M+p}{p} = \frac{(M+p)!}{M!\,p!}$$

$M = 2$ input variables



$|\boldsymbol{\alpha}| \leq 3$ $\quad$ $|\boldsymbol{\alpha}| \leq 4$ $\quad$ $|\boldsymbol{\alpha}| \leq 5$ $\quad$ $|\boldsymbol{\alpha}| \leq 6$

Mixed Legendre/Hermite polynomials

Computational model $Y = \mathcal{M}(X_1, X_2)$

Probabilistic model $X_1 \sim \mathcal{N}(\mu, \sigma) \quad ; \quad X_2 \sim \mathcal{U}(a, b)$

Isoprobabilistic transform

$$X_1 = \mu + \sigma \, \xi_1 \qquad \xi_1 \sim \mathcal{N}(0, 1)$$
$$X_2 = (a + b)/2 + (b - a)\xi_2/2 \qquad \xi_2 \sim \mathcal{U}(-1, 1)$$

Univariate polynomials

- Hermite polynomials in ξ_1, *i.e.* $\tilde{He}_n(\xi_1)$
- Legendre polynomials in ξ_2, *i.e.* $\tilde{P}_n(\xi_2)$

Multivariate polynomials

$$\Psi_{\alpha_1, \alpha_2}(\xi_1, \xi_2) = \tilde{He}_{\alpha_1}(\xi_1) \cdot \tilde{P}_{\alpha_2}(\xi_2)$$

Truncation example

Third order truncation $p = 3$

All the polynomials of ξ_1, ξ_2 that are product of univariate polynomials and whose total degree is less than 3 are considered

j	$\boldsymbol{\alpha}$	$\Psi_{\boldsymbol{\alpha}} \equiv \Psi_j$
0	$[0,0]$	$\Psi_0 = 1$
1	$[1,0]$	$\Psi_1 = \xi_1$
2	$[0,1]$	$\Psi_2 = \sqrt{3}\,\xi_2$
3	$[2,0]$	$\Psi_3 = (\xi_1^2 - 1)/\sqrt{2}$
4	$[1,1]$	$\Psi_4 = \xi_1 \sqrt{3}\,\xi_2$
5	$[0,2]$	$\Psi_5 = \sqrt{5/4}\,(3\xi_2^2 - 1)$
6	$[3,0]$	$\Psi_6 = (\xi_1^3 - 3\xi_1)/\sqrt{6}$
7	$[2,1]$	$\Psi_7 = \sqrt{3/2}\,(\xi_1^2 - 1)\xi_2$
8	$[1,2]$	$\Psi_8 = \sqrt{5/4}(3\xi_2^2 - 1)\xi_1$
9	$[0,3]$	$\Psi_9 = \sqrt{7/4}(5\xi_2^3 - 3\xi_2)$

$$
\begin{aligned}
\tilde{Y} \equiv \mathcal{M}^{\mathsf{PC}}(\xi_1, \xi_2) &= a_0 + a_1\,\xi_1 + a_2\,\sqrt{3}\,\xi_2 \\
&+ a_3\,(\xi_1^2 - 1)/\sqrt{2} + a_4\,\sqrt{3}\,\xi_1\xi_2 \\
&+ a_5\,\sqrt{5/4}\,(3\xi_2^2 - 1) + a_6\,(\xi_1^3 - 3\xi_1)/\sqrt{6} \\
&+ a_7\,\sqrt{3/2}\,(\xi_1^2 - 1)\xi_2 + a_8\,\sqrt{5/4}(3\xi_2^2 - 1)\xi_1 \\
&+ a_9\sqrt{7/4}(5\xi_2^3 - 3\xi_2)
\end{aligned}
$$

Conclusions

- Polynomial chaos expansions allow for an intrinsic representation of the random response as a series expansion
- The basis functions are multivariate orthonormal polynomials (based on the input distribution)
- Arbitrary PCE expansions can be computed numerically
- The input vector may also be transformed into independent reduced variables for which classical orthogonal polynomials are well-known
- A truncation scheme shall be introduced for pratical computations, *e.g.* by selecting the maximal degree of the polynomials
- Next step is the computation of the expansion coefficients

Outline

Polynomial chaos expansions

Computing and post-processing the PCE coefficients

- Least-square minimization
- Statistical moments and distribution
- Global sensitivity analysis

Sparse polynomial chaos expansions

Application examples

Various methods for computing the coefficients

Intrusive approaches

- Historical approaches: projection of the equations residuals in the Galerkin sense Ghanem & Spanos, 1991, 2003
- Proper generalized decompositions Nouy, 2007-2010

Non intrusive approaches

- Non intrusive methods consider the computational model $\mathcal{M}$ as a black box
- They rely upon a design of numerical experiments, *i.e.* a n-sample $\mathcal{X} = \{\boldsymbol{x}^{(i)} \in \mathcal{D}_{\boldsymbol{X}},\, i = 1, \ldots, n\}$ of the input parameters
- Different classes of methods are available:
  - Projection
  - Stochastic collocation
  - **Least-square minimization**
  - **Compressive sensing**

Statistical approach: least-square minimization

Principle

Berveiller *et al.* (2006)

The exact (infinite) series expansion is considered as the sum of a truncated series and a residual:

$$Y = \mathcal{M}(\boldsymbol{X}) = \sum_{\boldsymbol{\alpha} \in \mathcal{A}} y_{\boldsymbol{\alpha}} \Psi_{\boldsymbol{\alpha}}(\boldsymbol{X}) + \varepsilon_P \equiv \mathbf{Y}^{\mathsf{T}} \boldsymbol{\Psi}(\boldsymbol{X}) + \varepsilon_P(\boldsymbol{X})$$

where : $\mathbf{Y} = \{y_{\boldsymbol{\alpha}}, \, \boldsymbol{\alpha} \in \mathcal{A}\} \equiv \{y_0, \, \dots \, , y_{P-1}\}$ (P unknown coefficients)

$$\boldsymbol{\Psi}(\boldsymbol{x}) = \{\Psi_0(\boldsymbol{x}), \, \dots \, , \Psi_{P-1}(\boldsymbol{x})\}$$

Residual

$$\varepsilon_P(\boldsymbol{X}) = \mathcal{M}(\boldsymbol{X}) - \sum_{j=0}^{P-1} y_j \, \Psi_j(\boldsymbol{X})$$

Least-squares minimization: continuous solution

Least-square minimization

The unknown coefficients are estimated by minimizing the mean square residual error:

$$\hat{\mathbf{Y}} = \arg\min \mathbb{E}\left[\varepsilon_P^2(\boldsymbol{X})\right] = \arg\min \mathbb{E}\left[\left(\mathbf{Y}^{\mathsf{T}}\boldsymbol{\Psi}(\boldsymbol{X}) - \mathcal{M}(\boldsymbol{X})\right)^2\right]$$

Analytical solution (continuous case)

The least-square minimization problem may be solved analytically:

$$\hat{y}_{\boldsymbol{\alpha}} = \mathbb{E}\left[\mathcal{M}(\boldsymbol{X})\, \Psi_{\boldsymbol{\alpha}}(\boldsymbol{X})\right] \qquad \forall\, \boldsymbol{\alpha} \in \mathcal{A}$$

Coefficient $\hat{y}_{\boldsymbol{\alpha}}$ is the projection of the model onto polynomial $\Psi_{\boldsymbol{\alpha}}(\boldsymbol{X})$

See details in Appendix

Least-square minimization: discretized solution

Principle

An estimate of the mean square error (sample average) is minimized:

$$\hat{\mathbf{Y}} = \arg\min \hat{\mathbb{E}}\left[\left(\mathbf{Y}^{\mathsf{T}}\boldsymbol{\Psi}(\boldsymbol{X}) - \mathcal{M}(\boldsymbol{X})\right)^2\right]$$

$$= \arg\min \frac{1}{n}\sum_{i=1}^{n}\left(\mathbf{Y}^{\mathsf{T}}\boldsymbol{\Psi}(\boldsymbol{x}^{(i)}) - \mathcal{M}(\boldsymbol{x}^{(i)})\right)^2$$

$$= \arg\min \sum_{i=1}^{n}\left(\mathcal{M}(\boldsymbol{x}^{(i)}) - \sum_{j=0}^{P-1} y_j\,\Psi_j(\boldsymbol{x}^{(i)})\right)^2$$

Notation

- $\mathbf{A}_{ij} = \Psi_j\left(\boldsymbol{x}^{(i)}\right)$: experimental matrix of size $n \times P$
- $\mathbf{M}_i = \mathcal{M}(\boldsymbol{x}^{(i)})$: output of the computational model
- $\mathbf{Y} = \{y_0, \ldots, y_{P-1}\}$: unknown coefficients

Least-square minimization: discretized solution

- $\mathbf{M} - \mathbf{A}\mathbf{Y}$ is the vector containing the residuals
- The mean-square error is equal to $(\mathbf{M} - \mathbf{A}\mathbf{Y})^{\mathsf{T}} \cdot (\mathbf{M} - \mathbf{A}\mathbf{Y})$

Solution

$$\Delta = \sum_{i=1}^{n} \varepsilon_i^2 = (\mathbf{M} - \mathbf{A}\mathbf{Y})^{\mathsf{T}} \cdot (\mathbf{M} - \mathbf{A}\mathbf{Y})$$
$$= \mathbf{M}^{\mathsf{T}}\mathbf{M} - 2\,\mathbf{Y}^{\mathsf{T}}\mathbf{A}^{\mathsf{T}}\mathbf{M} + \mathbf{Y}^{\mathsf{T}}\left(\mathbf{A}^{\mathsf{T}}\mathbf{A}\right)\mathbf{Y}$$

- The mean-square error is minimized when its derivative w.r.t each unknown coefficient y_j vanishes:

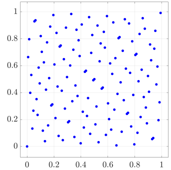
$$\frac{\partial \Delta}{\partial \mathbf{Y}^{\mathsf{T}}} = -2\,\mathbf{A}^{\mathsf{T}}\mathbf{M} + 2\,\left(\mathbf{A}^{\mathsf{T}}\mathbf{A}\right)\mathbf{Y} = 0$$

- This reduces to a linear system:

$$\hat{\mathbf{Y}} = \left(\mathbf{A}^{\mathsf{T}}\mathbf{A}\right)^{-1}\mathbf{A}^{\mathsf{T}}\mathbf{M}$$

Least-square minimization in a nutshell

- Select an experimental design $\mathcal{X} = \left\{ \boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(n)} \right\}^{\mathsf{T}}$ that covers at best the domain of variation of the parameters



- Evaluate the model response for each sample (exactly as in Monte carlo simulation)

$$\mathbf{M} = \left\{ \mathcal{M}(\boldsymbol{x}^{(1)}), \ldots, \mathcal{M}(\boldsymbol{x}^{(n)}) \right\}^{\mathsf{T}}$$

- Compute the experimental matrix

$$\mathbf{A}_{ij} = \Psi_j \left(\boldsymbol{x}^{(i)} \right) \quad i = 1, \ldots, n \; ; \; j = 0, \ldots, P-1$$

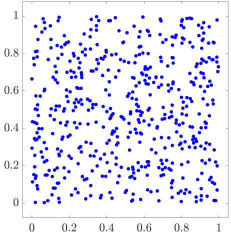
- Solve the resulting linear system

$$\hat{\mathbf{Y}} = (\mathbf{A}^{\mathsf{T}} \mathbf{A})^{-1} \mathbf{A}^{\mathsf{T}} \mathbf{M}$$

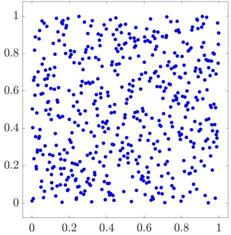
Choice of the experimental design

Random designs

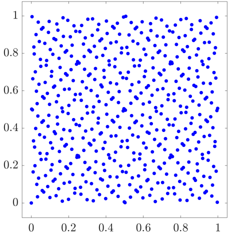
- Monte Carlo samples obtained by standard random generators
- Latin Hypercube designs that are both random and "space-filling"
- Quasi-random sequences (*e.g.* Sobol' sequence)



Monte Carlo



Latin Hypercube Sampling



Sobol sequence

Outline

Post-processing of polynomial chaos expansions

Polynomial chaos

$$Y = \mathcal{M}(\boldsymbol{X}) = \sum_{\boldsymbol{\alpha} \in \mathbb{N}^M} y_{\boldsymbol{\alpha}} \, \Psi_{\boldsymbol{\alpha}}(\boldsymbol{X})$$

Truncated series

$$Y^{PC} = \sum_{\boldsymbol{\alpha} \in \mathcal{A}} y_{\boldsymbol{\alpha}} \, \Psi_{\boldsymbol{\alpha}}(\boldsymbol{X})$$

- The computed coefficients ("coordinates" of the random variable in the PCE basis) are not the quantities of interest
- Depending on the situation, the PDF, the statistical moments or quantiles of Y are of interest (*e.g.* low quantiles in structural reliability analysis)

> The PC expansion must be post-processed in order to get relevant information on the model response

Mean value and variance

From the orthonormality of the polynomial chaos basis one gets:

$$\mathbb{E}\left[\Psi_{\boldsymbol{\alpha}}(\boldsymbol{X})\right] = 0 \qquad \mathbb{E}\left[\Psi_{\boldsymbol{\alpha}}(\boldsymbol{X})\Psi_{\boldsymbol{\beta}}(\boldsymbol{X})\right] = 0 \qquad \boldsymbol{\alpha} \neq \boldsymbol{\beta}$$

Mean value

$$\hat{\mu}_Y = y_0$$

The mean value is the first coefficient of the series

Variance

$$\hat{\sigma}_Y^2 \overset{\text{def}}{=} \mathbb{E}\left[\left(Y^{PC} - \hat{\mu}_Y\right)^2\right] = \mathbb{E}\left[\left(\sum_{\boldsymbol{\alpha}\in\mathcal{A}\setminus\boldsymbol{0}} y_{\boldsymbol{\alpha}}\,\Psi_{\boldsymbol{\alpha}}(\boldsymbol{X})\right)^2\right]$$

$$\hat{\sigma}_Y^2 = \sum_{\boldsymbol{\alpha}\in\mathcal{A}\setminus\boldsymbol{0}} y_{\boldsymbol{\alpha}}^2$$

The variance is the sum of the squares of the remaining coefficients

Higher order statistical moments

Skewness coefficient $\hat{\delta}_Y$

$$\mathbb{E}\left[\left(Y^{PC} - \hat{\mu}_Y\right)^3\right] = \mathbb{E}\left[\left(\sum_{\boldsymbol{\alpha}\in\mathcal{A}\setminus\mathbf{0}} y_{\boldsymbol{\alpha}}\,\Psi_{\boldsymbol{\alpha}}(\boldsymbol{X})\right)^3\right]$$

$$= \sum_{\boldsymbol{\alpha}\in\mathcal{A}\setminus\mathbf{0}}\ \sum_{\boldsymbol{\beta}\in\mathcal{A}\setminus\mathbf{0}}\ \sum_{\boldsymbol{\gamma}\in\mathcal{A}\setminus\mathbf{0}} y_{\boldsymbol{\alpha}} y_{\boldsymbol{\beta}} y_{\boldsymbol{\gamma}}\ \mathbb{E}\left[\Psi_{\boldsymbol{\alpha}}(\boldsymbol{X})\Psi_{\boldsymbol{\beta}}(\boldsymbol{X})\Psi_{\boldsymbol{\gamma}}(\boldsymbol{X})\right]$$

Kurtosis coefficient $\hat{\kappa}_Y$

$$\mathbb{E}\left[\left(Y^{PC} - \hat{\mu}_Y\right)^4\right] = \mathbb{E}\left[\left(\sum_{\boldsymbol{\alpha}\in\mathcal{A}\setminus\mathbf{0}} y_{\boldsymbol{\alpha}}\,\Psi_{\boldsymbol{\alpha}}\right)^4\right]$$

$$= \sum_{\boldsymbol{\alpha}\in\mathcal{A}\setminus\mathbf{0}}\ \sum_{\boldsymbol{\beta}\in\mathcal{A}\setminus\mathbf{0}}\ \sum_{\boldsymbol{\gamma}\in\mathcal{A}\setminus\mathbf{0}}\ \sum_{\boldsymbol{\delta}\in\mathcal{A}\setminus\mathbf{0}} y_{\boldsymbol{\alpha}} y_{\boldsymbol{\beta}} y_{\boldsymbol{\gamma}} y_{\boldsymbol{\delta}}\ \mathbb{E}\left[\Psi_{\boldsymbol{\alpha}}(\boldsymbol{X})\Psi_{\boldsymbol{\beta}}(\boldsymbol{X})\Psi_{\boldsymbol{\gamma}}(\boldsymbol{X})\Psi_{\boldsymbol{\delta}}(\boldsymbol{X})\right]$$

- Requires evaluating the expectation of products of 3, 4, etc. polynomials
- Analytical formulæ exist only in case of Hermite polynomials. Otherwise the expectation may be computed exactly using sparse quadrature rules

Probability density function

- The polynomial series expansion may be considered as a stochastic response surface, *i.e.* an analytical function of the input variables $\boldsymbol{\xi}$ (after some isoprobabilistic transform), which may be sampled easily using Monte Carlo simulation.
- A large sample set $\boldsymbol{\xi}$ of reduced variables is drawn, say of size $n_{sim} = 10^5 - 10^6$:

$$\mathcal{X}_{sim} = \left\{ \boldsymbol{\xi}_j, \; j = 1, \, \dots \, , n_{sim} \right\}$$

- The truncated series is evaluated onto this sample:

$$\mathcal{Y}_{sim} = \left\{ \mathfrak{y}_j = \sum_{\boldsymbol{\alpha} \in \mathcal{A}} y_{\boldsymbol{\alpha}} \, \Psi_{\boldsymbol{\alpha}}(\boldsymbol{\xi}_j), \; j = 1, \, \dots \, , n_{sim} \right\}$$

- The obtained sample set is plotted using histograms or kernel density smoothing

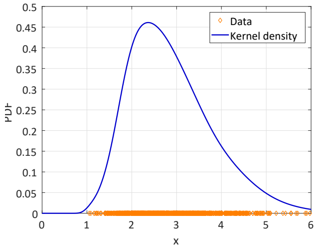
Probability density function

Response sample set

$$\mathcal{Y}_{sim} = \left\{ \mathfrak{y}_j = \sum_{\boldsymbol{\alpha} \in \mathcal{A}} y_{\boldsymbol{\alpha}} \, \Psi_{\boldsymbol{\alpha}}(\boldsymbol{\xi}_j), \; j = 1, \ldots, n_{sim} \right\}$$

Kernel smoothing

$$\hat{f}_Y(y) = \frac{1}{n_{sim}\, h} \sum_{j=1}^{n_{sim}} K\left(\frac{y - \mathfrak{y}_j}{h}\right)$$



- Kernel function : $K(t) = \frac{1}{\sqrt{2\pi}} e^{-t^2/2}$
- Bandwidth:

  $$h = 0.9\, n_{sim}{}^{-1/5} \, \min\left(\hat{\sigma}_{\mathcal{Y}}, (Q_{0.75} - Q_{0.25})/1.34\right)$$

  where $(Q_{0.75} - Q_{0.25})$ is the inter-quartile range computed from the sample

Outline

Sensitivity analysis

Goal

Sobol' (1993); Saltelli *et al.* (2008)

Global sensitivity analysis aims at quantifying which input parameter(s) (or combinations thereof) influence the most the response variability (variance decomposition)

Hoeffding-Sobol' decomposition

$(\boldsymbol{X} \sim \mathcal{U}([0,1]^M))$

$$\mathcal{M}(\boldsymbol{x}) = \mathcal{M}_0 + \sum_{i=1}^{M} \mathcal{M}_i(x_i) + \sum_{1 \le i < j \le M} \mathcal{M}_{ij}(x_i, x_j) + \cdots + \mathcal{M}_{12\ldots M}(\boldsymbol{x})$$

$$= \mathcal{M}_0 + \sum_{\mathbf{u} \subset \{1, \ldots, M\}} \mathcal{M}_{\mathbf{u}}(\boldsymbol{x}_{\mathbf{u}}) \qquad (\boldsymbol{x}_{\mathbf{u}} \stackrel{\text{def}}{=} \{x_{i_1}, \ldots, x_{i_s}\})$$

- The summands satisfy the orthogonality condition:

$$\int_{[0,1]^M} \mathcal{M}_{\mathbf{u}}(\boldsymbol{x}_{\mathbf{u}}) \, \mathcal{M}_{\mathbf{v}}(\boldsymbol{x}_{\mathbf{v}}) \, d\boldsymbol{x} = 0 \qquad \forall \, \mathbf{u} \neq \mathbf{v}$$

Sobol' indices

Total variance:

$$D \equiv \mathrm{Var}\left[\mathcal{M}(\boldsymbol{X})\right] = \mathrm{Var}\left[\sum_{\mathbf{u}\subset\{1,\ldots,M\}} \mathcal{M}_{\mathbf{u}}(\boldsymbol{X}_{\mathbf{u}})\right] = \sum_{\mathbf{u}\subset\{1,\ldots,M\}} \mathrm{Var}\left[\mathcal{M}_{\mathbf{u}}(\boldsymbol{X}_{\mathbf{u}})\right]$$

- Sobol' indices:

$$S_{\mathbf{u}} \overset{\text{def}}{=} \frac{\mathrm{Var}\left[\mathcal{M}_{\mathbf{u}}(\boldsymbol{X}_{\mathbf{u}})\right]}{D}$$

- First-order Sobol' indices:

$$S_i = \frac{D_i}{D} = \frac{\mathrm{Var}\left[\mathcal{M}_i(X_i)\right]}{D}$$

  Quantify the additive effect of each input parameter separately

- Total Sobol' indices:

$$S_i^T \overset{\text{def}}{=} \sum_{\mathbf{u}\supset i} S_{\mathbf{u}}$$

  Quantify the total effect of X_i, including interactions with the other variables.

Link with PC expansions

Sobol decomposition of a PC expansion

Sudret, CSM (2006); RESS (2008)

Obtained by reordering the terms of the (truncated) PC expansion $\mathcal{M}^{\text{PC}}(\boldsymbol{X}) \stackrel{\text{def}}{=} \sum_{\boldsymbol{\alpha} \in \mathcal{A}} y_{\boldsymbol{\alpha}} \, \Psi_{\boldsymbol{\alpha}}(\boldsymbol{X})$

Interaction sets

$$\text{For a given } \mathbf{u} \stackrel{\text{def}}{=} \{i_1, \, \dots \, , i_s\} : \qquad \mathcal{A}_{\mathbf{u}} = \{\boldsymbol{\alpha} \in \mathcal{A} \; : \; k \in \mathbf{u} \Leftrightarrow \alpha_k \neq 0\}$$

$$\mathcal{M}^{\text{PC}}(\boldsymbol{x}) = \mathcal{M}_0 + \sum_{\mathbf{u} \subset \{1, \dots, M\}} \mathcal{M}_{\mathbf{u}}(\boldsymbol{x}_{\mathbf{u}}) \qquad \text{where} \qquad \mathcal{M}_{\mathbf{u}}(\boldsymbol{x}_{\mathbf{u}}) \stackrel{\text{def}}{=} \sum_{\boldsymbol{\alpha} \in \mathcal{A}_{\mathbf{u}}} y_{\boldsymbol{\alpha}} \, \Psi_{\boldsymbol{\alpha}}(\boldsymbol{x})$$

PC-based Sobol' indices

$$S_{\mathbf{u}} = D_{\mathbf{u}}/D = \sum_{\boldsymbol{\alpha} \in \mathcal{A}_{\mathbf{u}}} y_{\boldsymbol{\alpha}}^2 / \sum_{\boldsymbol{\alpha} \in \mathcal{A} \setminus \mathbf{0}} y_{\boldsymbol{\alpha}}^2$$

The Sobol' indices are obtained analytically, at any order from the coefficients of the PC expansion

Outline

Validation of the PC expansion

- The truncated series expansions are convergent in the mean-square sense. However one does not know in advance where to truncate (problem-dependent)
- Most people truncate the series according to the total maximal degree of the polynomials, say $p = 2, 3, 4$, etc. Several values of p are tested until some kind of convergence is “empirically” observed
- Recent research deals with the development of error estimates through cross-validation in the least-square minimization approach

Error estimators

Coefficient of determination

- The least-squares technique is based on the minimization of the mean-square error. The generalization error is defined as:

$$E_{gen} = \mathbb{E}\left[\left(\mathcal{M}(\boldsymbol{X}) - \mathcal{M}^{\mathsf{PC}}(\boldsymbol{X})\right)^2\right] \qquad \mathcal{M}^{\mathsf{PC}}(\boldsymbol{X}) = \sum_{\boldsymbol{\alpha} \in \mathcal{A}} y_{\boldsymbol{\alpha}} \, \Psi_{\boldsymbol{\alpha}}(\boldsymbol{X})$$

- It may be estimated by the empirical error using the already computed response quantities ($\mathcal{Y} = \left\{\mathcal{M}(\boldsymbol{x}^{(i)}),\, i = 1, \ldots, n\right\}$):

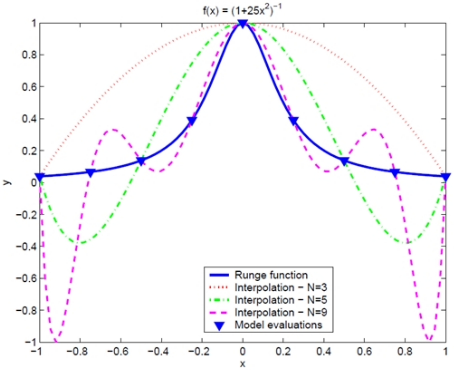
$$E_{emp} = \frac{1}{n} \sum_{i=1}^{n} \left(\mathcal{M}(\boldsymbol{x}^{(i)}) - \mathcal{M}^{\mathsf{PC}}(\boldsymbol{x}^{(i)})\right)^2$$

- The coefficient of determination R^2 is often used as an error estimator:

$$R^2 = 1 - \frac{E_{emp}}{\mathrm{Var}\left[\mathcal{Y}\right]} \qquad \mathrm{Var}\left[\mathcal{Y}\right] = \frac{1}{n}(\mathcal{M}(\boldsymbol{x}^{(i)}) - \bar{\mathcal{Y}})^2$$

This error estimator leads to overfitting

Overfitting – Illustration of the Runge effect



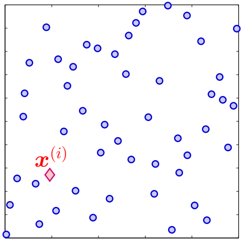
- If the degree of the polynomial model is equal to the size of the experimental design, one gets an interpolating approximation
- The empirical error is zero whereas the approximation gets worse and worse

Leave-one-out cross validation

Principle

- In statistical learning theory, cross validation consists in splitting the experimental design $\mathcal{Y}$ into two parts, namely a training set (which is used to build the model) and a validation set
- The leave-one-out cross validation technique consists in using each point of the experimental design as a single validation point for the meta-model built from the remaining $(n-1)$ points
- n different meta-models are built, and for each of them the empirical error is estimated on the remaining point. The resulting n errors are finally mean-square averaged

Leave-one-out cross validation



- An experimental design $\mathcal{X} = \{\boldsymbol{x}^{(j)},\ j = 1, \dots, n\}$ is selected
- For each $\boldsymbol{x}^{(i)}$, a polynomial chaos expansion is built using the following experimental design: $\mathcal{X} \backslash \boldsymbol{x}^{(i)} = \{\boldsymbol{x}^{(j)},\ j = 1, \dots, n,\ j \neq i\}$, denoted by $\mathcal{M}^{PC \backslash i}(.)$

- The predicted residual is computed in point $\boldsymbol{x}^{(i)}$:

$$\Delta_i = \mathcal{M}(\boldsymbol{x}^{(i)}) - \mathcal{M}^{PC \backslash i}(\boldsymbol{x}^{(i)})$$

- The procedure is used for each sample point in $\mathcal{X}$ and the results are averaged in the PRESS coefficient (*predicted residual sum of squares*):

$$PRESS = \sum_{i=1}^{n} \Delta_i^2$$

Leave-one-out error estimation

Reminder

The relative generalization error ε_{gen} reads:

$$\varepsilon_{gen} = \mathbb{E}\left[\left(\mathcal{M}(\boldsymbol{X}) - \mathcal{M}^{\text{PC}}(\boldsymbol{X})\right)^2\right] / \text{Var}\left[Y\right]$$

Leave-one-out error

$$E_{\text{LOO}} = \frac{1}{n}\sum_{i=1}^{n}\left(\mathcal{M}(\boldsymbol{x}^{(i)}) - \mathcal{M}^{PC\setminus i}(\boldsymbol{x}^{(i)})\right)^2$$

$$\varepsilon_{\text{LOO}} = \frac{\sum_{i=1}^{n}\left(\mathcal{M}(\boldsymbol{x}^{(i)}) - \mathcal{M}^{PC\setminus i}(\boldsymbol{x}^{(i)})\right)^2}{\sum_{i=1}^{n}\left(\mathcal{M}(\boldsymbol{x}^{(i)}) - \mu_{\mathcal{Y}}\right)^2} \qquad \mu_{\mathcal{Y}} = \frac{1}{n}\sum_{i=1}^{n}\mathcal{M}(\boldsymbol{x}^{(i)})$$

Problem: Do we really need a new meta-model based on $\mathcal{X}\setminus\boldsymbol{x}^{(i)} = \left\{\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(i-1)}, \boldsymbol{x}^{(i+1)}, \ldots, \boldsymbol{x}^{(n)}\right\}$ to compute Δ_i^2 ?

Leave-one-out: practical implementation

In practice one does not need to explicitly derive the n different models $\mathcal{M}^{PC\setminus i}(.)$

- In contrast, a single least-square analysis using $\mathcal{X}$ is carried out. The predicted residual reads:

$$\Delta_i = \mathcal{M}(\boldsymbol{x}^{(i)}) - \mathcal{M}^{PC\setminus i}(\boldsymbol{x}^{(i)}) = \frac{\mathcal{M}(\boldsymbol{x}^{(i)}) - \mathcal{M}^{PC}(\boldsymbol{x}^{(i)})}{1 - h_i}$$

where h_i is the i-th diagonal term of matrix $\mathbf{A}(\mathbf{A}^\mathsf{T}\mathbf{A})^{-1}\mathbf{A}^\mathsf{T}$, where:

$$\mathbf{A}_{ij} = \Psi_j(\boldsymbol{x}^{(i)})$$

- Thus:

$$E_{\text{LOO}} = \frac{1}{n}\sum_{i=1}^{n}\left(\frac{\mathcal{M}(\boldsymbol{x}^{(i)}) - \mathcal{M}^{PC}(\boldsymbol{x}^{(i)})}{1 - h_i}\right)^2$$

Conclusion

Given a truncation set $\mathcal{A}$ and an experimental design $\mathcal{X} = \left\{\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(n)}\right\}$:

- A polynomial chaos expansion can be computed, provided:

$$|\mathcal{X}| \geq k \cdot |\mathcal{A}| \qquad k = 2; 3$$

- An a posteriori error estimator allows one to check the accuracy of the approximation in the mean-square sense

Adaptive polynomial chaos expansions

- Assume a prescribed tolerance, *e.g.* $TOL = 10^{-3}$ is chosen
- An iterative algorithm may be run, increasing the candidate basis $\mathcal{A}$ until $\varepsilon_{\text{LOO}} < TOL$, *e.g.* with different $\mathcal{A}^{M,p}$ with $p = 1, 2, 3, \ldots$

Algorithm 1: Ordinary least-squares

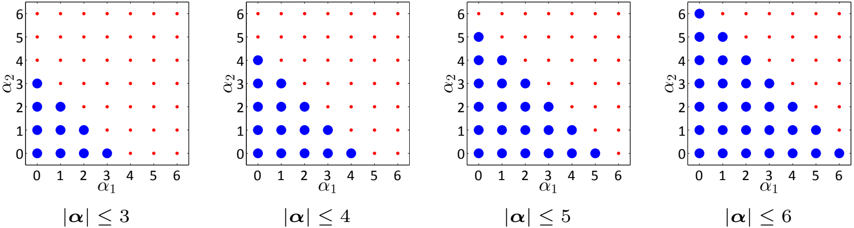
Input: Computational budget n
Initialization
  Experimental design $\mathcal{X} = \{\boldsymbol{x}^{(1)}, \dots, \boldsymbol{x}^{(n)}\}$
  Run model $\mathcal{Y} = \{\mathcal{M}(\boldsymbol{x}^{(1)}), \dots, \mathcal{M}(\boldsymbol{x}^{(n)})\}$
PCE construction
  for $p = p_{\min} : p_{\max}$ **do**
    Select candidate basis $\mathcal{A}^{M,p}$
    Solve OLS problem
    Compute $\varepsilon_{\text{LOO}}(p)$
  end
  $p^* = \arg\min \varepsilon_{\text{LOO}}(p)$
Return Best PCE of degree p^*

Outline

Classical truncation scheme

Classical truncation scheme

- Polynomials $\Psi_{\boldsymbol{\alpha}}$ with a total degree $|\boldsymbol{\alpha}| = \alpha_1 + \cdots + \alpha_M \leq p$ are usually selected



- The cardinality of such a truncated basis reads:

$$\text{card } \mathcal{A}^{M,p} = \binom{M+p}{p} = \frac{(M+p)!}{M!\,p!}$$

Curse of dimensionality: example

Size of the truncated PC basis $P \stackrel{\text{def}}{=} \left|\mathcal{A}^{M,p}\right|$

$M \backslash p$	2	3	5	7	10
2	6	10	21	36	66
3	10	20	56	120	286
5	21	56	252	792	3,003
10	66	286	3,003	19,448	184,756
50	1,326	23,426	3,478,761	264,385,836	75,394,027,566
100	5,151	176,851	96,560,646	26,075,972,546	46,897,636,623,981

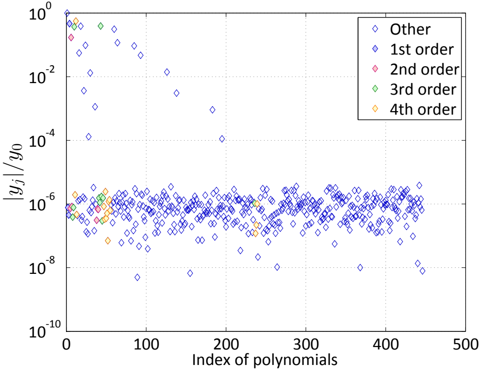
- Using the least-square approach the computational cost is related to the size of the experimental design:

$$n = k\,P \qquad \text{where } k = 2 - 3$$

Why are sparse representations relevant?

Example: Ishigami function

$$\mathcal{M}(\boldsymbol{x}) = \sin(x_1) + 7\,\sin^2(x_2) + 0.1\,x_3^4\,\sin(x_1)$$



- $M = 3$ input variables $X_1, X_2, X_3 \sim \mathcal{U}(-\pi, \pi)$
- $p = 12$
- $P = 455$ coefficients

Low-rank truncation schemes

Sparsity-of-effects principle

In most practical problems, only low-order interactions between the input variables are relevant. One shall select PC approximations using low-rank monomials

Definition

The rank of a multi-index $\boldsymbol{\alpha}$ is the number of active variables of $\Psi_{\boldsymbol{\alpha}}$, *i.e.* the number of non-zero terms in $\boldsymbol{\alpha}$:

$$||\boldsymbol{\alpha}||_0 = \sum_{i=1}^{M} \mathbf{1}_{\{\alpha_i > 0\}}$$

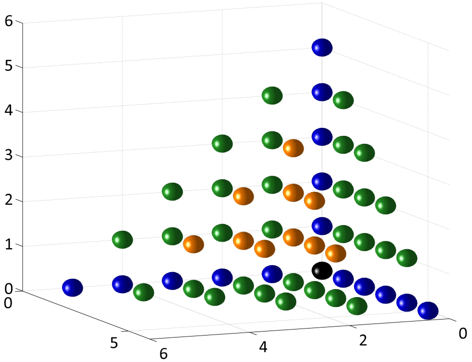
Example: $M = 5, p = 5$, Legendre polynomial chaos

$\boldsymbol{\alpha}$	$\Psi_{\boldsymbol{\alpha}}$	Rank
$[0\ 0\ 0\ 3\ 0]$	$\tilde{P}_3(x_4)$	1
$[2\ 0\ 0\ 0\ 1]$	$\tilde{P}_2(x_1) \cdot \tilde{P}_1(x_5)$	2
$[1\ 1\ 2\ 0\ 1]$	$\tilde{P}_1(x_1) \cdot \tilde{P}_1(x_2) \cdot \tilde{P}_2(x_3) \cdot \tilde{P}_1(x_5)$	4

Low-rank truncation set

Definition

$$\mathcal{A}^{M,p,r} = \{\boldsymbol{\alpha} \in \mathbb{N}^M \; : \; |\boldsymbol{\alpha}| \leq p, ||\boldsymbol{\alpha}||_0 \leq r\} \qquad r \leq p \; , \; r \leq M$$



All ranks ≤ 3

Hyperbolic truncation sets

Definition

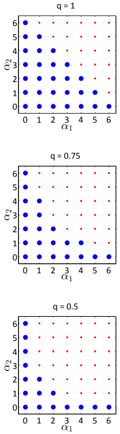
Blatman (2009), Blatman & Sudret, J. Comp. Phys (2011)

- The $q-$norm of a multi-index $\boldsymbol{\alpha}$ is defined by:

$$||\boldsymbol{\alpha}||_q \equiv \left(\sum_{i=1}^{M} \alpha_i^q\right)^{1/q}, \quad 0 < q < 1$$

- The hyperbolic truncation sets read:

$$\mathcal{A}_q^{M,p} = \{\boldsymbol{\alpha} \in \mathbb{N}^M \ : \ ||\boldsymbol{\alpha}||_q \leq p\}$$

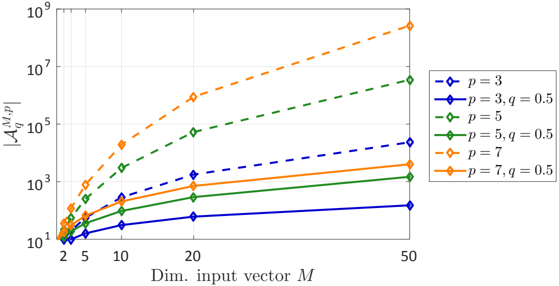


Limit cases

- $q = 1$: standard truncation scheme (all polynomials of maximal total degree p)
- $q \to 0$: additive model (no interaction)

Size of hyperbolic truncation sets

- For a given value of $0 < q \leq 1$, the index of sparsity tends to zero when M and p increase



Conclusions

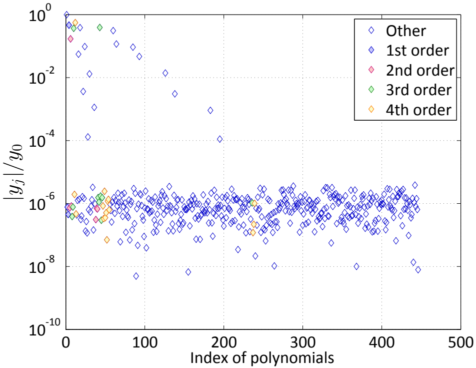
- For practical computations PC expansions have to be truncated
- The classical truncation scheme selects all polynomials up to a certain total degree, which leads to:

$$P = \frac{(M+p)!}{M!\,p!} \qquad \text{terms}$$

- This number blows up when $M > 10$ and / or $p > 5$
- The sparsity-of-effect principle allows one to select a priori truncation schemes with low-order interaction terms
- This can be achieved by limiting the rank of the polynomials or using an hyperbolic truncation scheme

Outline

Introduction



- Even when selecting a reduced set of polynomials a priori, most coefficients are negligible
- How to compute only the relevant basis function and associated coefficients?

Sparse polynomial chaos expansions

How to get sparse expansions?

Blatman & Sudret, JCP (2011)

- Finding the significant coefficients in the PC expansion is a variable selection problem
- It can be addressed by regularized regression techniques:

$$\boldsymbol{y}_{\boldsymbol{\alpha}} = \arg\min \frac{1}{n} \sum_{i=1}^{n} \left(\mathsf{Y}^{\mathsf{T}} \boldsymbol{\Psi}(\boldsymbol{x}^{(i)}) - \mathcal{M}(\boldsymbol{x}^{(i)}) \right)^2 + \lambda \parallel \boldsymbol{y}_{\boldsymbol{\alpha}} \parallel_m$$

Interpretation

- The regularization term:

$$\parallel \boldsymbol{y}_{\boldsymbol{\alpha}} \parallel_m = \sum_{j=1}^{|\mathcal{A}|} |y_j|^m$$

  corresponds to solving the least-square minimization under the constraint that the coefficients are "not too big"

- This avoids overfitting

Regularized regression: LASSO and least-angle regression

- Lasso corresponds to a L_1-norm ($m = 1$) penalization term:

$$\parallel \boldsymbol{y}_{\boldsymbol{\alpha}} \parallel_1 = \sum_{j=1}^{|\mathcal{A}|} |y_j|$$

- By selecting L_1 penalization, sparse solutions are favoured, *i.e.* solutions in which most of the coefficients in $\{\boldsymbol{y}_{\boldsymbol{\alpha}}, \ \boldsymbol{\alpha} \in \mathcal{A}\}$ are zero
- Least Angle Regression (LAR) is an efficient algorithm that solves the Lasso problem for different values of the penalty constant in a single run
- Various PC expansions are constructed with $1, 2, \dots, \min(n, |\mathcal{A}|)$ terms
- Among those models the best one is retained by comparing the leave-one-out cross validation error

Algorithm 2: LAR-based Sparse polynomial chaos expansion

Input: Computational budget n
Initialization
  Sample experimental design $\mathcal{X} = \{\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(n)}\}$
  Evaluate model response $\mathcal{Y} = \{\mathcal{M}(\boldsymbol{x}^{(1)}), \ldots, \mathcal{M}(\boldsymbol{x}^{(n)}\})$
PCE construction
  for $p = p_{\min} : p_{\max}$ **do**
    for $q \in \mathcal{Q}$ **do**
      Select candidate basis $\mathcal{A}_q^{M,p}$
      Run LAR for extracting the optimal sparse basis $\mathcal{A}^*(p, q)$
      Compute coefficients $\{y_{\boldsymbol{\alpha}},\ \boldsymbol{\alpha} \in \mathcal{A}^*(p, q)\}$ by OLS
      Compute $\varepsilon_{\text{LOO}}(p, q)$
    end
  end
  $(p^*, q^*) = \arg\min \varepsilon_{\text{LOO}}(p, q)$
Return Optimal sparse basis $\mathcal{A}^*(p, q)$, PCE coefficients, $\varepsilon_{\text{LOO}}(p^*, q^*)$

Conclusions

- Sparse PC expansions can be computed from a given experimental design using appropriate sparse solvers

  Lüthen, Marelli & Sudret, Sparse polynomial chaos expansions: Literature survey and benchmark, SIAM/ASA J. Unc. Quant., 2021.

  Lüthen, Marelli & Sudret, A benchmark of basis-adaptive sparse polynomial chaos expansions for engineering regression problems, Int. J. Uncertainty Quantification 2021.

- Problems with up to $\mathcal{O}(100)$ variables can be solved nowadays with $100 - 1000$ model runs

- Fully automated algorithms allow to get “the best PCE surrogate” given the data, and a fair estimate of the mean-square error

- Values of $\varepsilon_{\text{LOO}} \leq 10^{-2}$ are sufficient in most engineering applications

Outline

Polynomial chaos expansions

Computing and post-processing the PCE coefficients

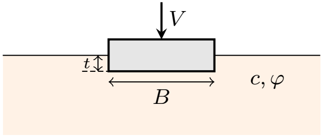
Sparse polynomial chaos expansions

Application examples
- Load bearing capacity
- Subsurface flow: global sensitivity analysis

Example: strip foundation

Load bearing capacity

$$P_{cr} = B\,\sigma_{cr} = B\left[c\,N_c + \gamma t\,N_q + \frac{1}{2}\gamma\,B N_\gamma\right]$$



with the load bearing factors:

$$N_q = e^{\pi\tan\varphi}\,\frac{1+\sin\varphi}{1-\sin\varphi}$$

$$N_c = (N_q - 1)/\tan\varphi$$

$$N_\gamma = 2\,(N_q - 1)\tan\varphi$$

Strip foundation – probabilistic model

Variable	Description	Distribution	Moments
γ	Self-weight	Gaussian	$\mu_\gamma = 21\ kN/m^3,\ COV_\gamma = 5\%$
c	Cohesion	Lognormal	$\mu_c = 5\ kPa,\ COV_c = 30\%$
φ	Effective friction angle	Lognormal	$\mu_\varphi = 30^\circ,\ COV_\varphi = 8\%$
B	Width	Deterministic	$3\ m$
t	Depth	Gaussian	$\mu_t = 0.5\ m,\ COV_t = 20\%$

Load bearing capacity

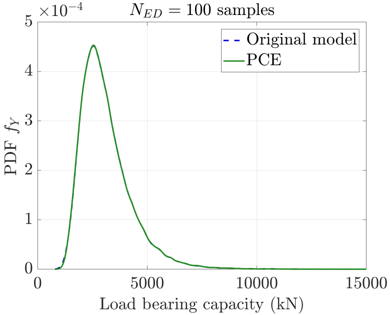
- A sparse polynomial chaos expansion is built from an experimental design of size $N_{\text{ED}} = 100$
- Mean, standard deviation and PDF are computed

```
% ------------ Polynomial chaos output ------------%
Number of input variables:    5
Maximal degree:               4
q-norm:                       1.00
Size of full basis:           70
Size of sparse basis:         33
Full model evaluations:       100
Leave-one-out error:          1.8327657e-05
Mean value:                 3123.5136
Standard deviation:         1168.5662
Coef. of variation:           37.412%
% -------------------------------------------------%
```

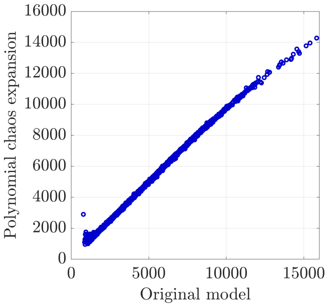
Distribution

The (kernel smoothing) density of the polynomial chaos expansion is plotted and compared to the one obtained from the original model (10^5 points)

$N_{\text{ED}} = 100$ points



PDF



Validation plot

PCE vs. Monte Carlo simulation (moments)

Reminder

N_{MCS}	100	$1,000$	$10,000$	$100,000$	$1,000,000$
Mean	3216	3082	3121	3125	3124
95% CI	[2942 – 3378]	[3057 – 3201]	[3105 – 3150]	[3115 – 3133]	[3122 – 3127]
Std. dev	1109	1080	1188	1173	1174
95% CI	[966 – 1565]	[1099 – 1313]	[1145 – 1207]	[1163 – 1185]	[1171 – 1178]

Polynomial chaos expansion

	Experimental design of size $N_{\text{ED}} = 100$
Mean	3123
95% CI	[3121 – 3125]
Std. dev	1169
95% CI	[1162 – 1171]

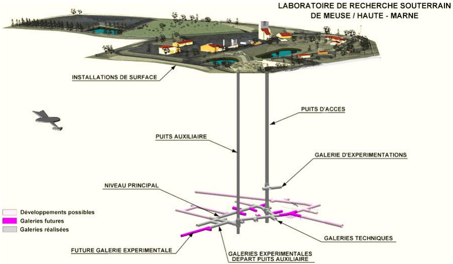
PCE vs. Monte Carlo simulation: Sobol' indices

N_{MCS}	100	1,000	10,000	100,000	1,000,000
γ	$[0.007 - 0.020]$	$[0.013 - 0.017]$	$[0.014 - 0.015]$	$[0.015 - 0.015]$	$[0.015 - 0.015]$
c	$[0.006 - 0.018]$	$[0.013 - 0.019]$	$[0.013 - 0.015]$	$[0.014 - 0.015]$	$[0.015 - 0.015]$
φ	$[0.917 - 1.201]$	$[0.872 - 1.014]$	$[0.965 - 1.003]$	$[0.958 - 0.969]$	$[0.963 - 0.966]$
t	$[0.004 - 0.012]$	$[0.009 - 0.013]$	$[0.011 - 0.012]$	$[0.011 - 0.012]$	$[0.012 - 0.012]$
N_{TOT}	600	6,000	60,000	600,000	**6,000,000**

	Experimental design of size $N_{ED} = 100$
γ	$[0.015 - 0.016]$
c	$[0.014 - 0.014]$
φ	$[0.962 - 0.964]$
t	$[0.011 - 0.012]$
N_{TOT}	**100**

Outline

Example: sensitivity analysis in hydrogeology



Source: http://www.futura-sciences.com/

Source: http://lexpansion.lexpress.fr/

- When assessing a nuclear waste repository, the Mean Lifetime Expectancy MLE(x) is the time required for a molecule of water at point x to get out of the boundaries of the system
- Computational models have numerous input parameters (in each geological layer) that are difficult to measure, and that show scattering

Geological model

Deman, Konakli, Sudret, Kerrou, Perrochet & Benabderrahmane, Reliab. Eng. Sys. Safety (2016)

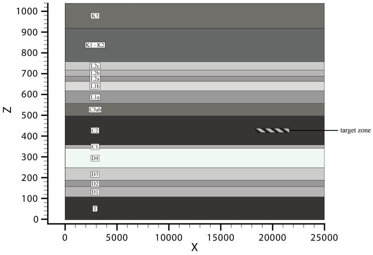
- Two-dimensional idealized model of the Paris Basin (25 km long / 1,040 m depth) with 5×5 m mesh $(10^6$ elements)
- Steady-state flow simulation with Dirichlet boundary conditions:

$$\nabla \cdot (\mathbf{K} \cdot \nabla H) = 0$$

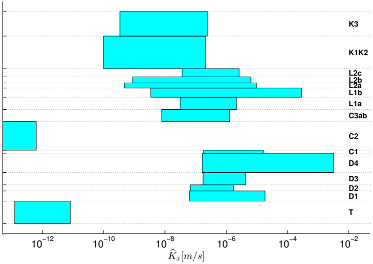
- 15 homogeneous layers with uncertainties in:
  - Porosity (resp. hydraulic conductivity)
  - Anisotropy of the layer properties (inc. dispersivity)
  - Boundary conditions (hydraulic gradients)

78 input parameters

Sensitivity analysis



Geometry of the layers

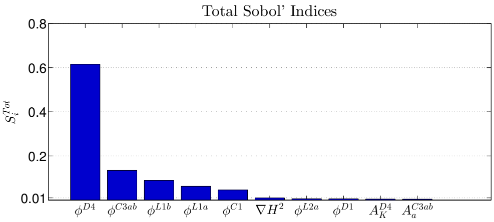


Conductivity of the layers

Question

What are the parameters (out of 78) whose uncertainty drives the uncertainty of the prediction of the mean life-time expectancy?

Sensitivity analysis: results



Parameter	$\sum_j S_j$
ϕ (resp. K_x)	0.8664
A_K	0.0088
θ	0.0029
α_L	0.0076
A_α	0.0000
∇H	0.0057

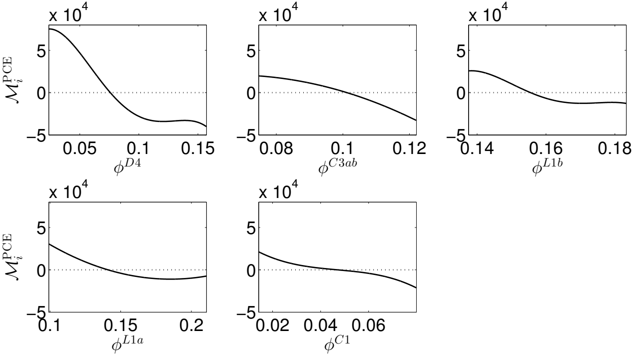
Conclusions

- Only 200 model runs allow one to detect the 10 important parameters out of 78
- Uncertainty in the porosity/conductivity of 5 layers explain 86% of the variability
- Small interactions between parameters detected

Bonus: univariate effects

The univariate effects of each variable are obtained as a straightforward post-processing of the PCE

$$\mathcal{M}_i(x_i) \stackrel{\text{def}}{=} \mathbb{E}\left[\mathcal{M}(\boldsymbol{X})|X_i = x_i\right],\ i = 1,\ \dots\ , M$$



Conclusions

- Polynomial chaos expansions are a mature, powerful technique for uncertainty propagation
- Nonintrusive methods are based on repeated runs of the computational model over an experimental design (similar to Monte Carlo simulation)
- Coefficients may be computed by least-square minimization, which has opened the path to sparse solvers
- Post-processing the coefficients gives the mean, variance, higher moments and global sensitivity indices. The output PDF is obtained by sampling the PC expansion
- All the algorithms described in this talk are available in UQLab (www.uqlab.com) !

Thank you very much for your attention

Questions ?

Chair of Risk, Safety & Uncertainty Quantification

www.rsuq.ethz.ch

The Uncertainty Quantification Software

www.uqlab.com

The Uncertainty Quantification Community

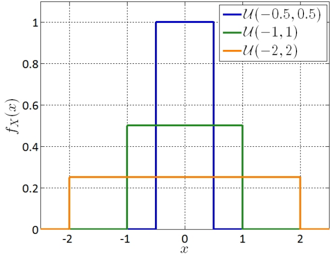
www.uqworld.org

APPENDIX

Legendre polynomials

Legendre polynomials are defined over $[-1,\,1]$ so as to be orthogonal with respect to the uniform distribution:

$$w(x) = 1/2 \qquad x \in [-1,\,1]$$



- Notation: $P_n(x),\; n \in \mathbb{N}$
- 3-term recurrence

$$P_0(x) = 1 \quad ; \quad P_1(x) = x$$

$$(n+1)P_{n+1}(x) = (2n+1)xP_n(x) - nP_{n-1}(x)$$

- P_n is solution of the ordinary differential equation

$$\left[(1-x^2)\,P_n'(x)\right]' + n(n+1)\,P_n(x) = 0$$

First Legendre polynomials

- The norm of the n-th Legendre polynomial reads:

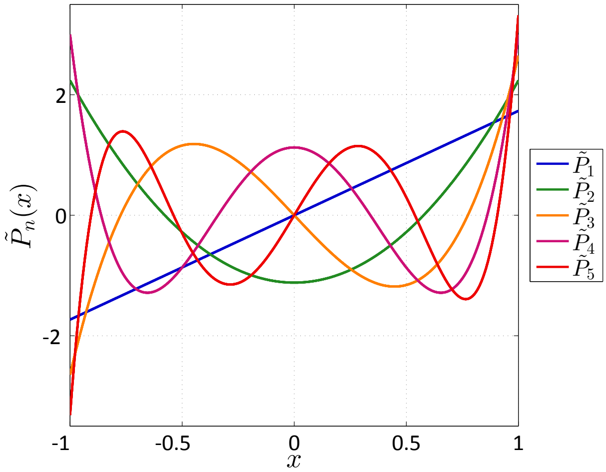
$$\| P_n \|^2 = < P_n \, , \, P_n > \int_{-1}^{1} P_n^2(x) \cdot \frac{1}{2} \mathrm{d}x = \frac{1}{2n+1}$$

- The orthonormal Legendre polynomials read:

$$\tilde{P}_n(x) = \sqrt{2n+1} \, P_n(x)$$

n	$P_n(x)$	$\| P_n \|^2$	$\tilde{P}_n(x)$
0	1	1	1
1	x	$1/3$	$\sqrt{3}\, P_1$
2	$\frac{1}{2}(3\,x^2 - 1)$	$1/5$	$\sqrt{5}\, P_2$
3	$\frac{1}{2}(5\,x^3 - 3\,x)$	$1/7$	$\sqrt{7}\, P_3$
4	$\frac{1}{8}(35\,x^4 - 30\,x^2 + 3)$	$1/9$	$\sqrt{9}\, P_4$
5	$\frac{1}{8}(63\,x^5 - 70\,x^3 + 15\,x)$	$1/11$	$\sqrt{11}\, P_5$

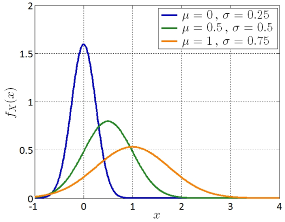
First Legendre polynomials



Hermite polynomials

Hermite polynomials are defined over $\mathbb{R}$ so as to be orthogonal with respect to the Gaussian distribution:

$$w(x) = \frac{1}{\sqrt{2\pi}} e^{-x^2/2} \qquad x \in \mathbb{R}$$



- Notation: $He_n(x), \quad n \in \mathbb{N}$
- 3-term recurrence:

$$He_0(x) = 1 \quad ; \quad He_1(x) = x$$

$$He_{n+1}(x) = x\, He_n(x) - n\, He_{n-1}(x)$$

- Normalization

$$\| He_n \|^2 = \int_{-\infty}^{+\infty} He_n^2(x) \, \frac{1}{\sqrt{2\pi}} e^{-x^2/2} dx = n! \qquad n! = 1 \cdot 2 \cdot 3 \ldots n$$

Hermite polynomials

- He_n is solution of the ordinary differential equation:

$$He_n''(x) - x\,He_n'(x) + n\,He_n(x) = 0$$

and satisfies:

$$He_n(x) = (-1)^n\, e^{x^2/2} \frac{d^n}{dx^n} \left(e^{-x^2/2} \right)$$

$$He_n'(x) = n\,He_{n-1}(x)$$

Important remark

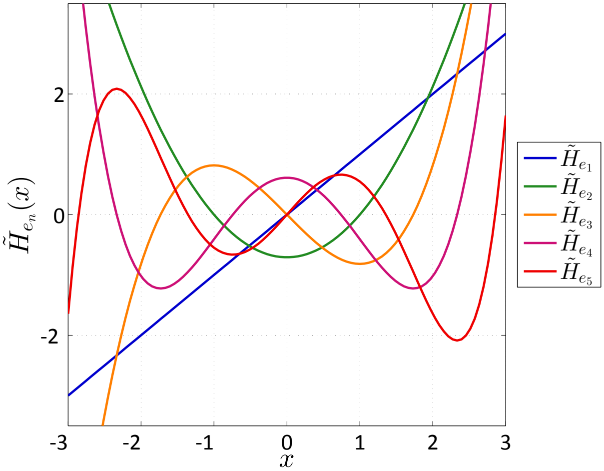
In the literature, two families of Hermite polynomials (HP) are known:

- The “physicist” HP are orthogonal w.r.t e^{-x^2}
- The “probabilistic” HP are orthogonal w.r.t the standard normal PDF $e^{-x^2/2}/\sqrt{2\pi}$

First “probabilistic” Hermite polynomials

n	$He_n(x)$	$\parallel He_n \parallel^2$	$\tilde{He}_n(x)$
0	1	1	He_0
1	x	1	He_1
2	$x^2 - 1$	2	$He_2/\sqrt{2}$
3	$x^3 - 3\,x$	6	$He_3/\sqrt{6}$
4	$x^4 - 6\,x^2 + 3$	24	$He_4/\sqrt{24}$
5	$x^5 - 10\,x^3 + 15\,x$	120	$He_5/\sqrt{120}$

First Hermite polynomials



Orthonormality of multivariate polynomials

Thus:

$$\begin{aligned}\mathbb{E}\left[\Psi_{\boldsymbol{\alpha}}(\boldsymbol{X})\Psi_{\boldsymbol{\beta}}(\boldsymbol{X})\right] &= \int_{\mathcal{D}_{\boldsymbol{X}}} \prod_{i=1}^{M} \left[P_{\alpha_i}^{(i)}(x_i)P_{\beta_i}^{(i)}(x_i)\, f_{X_i}(x_i)\right]\, d\boldsymbol{x} \\ &= \prod_{i=1}^{M} \left[\int_{\mathcal{D}_{X_i}} P_{\alpha_i}^{(i)}(x_i)P_{\beta_i}^{(i)}(x_i)\, f_{X_i}(x_i)dx_i\right] \\ &= \prod_{i=1}^{M} \delta_{\alpha_i\beta_i} \qquad \text{where } \delta_{\alpha_i\beta_i} = 1 \text{ if } \alpha_i = \beta_i \text{ and } 0 \text{ otherwise}\end{aligned}$$

As a consequence the orthogonality of the univariate polynomials propagates to the multivariate ones:

$$\mathbb{E}\left[\Psi_{\boldsymbol{\alpha}}(\boldsymbol{X})\Psi_{\boldsymbol{\beta}}(\boldsymbol{X})\right] = \delta_{\boldsymbol{\alpha}\boldsymbol{\beta}}$$

PCE coefficients as a projection

$$\varepsilon_P^2(\boldsymbol{X}) = \left(\sum_{j=0}^{P-1} y_j\, \Psi_j(\boldsymbol{X}) - \mathcal{M}(\boldsymbol{X})\right)^2$$

$$= \left(\sum_{j=0}^{P-1} y_j\, \Psi_j(\boldsymbol{X})\right)^2 + \mathcal{M}^2(\boldsymbol{X}) - 2\,\mathcal{M}(\boldsymbol{X}) \sum_{j=0}^{P-1} y_j\, \Psi_j(\boldsymbol{X})$$

$$= \sum_{j=0}^{P-1} \sum_{k=0}^{P-1} y_j\, y_k \Psi_j(\boldsymbol{X})\, \Psi_k(\boldsymbol{X}) + \mathcal{M}^2(\boldsymbol{X}) - 2 \sum_{j=0}^{P-1} y_j\, \mathcal{M}(\boldsymbol{X})\, \Psi_j(\boldsymbol{X})$$

$$\mathbb{E}\left[\varepsilon_P^2(\boldsymbol{X})\right] = \sum_{j=0}^{P-1} \sum_{k=0}^{P-1} y_j\, y_k\, \overbrace{\mathbb{E}\left[\Psi_j(\boldsymbol{X})\, \Psi_k(\boldsymbol{X})\right]}^{\delta_{jk}} + \mathbb{E}\left[\mathcal{M}^2(\boldsymbol{X})\right] - 2 \sum_{j=0}^{P-1} y_j\, \mathbb{E}\left[\mathcal{M}(\boldsymbol{X})\Psi_j(\boldsymbol{X})\right]$$

$$= \sum_{j=0}^{P-1} y_j^2 - 2 \sum_{j=0}^{P-1} y_j\, \mathbb{E}\left[\mathcal{M}(\boldsymbol{X})\Psi_j(\boldsymbol{X})\right] + \mathbb{E}\left[\mathcal{M}^2(\boldsymbol{X})\right]$$

PCE coefficients as a projection (cont')

$$\mathbb{E}\left[\varepsilon_P^2(\boldsymbol{X})\right] = \sum_{j=0}^{P-1} y_j^2 - 2\sum_{j=0}^{P-1} y_j\, \mathbb{E}\left[\mathcal{M}(\boldsymbol{X})\Psi_j(\boldsymbol{X})\right] + \mathbb{E}\left[\mathcal{M}^2(\boldsymbol{X})\right]$$

This is a quadratic function of the unknowns $\{y_j,\ j = 0, \ldots, P-1\}$

- The mean-square error is minimized when its derivative w.r.t each unknown coefficient y_j vanishes:

$$\frac{\partial \mathbb{E}\left[\varepsilon_P^2(\boldsymbol{X})\right]}{\partial y_j} = 2\, y_j - 2\,\mathbb{E}\left[\mathcal{M}(\boldsymbol{X})\Psi_j(\boldsymbol{X})\right] = 0$$

which reduces to:

$$\hat{y}_j = \mathbb{E}\left[\mathcal{M}(\boldsymbol{X})\,\Psi_j(\boldsymbol{X})\right] \qquad \forall\, j = 0, \ldots, P-1$$